



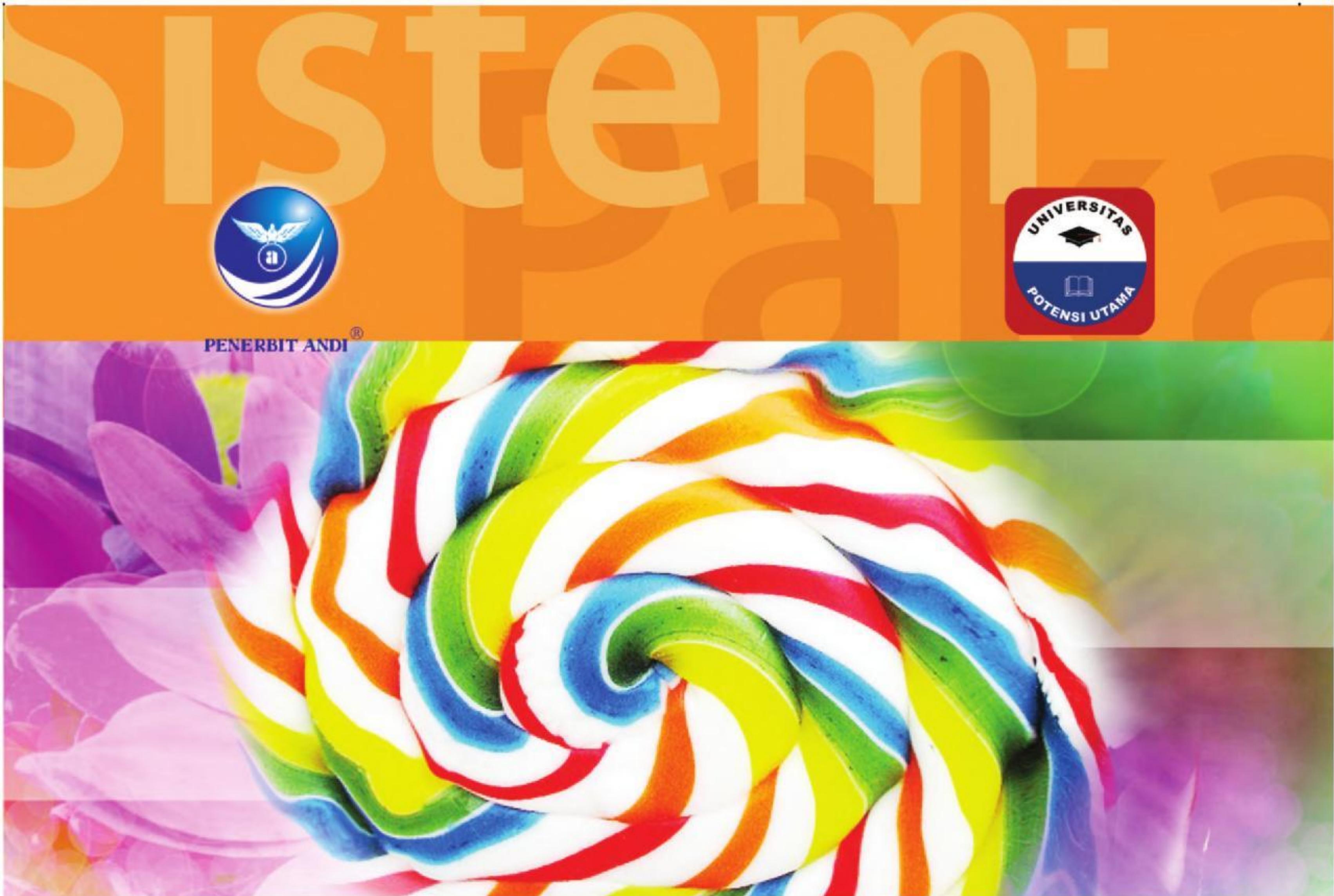
You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



PENERBIT ANDI®



sistem pakar

konsep dan teori

Rika Rosnelly

Sistem Pakar Konsep dan Teori

Oleh: Rika Rosnelly

Hak Cipta @ 2012 pada penulis.

Editor : Paulus Yesaya Jati
Setter : Arif Budi Permana
Desain Cover : R. Budo Wibowo
Korektor : Putri Christian

Hak Cipta dilindungi undang-undang. Dilarang memperbanyak atau memindahkan sebagian atau seluruh isi buku ini dalam bentuk apapun, baik secara elektronis maupun mekanis, termasuk memfotocopy, merekam atau dengan sistem penyimpanan lainnya, tanpa izin tertulis dari Penulis.

Penerbit: CV ANDI OFFSET (Penerbit ANDI)

Jl. Beo 38-40, Telp. (0274) 561881 (Hunting), Fax. (0274) 588282 Yogyakarta 55281

Percetakan: ANDI OFFSET

Jl. Beo 38-40, Telp. (0274) 561881 (Hunting), Fax. (0274) 588282 Yogyakarta 55281

Perpustakaan Nasional: Katalog dalam Terbitan (KDT)

Rosnelly, Rika

Sistem Pakar Konsep dan Teori /Rika Rosnelly

– Ed. I. – Yogyakarta: Andi,

20 - 19 - 18 - 17 - 16 - 15 - 14 - 13 - 12

viii + 122 hlm .; 16 x 23 Cm.

10 9 8 7 6 5 4 3 2 1

ISBN: 978 – 979 – 29 – 3416 – 8

I. Judul

I. Expert System - Computer

DDC'21: 006. 33

Daftar Isi

PRAKATA ----- iii

Daftar Isi ----- v

Bab 1 Konsep Sistem Pakar ----- 1

I.1 Apa itu sistem pakar ? ----- 2

I.2 Kelebihan Sistem Pakar ----- 5

I.3 Konsep Umum Sistem Pakar ----- 6

I.4 Elemen manusia pada sistem pakar ----- 9

Bab 2 Struktur Sistem Pakar ----- 13

II.1 Struktur Sistem Pakar ----- 13

II.2 Karakteristik Sistem Pakar ----- 20

Bab 3 Representasi Pengetahuan ----- 23

III.1 Definisi Pengetahuan (*Knowledge*) ----- 23

III.2 Jaringan Semantik ----- 26

III.3 Frame ----- 28

III.4 Script ----- 30

**Bab 4 Representasi Pengetahuan:
Logika dan Himpunan ----- 33**

IV.1 Logika dan Himpunan ----- 34

IV.1.1 Logika Proposisi ----- 37

IV.1.2 Logika Predikat ----- 44

**Bab 5 Representasi Pengetahuan:
Sistem Produksi ----- 49**

V.1 Sistem Produksi ----- 49

V.1 Sistem Produksi ----- 49

V.1.1 Definisi Sistem Produksi ----- 50

**V.1.2 Kaidah Produksi, Pengetahuan dan
Kaidah Inferensi ----- 53**

Bab 6 Metode Inferensi ----- 57

VI.1 Forward Chaining ----- 57

VI.2 Backward Chaining ----- 60

Bab 7 Metode Inferensi Fuzzy ----- 63

VII.1 Pendahuluan ----- 63

VII.2 Alasan Digunakannya Logika Fuzzy ----- 65

VII.3 Aplikasi ----- 65

VII.4 Himpunan Fuzzy ----- 66

VII.5 Fungsi Keanggotaan ----- 69

**Bab 8 Penalaran dengan
Ketidakpastian (Uncertainty) ----- 73**

VIII.1 Ketidakpastian (Uncertainty) ----- 73

VIII.2 Kesalahan dan Induksi ----- 76

VIII.3 Peluang ----- 77

Bab 9 Peluang dan Teorema Bayes ----- 79

Bab 10 Penalaran Inexact ----- 85

X.1 Ketidakpastian dan Kaidah ----- 85

X.2 Faktor Kepastian (Certainty Factor) ----- 89

**Bab 11 Tahapan Pengembangan
Sistem Pakar ----- 95**

Bab 12 Perancangan Sistem Pakar ----- 99

XII.1 Pendahuluan ----- 99

XII.2 Pemilihan Masalah yang Tepat ----- 100

XII.3 Jenis Alat Pengembangan ----- 102

**Bab 13 Penjelasan Baru untuk Penjelasan
Sistem Pakar ----- 105****XIII.1 Penjelasan Sistem Pakar ----- 105****XIII.2 Tipe Pengetahuan dalam****Penjelasan Sistem Pakar ----- 108****XIII.3 Asisten Keamanan ----- 109****XIII.4 Metodologi ----- 110****Bab 14 Project Pembuatan
Sistem Pakar ----- 113****Daftar Pustaka ----- 121**

