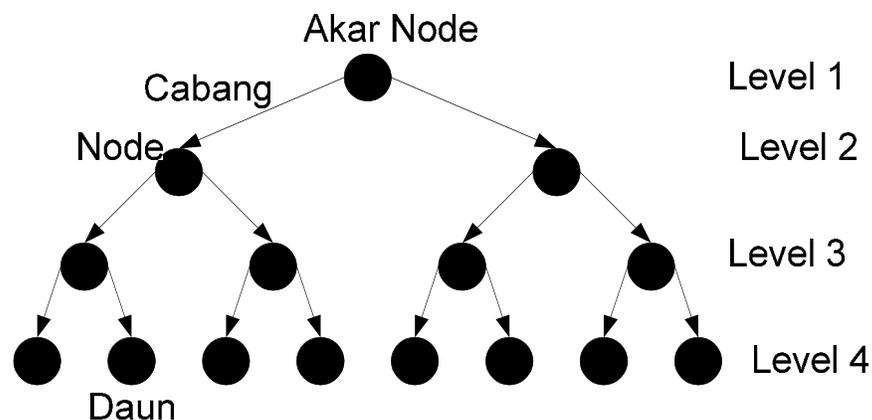


METODE INFERENSI

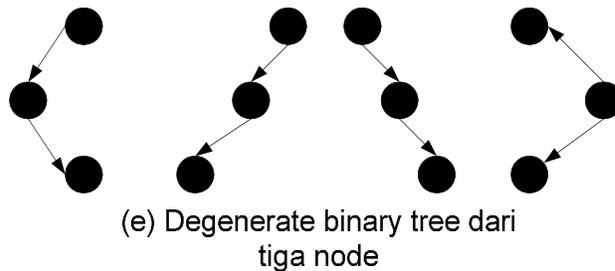
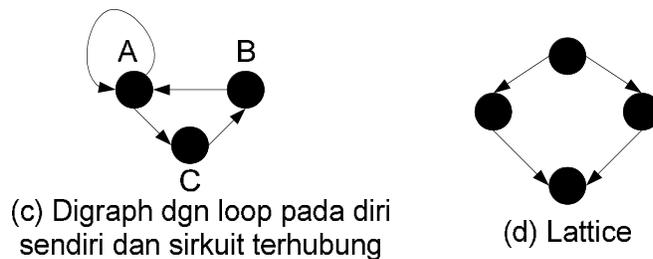
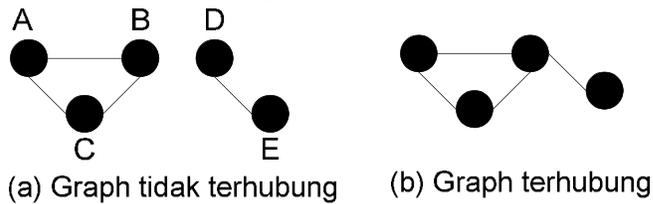
Tree (Pohon) dan Graph

- *Tree* (pohon) adalah suatu hierarki struktur yang terdiri dari *Node* (*simpul/veteks*) yang menyimpan informasi atau pengetahuan dan cabang (*link/edge*) yang menghubungkan node.
- Binary tree mempunyai 0,1 atau 2 cabang per-node.
 - o Node tertinggi disebut root
 - o Node terendah disebut daun