BAB 1

Konsep Sistem Pakar

Pokok Bahasan

- I.1 Apa itu sistem pakar
- I.2 Kelebihan sistem pakar
- I.3 Konsep umum sistem pakar
- I.4 Elemen manusia pada sistem pakar

Mempelajari bab ini diharapkan Anda dapat memahami apa itu sistem pakar, kelebihan, konsep umum dan elemen manusia pada sistem pakar. Langkah pertama dalam menyelesaikan setiap masalah adalah dengan mendefinisikan terlebih dahulu ruang lingkup permasalahan tersebut atau domain untuk permasalahan yang akan diselesaikan. Hal ini juga berlaku untuk pemrograman *Artificial Intelligence* (AI). Namun karena hal-hal yang berkaitan dengan mistis berpadu dengan AI maka masih ada sesuatu yang melekat untuk tetap mempercayai pepatah lama yang merupakan bagian dari masalah AI jika masalah tersebut belum diselesaikan. Definisi yang populer dari AI lainnya adalah bahwa AI menjadikan komputer berakting dan bergaya seperti halnya para artis berakting di bioskop. Untuk saat ini banyak permasalahan dunia nyata yang diselesaikan menggunakan AI dan banyak juga aplikasinya yang dikomersialkan.

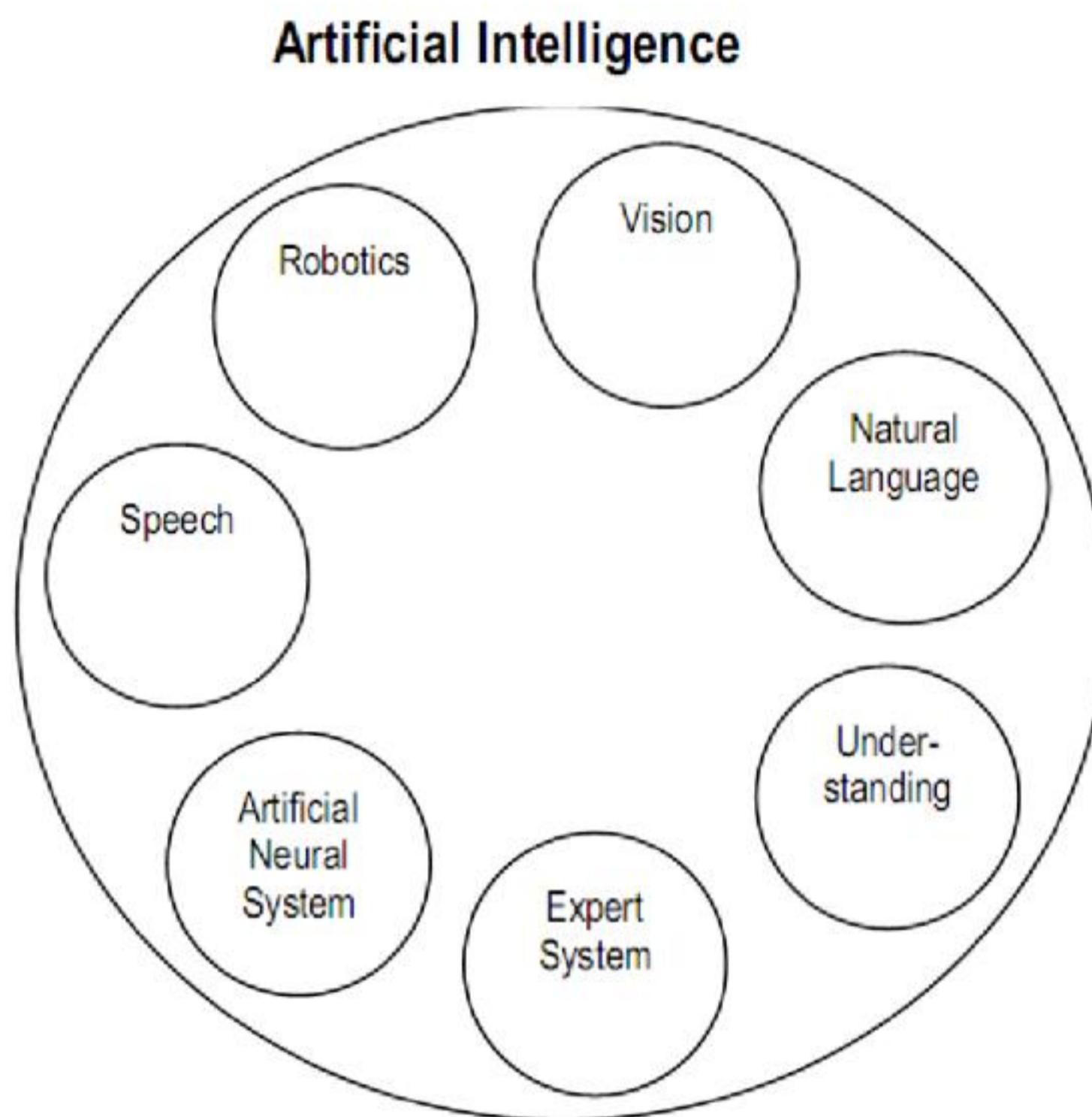


Walaupun penyelesaian umum untuk masalah AI klasik seperti translasi bahasa alami, pemahaman ucapan, dan visi belum ditemukan, tetapi pembatasan domain permasalahannya telah dapat menghasilkan suatu penyelesaian yang bermanfaat. Sebagai contoh, tidaklah terlalu sukar untuk membangun suatu sistem bahasa alami yang sederhana jika masukan dibatasi untuk kalimat dengan bentuk kata benda, kata kerja dan objek. Untuk saat ini sistem dari tipe ini bekerja sangat baik dalam menyediakan antarmuka yang familiar dengan pengguna untuk banyak produk perangkat lunak seperti sistem database dan *spreadsheets*.

I.1 Apa itu sistem pakar ?

Artificial intelligence (AI) memiliki beberapa domain masalah atau area seperti pada Gambar I.1 di bawah. Bidang sistem pakar merupakan penyelesaian pendekatan yang sangat berhasil dan bagus untuk permasalahan AI klasik dari pemrograman *intelligent* (cerdas). Sistem pakar (*expert system*) merupakan solusi AI bagi masalah pemrograman pintar (*intelligent*). Profesor Edward Feigenbaum dari Stanford University yang merupakan pionir dalam teknologi sistem pakar mendefinisikan sistem pakar sebagai sebuah program komputer pintar (*intelligent computer program*) yang memanfaatkan pengetahuan (*knowledge*) dan prosedur inferensi (*inference procedure*) untuk memecahkan masalah yang cukup sulit hingga membutuhkan keahlian khusus dari manusia.

Dengan kata lain, sistem pakar adalah sistem komputer yang ditujukan untuk meniru semua aspek (*emulates*) kemampuan pengambilan keputusan (*decision making*) seorang pakar. Sistem pakar memanfaatkan secara maksimal pengetahuan khusus selayaknya seorang pakar untuk memecahkan masalah.

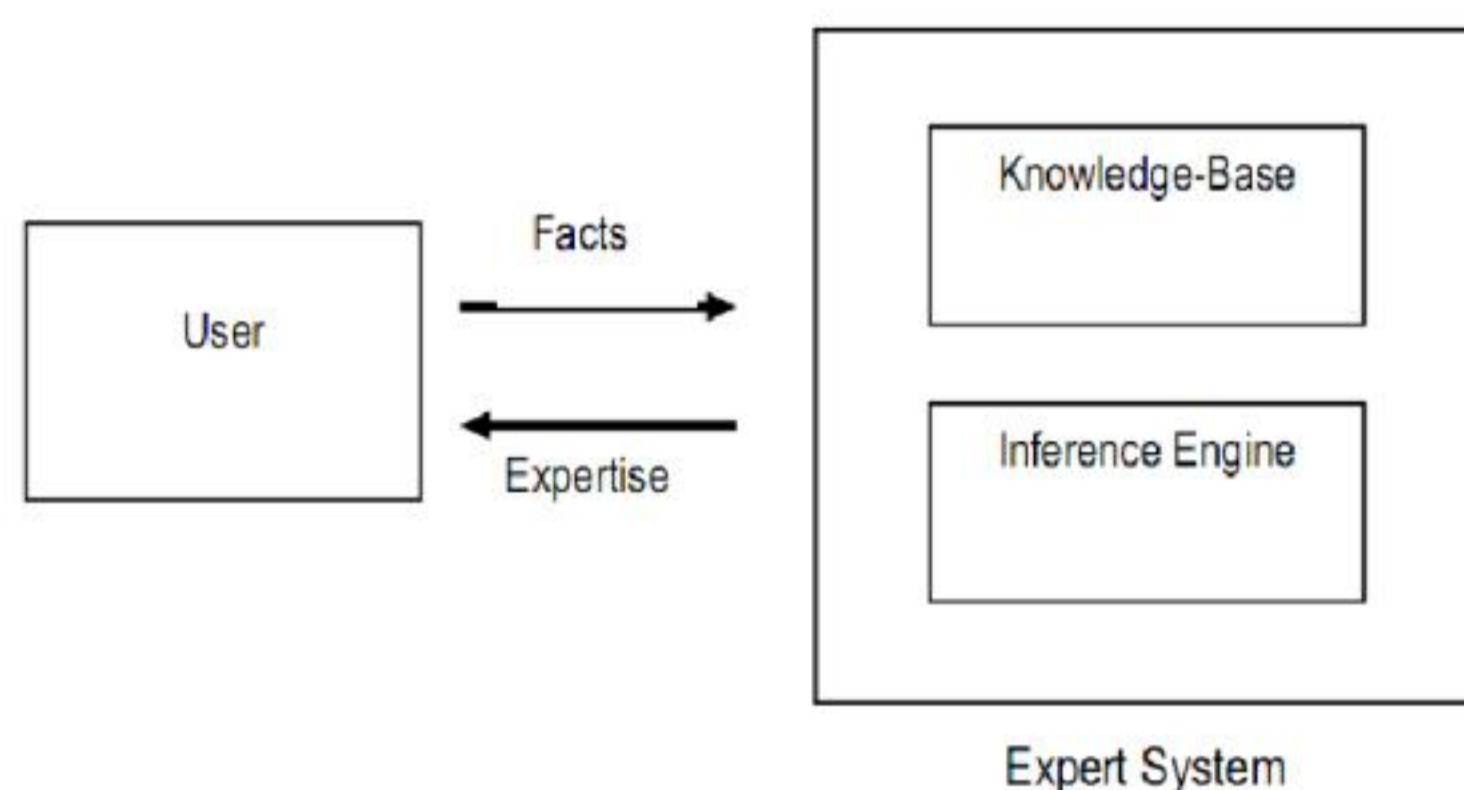


Gambar I.1 Area dari Artificial Intelligence (AI)

Pakar atau ahli (*expert*) didefinisikan sebagai seseorang yang memiliki pengetahuan atau keahlian khusus yang tidak dimiliki oleh kebanyakan orang. Seorang pakar dapat memecahkan masalah yang tidak mampu dipecahkan kebanyakan orang. Dengan kata lain, dapat memecahkan suatu masalah dengan lebih efisien namun bukan berarti lebih murah. Pengetahuan yang dimuat ke dalam sistem pakar dapat berasal dari seorang pakar atau pun pengetahuan yang berasal dari buku, jurnal, majalah, dan dokumentasi yang dipublikasikan lainnya, serta orang yang memiliki pengetahuan meskipun bukan ahli. Istilah sistem pakar (*expert system*), sering disinonimkan dengan sistem berbasis pengetahuan (*knowledge-based system*) atau sistem pakar berbasis pengetahuan (*knowledge based expert system*).

Gambar I.2 di bawah ini mengilustrasikan konsep dasar sistem pakar berbasis pengetahuan (*knowledge based expert system*). User memberikan informasi atau fakta kepada sistem dan menerima respon berupa saran ahli (*advice/expertise*). Secara internal, sistem terdiri dari

dua komponen utama yaitu basis pengetahuan (*knowledge based*), berisi pengetahuan yang akan digunakan oleh komponen lainnya yaitu mesin inferensi (*inference engine*) untuk menghasilkan kesimpulan sebagai respon terhadap kueri yang dilakukan user.



Gambar I.2 Konsep dasar fungsi Sistem Pakar Berbasis Pengetahuan

Pengetahuan yang dimiliki pakar bersifat spesifik dalam satu area masalah (*problem domain*). Area masalah merupakan satu wilayah masalah yang spesifik seperti kedokteran atau pengobatan (*medicine*), keuangan (*finance*), rekayasa (*engineering*) dan lainnya. Pengetahuan si pakar untuk memecahkan masalah yang spesifik tersebut dikenal sebagai area pengetahuan (*knowledge domain*). Gambar I.3 berikut ini menggambarkan hubungan antara area masalah (*problem domain*) dengan area pengetahuan (*knowledge domain*). Area pengetahuan seluruhnya berada dalam area masalah. Bagian yang berada di luar area pengetahuan menyatakan pengetahuan mengenai masalah yang tidak dimiliki oleh sistem. Sebuah sistem pakar umumnya tidak memiliki pengetahuan lain diluar area pengetahuannya kecuali jika diprogram dan dimuat ke dalam sistem. Misalkan sebuah sistem pakar yang memuat pengetahuan mengenai penyakit infeksi mungkin tidak memiliki pengetahuan lain dalam area masalah kedokteran. Dalam area pengetahuan yang dimiliki, sistem pakar melakukan inferensi atau membuat kesimpulan dengan cara yang sama seperti seorang pakar menarik kesimpulan.