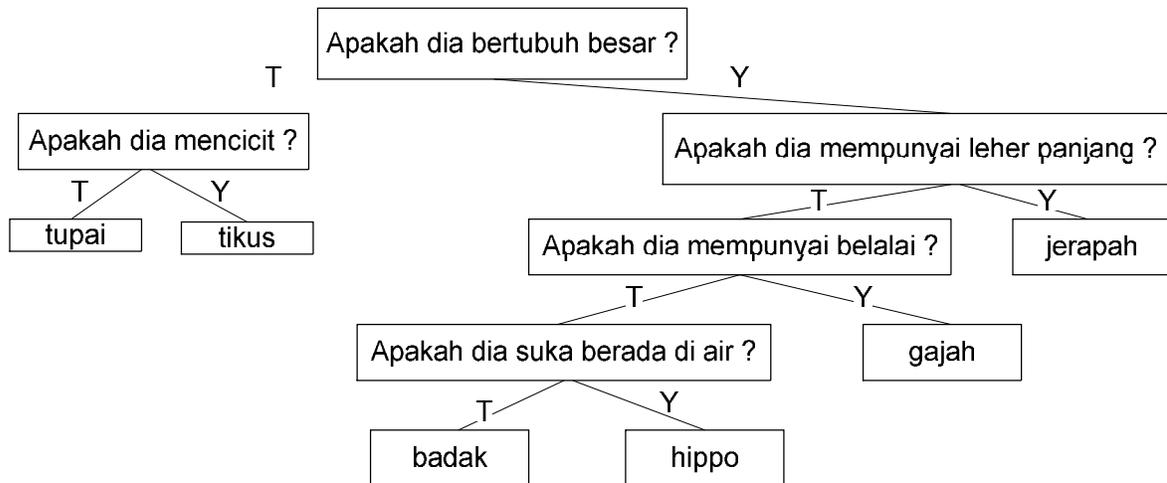


- Tree merupakan tipe khusus dari jaringan semantic, yang setiap nodenya kecuali akar, mempunyai satu node orang tua dan mempunyai nol atau lebih node anak.
- Tree adalah kasus khusus dalam Graph
- Graph dapat mempunyai nol atau lebih link di antara node dan tidak ada perbedaan antara orangtua dan anak.

- Dalam graph, link dapat ditunjukkan berupa panah atau arah yang memadukan node dan bobot yang merupakan karakteristik beberapa aspek dari link.
- Beberapa contoh graph sederhana:



- Graph *asiklik* adalah graph yang tidak mengandung siklus.
- Graph dengan link berarah disebut *digraph*.
- Graph asiklik berarah disebut *lattice*.
- Tree yang hanya dengan path tunggal dari akar untuk satu daun disebut *degenerate tree*.
- Aplikasi tree dan lattice adalah pembuatan keputusan disebut *decision tree* dan *decision lattice*.
- Contoh : decision tree yang menunjukkan pengetahuan tentang hewan.



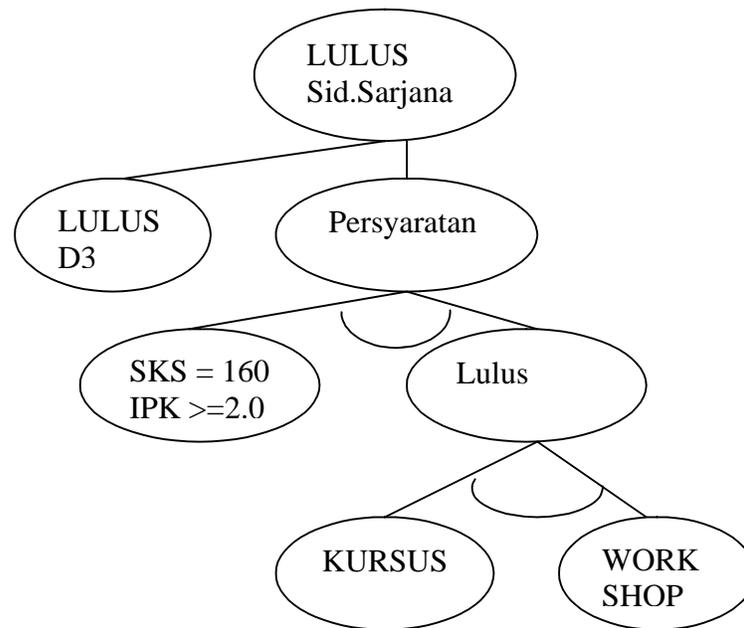
- Aturan produksi (IF...THEN...) dari contoh di atas :
 JIKA pertanyaan="Apakah dia bertubuh besar ?"
 DAN jawaban="Tidak"
 MAKA pertanyaan="Apakah dia menciit?"

JIKA pertanyaan="Apakah dia bertubuh besar ?"
 DAN jawaban="Ya"
 MAKA pertanyaan="Apakah dia mempunyai leher panjang?"
 dst.....

Pohon AND-OR dan Tujuan

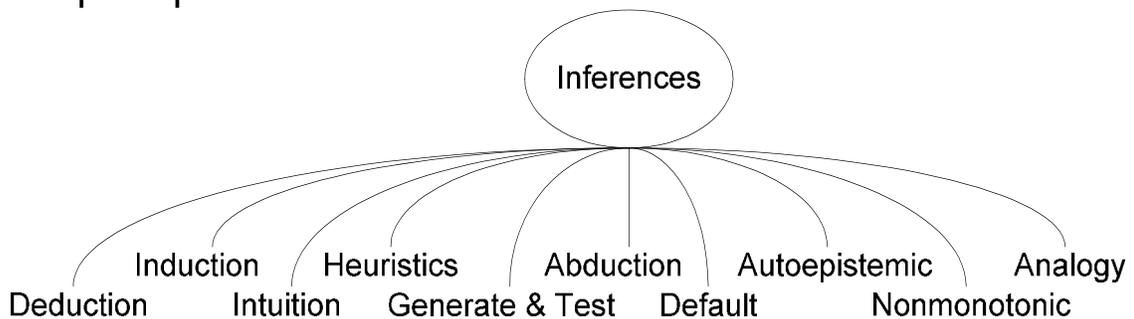
- Banyak tipe system pakar menggunakan *backward chaining* untuk mendapatkan solusi dari permasalahan.
- Salah satu tipe dari tree atau lattice yang digunakan dalam masalah representasi *backward chaining* adalah Pohon AND-OR.