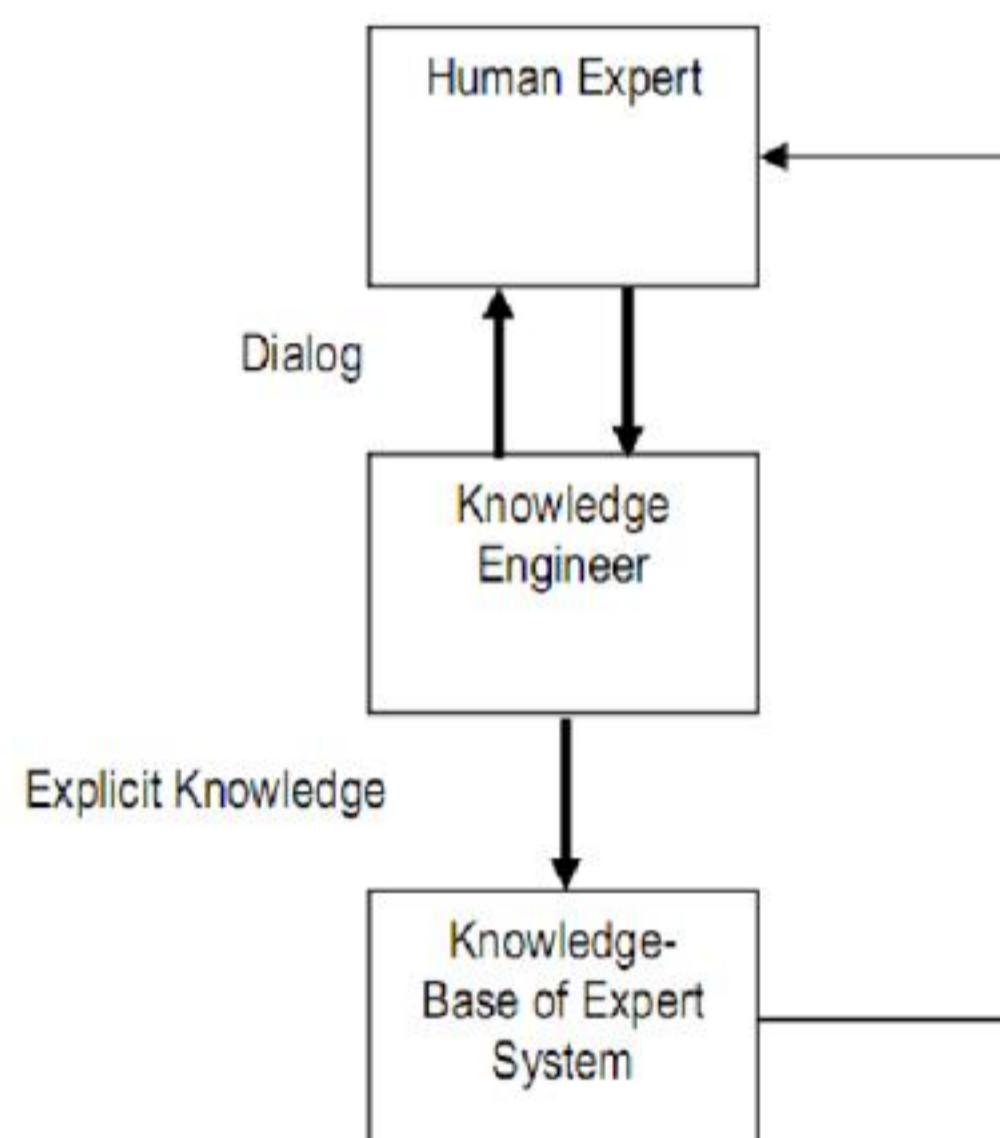
*image
not
available*

*image
not
available*



yang dibangun dengan mengekspresikan pengetahuan dalam bentuk rules. Bahkan, pendekatan berbasis pengetahuan (*knowledge-based approach*) untuk membangun sistem pakar telah mematahkan pendekatan awal yang digunakan pada sekitar tahun 1950-an dan 1960-an yang menggunakan teknik penalaran (*reasoning*) yang tidak mengandalkan pengetahuan.

Pengetahuan tidak tertulis yang dimiliki oleh seorang pakar harus diekstraksi melalui wawancara secara ekstensif oleh *knowledge engineer*. Proses pengembangan sistem pakar yang berhubungan dengan perolehan pengetahuan dari pakar maupun sumber lain dan kodingnya disebut sebagai *knowledge engineering* yang dilaksanakan oleh *knowledge engineer*. Tahapan pengembangan sistem pakar secara umum tergambar pada Gambar I.4 di bawah ini.



Gambar I.4 Pengembangan Sistem Pakar

Tahap awal, *knowledge engineer* melakukan diskusi dengan pakar untuk mengumpulkan pengetahuan yang dimiliki pakar yang bersangkutan. Tahap ini serupa dengan proses diskusi persyaratan atau kebutuhan yang dilakukan *system engineer* pada sistem konvensional dengan kliennya. Setelah itu *knowledge engineer* melakukan koding pengetahuan secara eksplisit ke dalam *knowledge base*. Pakar kemudian mengevaluasi

*image
not
available*

*image
not
available*

*image
not
available*



mengecek tali kipas dan menginterpretasikan arti dari lepas atau hilangnya tali dengan akibat-akibat yang ditimbulkannya. Ini adalah contoh keahlian. Jika digunakan lebih dari satu pakar, situasi akan menjadi sulit jika pakar-pakar saling tidak setuju. Perlu ada kesepakatan antar pakar dalam menyelesaikan permasalahan.

Pembangun/Pembuat Pengetahuan

Pembangun pengetahuan memiliki tugas utama menerjemahkan dan merepresentasikan pengetahuan yang diperoleh dari pakar, baik berupa pengalaman pakar dalam menyelesaikan masalah maupun sumber terdokumentasi lainnya ke dalam bentuk yang bisa diterima oleh sistem pakar. Dalam hal ini pembangun pengetahuan (*knowledge engineer*) menginterpretasikan dan merepresentasikan pengetahuan yang diperoleh dalam bentuk jawaban-jawaban atas pertanyaan-pertanyaan yang diajukan pada pakar atau pemahaman, penggambaran analogis, sistematis, konseptual yang diperoleh dari membaca beberapa dokumen cetak seperti *text book*, jurnal, makalah, dan sebagainya, Kurangnya pengalaman *knowledge engineer* merupakan kesulitan utama dalam mengkonstruksi sistem pakar. Untuk mengatasi hal tersebut, perancang sistem pakar menggunakan *tools* komersial.(Seperti pada editor-editor khusus maupun *logic debuggers*) dan usahanya akan dipusatkan pada pembangunan mesin inferensi.

Pembangun/Pembuat Sistem

Pembangun sistem adalah orang yang bertugas untuk merancang antarmuka pemakai sistem pakar, merancang pengetahuan yang sudah diterjemahkan oleh pembangun pengetahuan ke dalam bentuk yang sesuai dan dapat diterima oleh sistem pakar dan mengimplementasikannya ke dalam mesin inferensi. Selain hal tersebut, pembangun sistem juga bertanggung jawab apabila sistem pakar akan diintegrasikan dengan sistem komputerisasi lain. Alat pembangun (*tool builder*) dapat dipakai untuk menyajikan atau membangun *tool* yang spesifik. Penjual (*vendor*) dapat memberikan *tool* dan saran, staf

*image
not
available*

*image
not
available*

*image
not
available*



4. **Explanation facility**

Menyediakan kebenaran dari solusi yang dihasilkan kepada user (*reasoning chain*).

5. **Knowledge acquisition facility**

Meliputi proses pengumpulan, pemindahan dan perubahan dari kemampuan pemecahan masalah seorang pakar atau sumber pengetahuan terdokumentasi ke program computer, yang bertujuan untuk memperbaiki atau mengembangkan basis pengetahuan.

6. **User Interface**

Mekanisme untuk memberi kesempatan kepada user dan sistem pakar untuk berkomunikasi. Antar muka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu antarmuka menerima informasi dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.

Pada umumnya, antar muka pemakai juga berfungsi untuk menginputkan pengetahuan baru kedalam basis pengetahuan sistem pakar, menampilkan fasilitas penjelasan sistem dan memberikan tuntunan penggunaan sistem secara menyeluruh langkah demi langkah sehingga pemakai mengerti apa yang harus dilakukan terhadap sistem.

Syarat utama membangun antar muka pemakai adalah kemudahan dalam menjalankan sistem. Semua kesulitan dalam membangun suatu program harus disembunyikan, yang ditampilkan hanyalah tampilan yang interaktif, komunikatif dan kemudahan pakai.

*image
not
available*

*image
not
available*

*image
not
available*



situation-action cycle. Dalam sebuah siklus, *inference engine* akan mengeksekusi sekumpulan tugas secara berulang hingga terdapat kriteria tertentu yang mengakibatkan eksekusi berhenti. Pada sebuah siklus mungkin terdapat lebih dari satu aturan yang aktif dan aturan yang masih aktif yang berasal dari siklus sebelumnya dalam agenda. Sehingga jumlah aturan yang aktif dalam agenda berubah-ubah seiring dengan berjalannya eksekusi. *Inference engine* akan mengeksekusi aksi dengan prioritas tertinggi, kemudian prioritas tinggi berikutnya dan seterusnya hingga tidak ada lagi yang perlu dieksekusi. Berbagai *shell* yang ada memiliki berbagai jenis skema prioritas. Namun umumnya, *shell* membebaskan *knowledge engineer* untuk mendefinisikan prioritas aturan.

Konflik di agenda terjadi jika aturan aktif yang berbeda memiliki prioritas yang sama. Dalam hal ini, *inference engine* harus memilih aturan yang akan dieksekusi. Beberapa *shell* memiliki pendekatan yang berbeda untuk menyelesaikan masalah ini. Paradigma Newell dan Simon mengasumsikan bahwa aturan yang pertama dimuat ke dalam sistem memiliki prioritas tertinggi. OPS5 menganggap aturan yang memiliki pola yang lebih kompleks memiliki prioritas yang lebih tinggi. Pada ART dan CLIPS, aturan memiliki default prioritas yang sama kecuali jika ditetapkan prioritas tertentu oleh *knowledge engineer*.

Setelah tahap ini, kendali sistem pakar beralih ke *command interpreter* tingkat tinggi (*top-level command interpreter*) yang digunakan user untuk memberikan instruksi lebih lanjut pada *shell* sistem pakar. *Top level command interpreter* adalah user interface bagi *shell* selama sistem pakar dalam proses pengembangan. *User interface* yang lebih canggih umumnya dirancang untuk memfasilitasi operasi yang dilakukan sistem. Seperti misalnya *user interface* untuk memonitor atau mengendalikan pabrik yang menampilkan diagram blok dengan *display* beresolusi tinggi dan *bit-mapped color*. Faktanya, lebih banyak usaha yang dilakukan untuk merancang dan mengimplementasikan

*image
not
available*

*image
not
available*

*image
not
available*

BAB 3

Representasi Pengetahuan

Pokok Bahasan

III.1 Definisi Pengetahuan (*Knowledge*)

III.2 Jaringan Semantik

III.3 Frame

III.4 Script

Mempelajari bab ini Anda diharapkan dapat memahami definisi *Knowledge*, jaringan semantik, *frame* dan *script*. Kata *knowledge* (pengetahuan), seperti juga halnya kata *love* (cinta), adalah salah satu kata yang setiap orang tahu makna dan pengertiannya. Seperti halnya *love*, *knowledge* juga mempunyai pengertian yang bermacam-macam. *Knowledge* sering disinonimkan dengan data, fakta dan informasi.

III.1 Definisi Pengetahuan (*Knowledge*)

Pelajaran dari *knowledge* merupakan suatu *epistemology* (Angels, 81) yang merupakan bagian dari ilmu filsafat yang membahas tentang asal. Hal ini berkenaan dengan sifat, struktur dan keaslian dari *knowledge*. Gambar III.1 menggambarkan beberapa kategori dari *epistemology*. Di samping jenis filosofi yang dikemukakan oleh Aristoteles, Plato, Descartes, Hume, Kant dan lainnya ada juga tipe khusus yang dinamakan *priori* dan *posterior*. Istilah *priori* berasal dari bahasa Latin yang berarti “yang mendahului”. *Priori knowledge* dianggap menjadi kebenaran

*image
not
available*

*image
not
available*

*image
not
available*

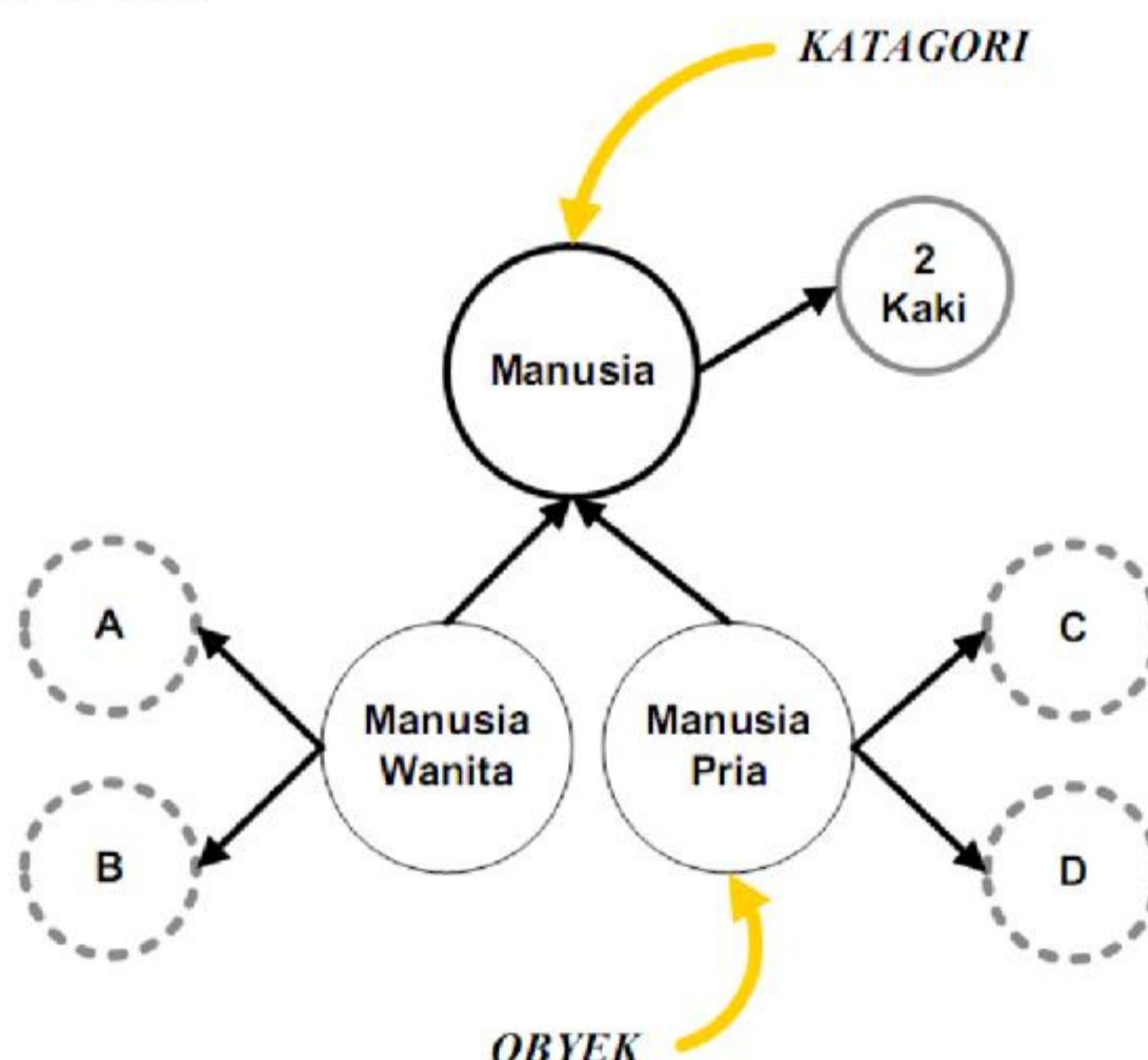


a. Katagori

Objek-objek yang mempunyai kemiripan karakteristik dapat digolongkan dalam kategori tertentu. Kategori merupakan pengorganisasian objek yang merupakan representasi pengetahuan yang vital dalam *semantic net*. Sebagai contoh : jika ada dua objek "elang" dan "perkutut", maka kedua objek tersebut dapat diturunkan dari kategori "burung".

b. Objek

Objek merupakan individu tersendiri yang mempunyai sifat-sifat karakteristik yang spesifik. Dalam kaitan kategori, jika sebuah objek diturunkan dari kategori, maka objek akan mempunyai sifat dari katagori secara keseluruhan dan juga mempunyai sifat spesifik objek itu sendiri. Berikut contoh jaringan semantik ditunjukkan pada Gambar III.3



Gambar III.3 Contoh Jaringan Semantik

Dari Gambar III.3. dapat dilihat bahwa objek "Manusia pria" merupakan turunan dari katagori "Manusia", objek "Manusia pria" selain mempunyai sifat-sifat individu (C dan D) juga mempunyai sifat dari "Manusia" yaitu punya 2 kaki.

*image
not
available*

*image
not
available*

*image
not
available*



pada *script* menggunakan serangkaian slot yang berisi informasi tentang orang, objek dan tindakan-tindakan yang terjadi dalam suatu peristiwa.

Contoh sederhana dapat dilihat untuk *script* berangkat ke restoran seperti berikut ini :

SCRIPT Restoran

Jalur (*Track*) : Restoran swalayan (fast food)

Peran (*Roles*) : Tamu pelayan

Pendukung(*Prop*) : Counter, baki, makanan, uang, serbet, garam, merica, kecap sedotan, dan lain-lain

Kondisi masukan : tamu lapar-tamu punya uang

Adegan (*Scene*) 1 : Masuk

- » Tamu parkir mobil.
- » Tamu masuk restoran.
- » Tamu Antri.
- » Tamu baca menu di list menu dan mengambil putusan tentang apa yang akan diminta.

Adegan 2 : Pesanan

- » Tamu memberikan pesanan pada pelayan.
- » Pelayan mengambil pesanan dan meletakkan makanan di atas baki.
- » Tamu membayar.

Adegan 3 : Makan

- » Tamu mengambil serbet, sedotan, garam, dan lain-lain.
- » Tamu membawa baki makanan ke meja kosong.
- » Tamu makan dengan cepat.

Adegan 4 : Pulang

- » Tamu membersihkan meja.
- » Tamu membuang sampah.
- » Tamu meninggalkan restoran.
- » Tamu naik mobil dan pulang.

*image
not
available*

*image
not
available*

*image
not
available*



Dalam istilah matematika, lingkaran dari diagram Venn merepresentasikan sebuah himpunan yang merupakan kumpulan objek-objek. Objek dalam himpunan dinamakan dengan elemen/unsur. Beberapa contoh himpunan :

$$A=\{1,3,5,7\}$$

$$B=\{1,2,3,4,5\}$$

$$C=\{0,2,4\dots\}$$

$$D=\{\dots,-4,-2,0,2,4,\dots\}$$

$$E=\{\text{pesawat,ballon,blimps,jet}\}$$

$$F=\{\text{pesawat,ballon}\}$$

Di mana titik-titik,..., dinamakan *ellipses*, yang menunjukkan istilah seterusnya sampai tak berhingga.

Simbol Yunani epsilon ϵ menunjukkan bahwa suatu elemen merupakan anggota dari suatu himpunan. Sebagai contoh, $1 \in A$, artinya bilangan 1 adalah elemen dari himpunan A yang telah didefinisikan sebelumnya. Jika suatu elemen bukan anggota dari suatu himpunan maka simbol yang digunakan adalah \notin . Misalnya $2 \notin A$.

Jika dua himpunan sembarang, misalkan X dan Y, didefinisikan bahwa setiap elemen X merupakan elemen Y, maka x adalah subset dari Y dan dengan simbol matematika dapat dituliskan dalam bentuk $X \subset Y$ atau $Y \supset X$. Dari definisi tentang himpunan bagian, maka dapat dikatakan bahwa setiap himpunan merupakan himpunan bagian dari dirinya sendiri. Sedangkan himpunan bagian yang bukan merupakan himpunan dari dirinya sendiri dinamakan dengan *proper subset*. Sebagai contoh, himpunan yang telah didefinisikan sebelumnya adalah merupakan *proper subset* dari Y.

Gambar IV.2 menggambarkan himpunan bagian dalam bentuk irisan (*intersection*) dari dua himpunan.

*image
not
available*

*image
not
available*

*image
not
available*



jika kedua proposisi bernilai salah. Tabel IV.3 menunjukkan tabel kebenaran untuk operator OR.

Tabel IV.3 Tabel Kebenaran Operator OR

P	Q	P OR Q
B	B	B
B	S	B
S	B	B
S	S	S

d. Implikasi

Implikasi : Jika P maka Q akan menghasilkan nilai salah jika P benar dan Q salah, selain itu akan selalu bernilai benar. Tabel IV.4 menunjukkan tabel kebenaran untuk operator Implikasi.

Tabel IV.4 Tabel Kebenaran Operator OR

P	Q	$P \rightarrow Q$
B	B	B
B	S	S
S	B	B
S	S	B

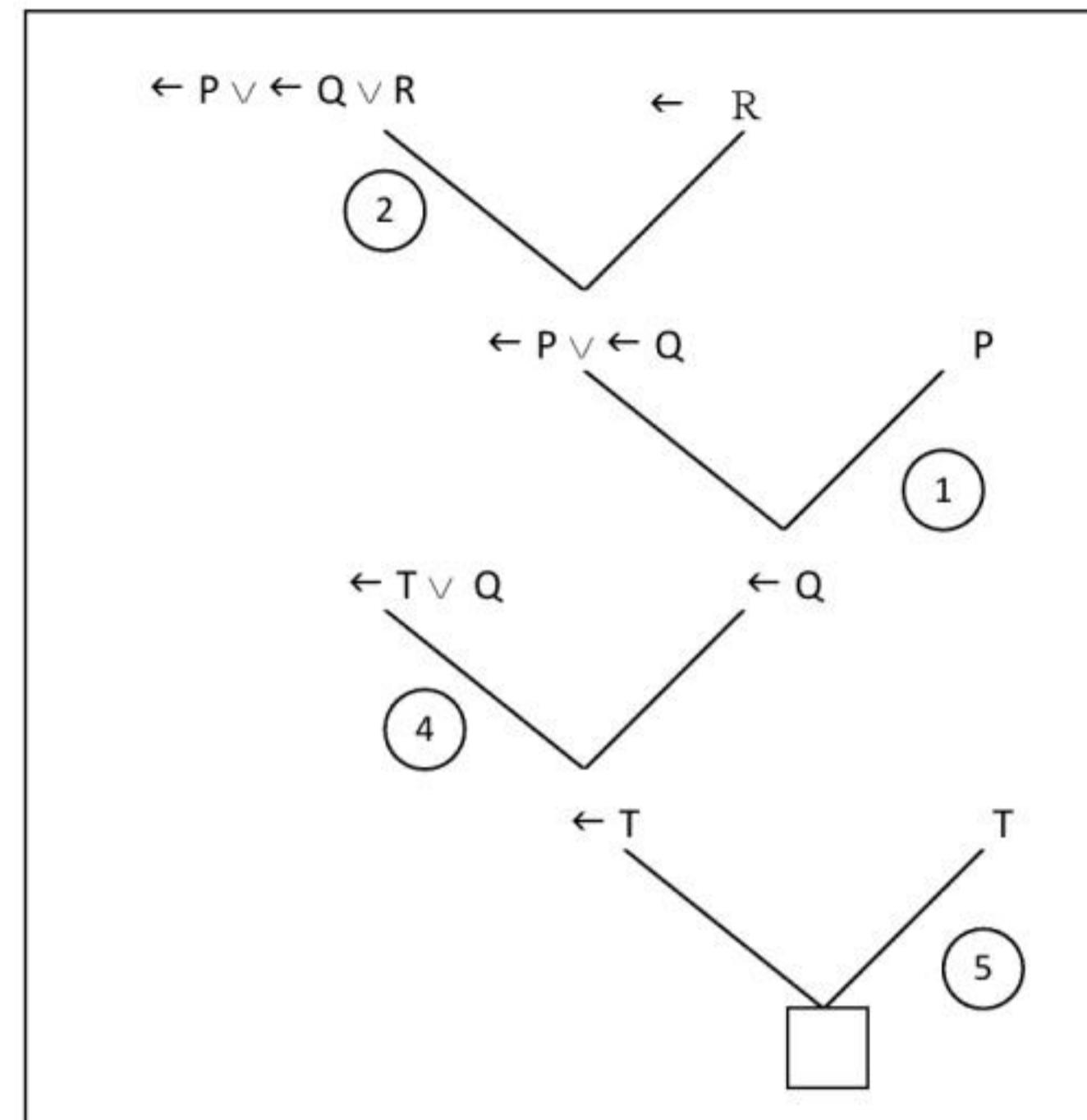
e. Ekuivalensi

Ekuivalensi akan menghasilkan nilai benar jika P dan Q keduanya benar atau keduanya benar atau keduanya salah, Tabel IV.5 menunjukkan tabel kebenaran untuk operator ekuivalensi.

*image
not
available*

*image
not
available*

*image
not
available*



Gambar IV.4 Resolusi Pada Logika Proposisi

Contoh apabila diterapkan dalam kalimat :

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

Kalimat yang terbentuk :

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

Setelah dilakukan ke bentuk CNF, didapat :

- Fakta ke-2 : Andi tidak cerdas atau Andi tidak rajin belajar atau Andi akan menjadi juara kelas
- Fakta ke-3 : Andi tidak makan banyak atau Andi rajin belajar
- Fakta ke-4 : Andi tidak cukup istirahat atau Andi rajin belajar.

*image
not
available*

*image
not
available*

*image
not
available*

**Latihan**

1. Pahami kembali logika proposisi dan logika predikat.
2. Buatlah contoh rancangan representasi pengetahuan dengan menggunakan logika proposisi dan logika predikat.

*image
not
available*

*image
not
available*

*image
not
available*



THEN [object mempunyai kualitas luar]

IF [kategori adalah sebuah ukuran OR kualitas luar OR massa OR suatu zat]

THEN [object mempunyai kualitas fisik]

.....

Struktur intuitif secara alami untuk representasi pengetahuan yang telah dihasilkan oleh sistem produksi merupakan metode yang lebih disukai oleh banyak sistem pakar, termasuk perintis program DENDRAL dan MYCIN. Sebagai contoh, kaidah produksi dari sistem tersebut seperti yang ditunjukkan dibawah ini.

DENDRAL

IF [the spectrum of the molecule has two peaks at masses x_1 dan x_2 such that

$x_1 + x_2 = M + 28$ AND

$x_1 - 28$ is a high peak AND

$x_1 - 28$ is a high peak AND

at least one of x_1 or x_2 is high]

THEN [the molecule contains a ketone group]

MYCIN

IF [a) the stain of the organism is gramneg AND

b) the morphology of the organism is rod AND

c) the patient is a compromised host]

THEN [there is suggestive evidence (0,6) that the identity of the organism is pseudomonas]

Dari kesamaan perumusan kedua kaidah produksi itu, kita catat suatu gambaran atraktif pertama dari sistem produksi secara formal,

*image
not
available*

*image
not
available*

*image
not
available*



- Kaidah 2 IF tingkat inflasi mengalami kemunduran THEN harga emas turun.
- Kaidah 3 IF konflik internasional berlangsung selama lebih dari 7 hari AND IF konflik ini terjadi di Timur Tengah THEN beli emas.

Kaidah Inferensi (prosedural) seperti yang diberikan berikut ini.

- Kaidah 1 IF Data dibutuhkan tidak dalam sistem THEN permintaannya dari pengguna.
- Kaidah 2 IF lebih dari satu kaidah digunakan THEN nonaktifkan kaidah sembarang yang tidak menambah data baru.
- Kaidah inferensi berisi kaidah tentang kaidah. Tipe kaidah ini dinamakan dengan *metarule*. Mereka menynggung kaidah lain (atau bahkan sesama mereka).

Perekayasa pengetahuan memisahkan dua tipe ini. Kaidah pengetahuan menjadi basis pengetahuan, sedangkan kaidah inferensi menjadi bagian dari mesin inferensi.

Representasi kaidah terutama dapat diaplikasikan bila dibutuhkan untuk merekomendasikan suatu bagian aksi berdasarkan kejadian yang dapat diobservasi. Ada beberapa keuntungan penggunaan kaidah, yaitu:

- Kaidah mudah dimengerti. Mereka mudah disampaikan karena merupakan bentuk alami dari pengetahuan.
- Inferensi dan penjelasan mudah diperoleh atau diturunkan.
- Modifikasi dan perawatan relatif lebih mudah.
- Ketidakpastian lebih mudah dikombinasikan dengan kaidah.
- Setiap kaidah sering saling independent dari semua kaidah.

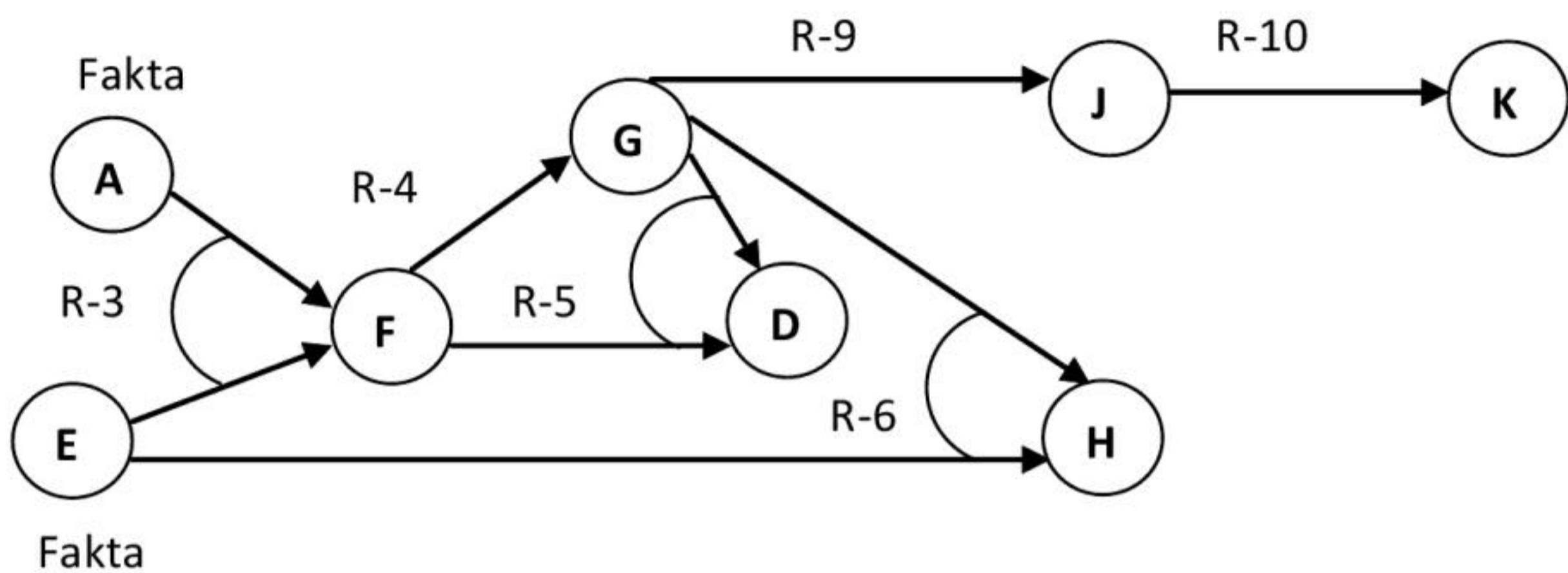
Selain itu terdapat pula keterbatasan dari representasi kaidah ini, yaitu:

- Pengetahuan yang kompleks membutuhkan beribu-ribu kaidah, yang mungkin agak sukar membuatnya, baik untuk menggunakan sistem maupun untuk perawatannya.
- Pembangun menyukai kaidah oleh karenanya mereka mencoba

*image
not
available*

*image
not
available*

*image
not
available*



Gambar VI.1 *Forward Chaining*

VI.2 Backward Chaining

Backward Chaining merupakan 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 dicari harus dicari fakta-fakta yang ada dalam basis pengetahuan.

Contoh VI.2 :

Seperti halnya pada Contoh VI.1 pada Tabel VI.1 terlihat ada 10 aturan yang tersimpan dalam basis pengetahuan. Fakta awal yang diberikan hanya : A & F (artinya : A dan F bernilai benar).

Ingin dibuktikan apakah K bernilai benar (hipotesis : K)?

- Langkah-langkah inferensi adalah sebagai berikut :
- Pertama-tama kita cari terlebih dahulu mulai dari R-1, aturan mana yang memiliki konsekuensi K. Ternyata setelah ditelusur, aturan dengan konsekuensi K baru ditemukan pada R-10. Untuk membuktikan bahwa K benar, maka perlu dibuktikan bahwa J benar.
- Kita cari aturan yang memiliki konsekuensi J. Kita mulai dari R-1, dan ternyata kita baru akan menemukan aturan dengan konsekuensi J pada R-8. Untuk membuktikan bahwa J benar, maka perlu dibuktikan bahwa I dan A benar. Untuk membuktikan

*image
not
available*

*image
not
available*

*image
not
available*



Antara *input* dan *output* terdapat satu kotak hitam yang harus memetakan *input* ke *output* yang sesuai.

VII.2 Alasan Digunakannya Logika Fuzzy

Ada beberapa alasan mengapa orang menggunakan logika fuzzy, antara lain :

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

VII.3 Aplikasi

Beberapa aplikasi logika fuzzy, antara lain :

1. Pada tahun 1990 pertama kali dibuat 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 adalah seberapa kotor, jenis kotoran, dan 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. Disamping itu, sistem juga dapat menentukan jenis kotoran (daki atau minyak).

*image
not
available*

*image
not
available*

*image
not
available*

pendekatan fungsi. Dari gambar VII.3 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 :

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

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

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

Gambar VII.3 Tiga himpunan Fuzzy

a. Aturan Rule If Then Fuzzy

- Aturan IF-THEN fuzzy adalah pernyataan IF-THEN dimana beberapa kata-kata dalam pernyataan tersebut ditentukan oleh fungsi keanggotaan.
- Aturan produksi fuzzy adalah relasi fuzzy antara dua proposisi. Aturan tersebut dinyatakan dalam bentuk :

IF (proposisi fuzzy 1) THEN (proposisi fuzzy 2)



Disebut antecedent/premis Disebut consequent/kesimpulan

*image
not
available*

*image
not
available*

*image
not
available*

secara akurat. Contoh berikut mengungkapkan bahwa ketakpastian dan ketakpersisan dapat muncul secara bebas satu sama lain.

VIII.2 Kesalahan dan Induksi

Proses induksi merupakan lawan dari deduksi. Deduksi merupakan hasil dari hal yang umum ke khusus, seperti :

Semua laki-laki adalah makhluk hidup.

Socrates adalah laki-laki

Dapat ditarik kesimpulan :

Socrates adalah makhluk hidup.

Induksi mencoba untuk menggeneralisasikan dari hal khusus ke umum, seperti :

Disk saya belum pernah rusak

∴ Disk saya tidak akan pernah rusak

Simbol ∴ mewakili "oleh karena" untuk induksi dan ∵ mewakili "oleh karena" untuk deduksi.

Kecuali untuk induksi matematika, argumen induksi tidak pernah dapat dibuktikan dengan benar. Bahkan argumen induksi hanya dapat menyediakan beberapa tingkat kepercayaan bahwa konklusi tersebut adalah benar. Kita belum mempunyai kepercayaan yang lebih pada argumen induktif terdahulu. Berikut ini adalah contoh argumen yang lebih kuat :

Alarm kebakaran berbunyi

∴ ada kebakaran

*image
not
available*

*image
not
available*

*image
not
available*

BAB 10

Penalaran Inexact

Pokok Bahasan

X.1 Ketidakpastian dan Kaidah

X.2 Faktor Kepastian (*Certainty Factor*)

Setelah mempelajari bab ini Anda diharapkan dapat memahami ketidakpastian dan kaidah, faktor kepastian, dan dapat menerapkan metode *certainty factor* pada kasus.

X.1 Ketidakpastian dan Kaidah

Salah satu karakteristik umum dari suatu informasi yang tersedia untuk seorang ahli adalah ketidak sempurnaan atau kecacatan. Informasi bisa jadi tidak lengkap, tidak konsisten, tidak tentu dan sebagainya. Dengan kata lain informasi sering tidak sesuai untuk meyelesaikan suatu permasalahan, akan tetapi seorang pakar dapat mengatasi kerusakan dan biasanya dapat membuat suatu pertimbangan benar dan keputusan yang benar. Sistem pakar juga harus mampu untuk mengatasi ketidakpastian dan menggambarkan suatu konklusi yang valid.

*image
not
available*

*image
not
available*

*image
not
available*

$$MB[h, e_1 \wedge e_2] = \begin{cases} 0 & MD[h, e_1 \wedge e_2] = 1 \\ MB[h, e_1] + MB[h, e_2] \cdot (1 - MB[h, e_1]) & \text{otherwise} \end{cases}$$

$$MD[h, e_1 \wedge e_2] = \begin{cases} 0 & MB[h, e_1 \wedge e_2] = 1 \\ MD[h, e_1] + MD[h, e_2] \cdot (1 - MD[h, e_1]) & \text{otherwise} \end{cases}$$

Contoh X.1 :

Andaikan suatu observasi memberikan kepercayaan terhadap h dengan $MB[h, e_1] = 0,3$ dan $MD[h, e_1] = 0$.

$$\text{sehingga } CF[h, e_1] = 0,3 - 0 = 0,3$$

Jika ada observasi baru dengan $MB[h, e_2] = 0,2$ dan $MD[h, e_2] = 0$, maka :

$$MB[h, e_1 \wedge e_2] = 0,3 + 0,2 \cdot (1 - 0,3) = 0,44$$

$$MD[h, e_1 \wedge e_2] = 0$$

$$CF[h, e_1 \wedge e_2] = 0,44 - 0 = 0,44$$

Contoh X.2 :

Si Ani menderita bintik-bintik di wajahnya. Dokter memperkirakan Si Ani terkena cacar dengan kepercayaan,

$MB[\text{cacar, bintik-bintik}] = 0,8$ dan $MD[\text{cacar, bintik-bintik}] = 0,01$. Maka :

$$CF[\text{cacar, bintik-bintik}] = 0,8 - 0,01 = 0,79$$

Jika observasi tersebut juga memberikan kepercayaan bahwa Si Ani mungkin juga terkena alergi dengan kepercayaan, $MB[\text{alergi, bintik-bintik}] = 0,4$ dan $MD[\text{alergi, bintik-bintik}] = 0,3$; maka :

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

Untuk mencari $CF[\text{cacar} \wedge \text{alergi, bintik-bintik}]$ dapat diperoleh dari :

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

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

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

*image
not
available*

*image
not
available*

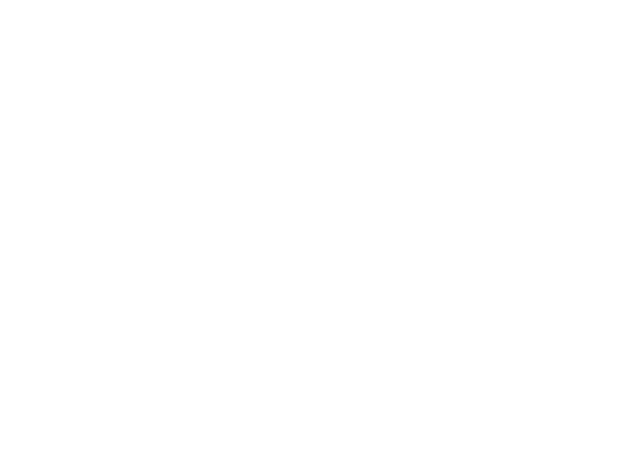
*image
not
available*

Latihan

1. Pahami kembali penalaran *Inexact*.
2. Buatlah contoh rancangan dengan menggunakan *certainty factor*.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.