- Contoh :



Penalaran Deduktif dan Silogisme

- Tipe-tipe Inferensi



Deduction

- Pemberian alasan logikal dimana kesimpulan harus mengikuti premis

Induction

- Inferensi dari khusus ke umum

Intuition

- Tidak ada teori yg menjamin. Jawabannya hanya muncul, mungkin dengan penentuan pola yg ada secara tidak disadari.

Heuristic

- Aturan yg didasarkan pada pengalaman

Generate & Test

- Trial dan error. Digunakan dgn perencanaan.

Abduction

- Pemberian alasan kembali dari kesimpulan yg benar ke premis .

Default

- Diasumsikan pengetahuan umum sebagai default

Autoepistemic

- Self-knowledge

Nonmonotonic

- Pengetahuan yg sebelumnya mungkin tdk benar jika bukti baru didapatkan

Analogy

- Kesimpulan yg berdasarkan pada persamaan untuk situasi yg lainnya.

- Suatu logika argument adalah kumpulan dari pernyataan-pernyataan yang dinyatakan untuk dibenarkan sebagai dasar dari rantai penalaran.

- Salah satu jenis logika argumen adalah *Silogisme*.

- Contoh :

Premis : Siapapun yang dapat membuat program adalah pintar

Premis : John dapat membuat program

Konklusi : Oleh karenanya John adalah pintar

Proses deduktif pada contoh di atas bergerak dari prinsip umum menuju konklusi khusus.

- Penalaran deduktif umumnya terdiri dari tiga bagian : *premis mayor*, *premis minor* dan *konklusi*.
- Premis disebut juga *antecedent*
- Konklusi/kesimpulan disebut juga *consequent*
- Silogisme dapat direpresentasikan ke dalam bentuk aturan JIKA.....MAKA..... (IF...THEN.....), contoh :
 JIKA siapapun yang dapat membuat program adalah pintar
 DAN John dapat membuat program
 MAKA John adalah pintar
- Silogisme klasik disebut *categoricall syllogism* (silogisme yang pasti)
- Premis dan konklusi didefinisikan sebagai statement yang pasti dari empat bentuk berikut :

Bentuk	Skema	Arti
A	Semua S adalah P	Universal Affirmative
E	Tidak S adalah P	Universal Negative
I	Beberapa S adalah P	Particular Affirmative
O	Beberapa S bukan P	Particular Negative

- Subjek dari konklusi S disebut bagian minor bila predikat konklusi P adalah bagian mayor.
- Premis terdiri dari premis mayor dan premis minor.
- Contoh :
 Premis mayor : Semua M adalah P
 Premis minor : Semua S adalah M
 Konklusi : Semua S adalah P
 Silogisme di atas adalah bentuk standar karena premis mayor dan minor sudah diketahui.

Contoh :

“Semua mikrokomputer adalah computer”

Subjeknya (objek yang digambarkan) adalah mikrokomputer.

Predikatnya (beberapa sifat subjek) adalah computer

- M (middle term) adalah hal yang penting karena silogisme didefinisikan sedemikian sehingga konklusi tidak dapat disimpulkan dengan mengambil salah satu premis.
- Q (quantifier) menggambarkan porsi dari kelas yang diketahui.
 - o Quantifier “semua” dan “tidak” adalah universal karena menunjukkan keseluruhan kelas.
 - o “beberapa” adalah khusus (particular) karena hanya menunjukkan satu bagian dari kelas yang diketahui.
- Mood dari silogisme didefinisikan sebagai tiga huruf yang memberikan bentuk masing-masing premis mayor, minor dan konklusi.

Contoh :

Semua M adalah P

Semua S adalah M

∴ Semua S adalah P

menunjukkan suatu mood AAA-1

- Ada 4 kemungkinan pola susunan istilah S, P dan M :

	Figure 1	Figure 2	Figure 3	Figure 4
Premis Mayor	MP	PM	MP	PM
Premis Minor	SM	SM	MS	MS

- Tidak selalu argument yang mempunyai bentuk silogisme merupakan silogisme yang valid.

- Contoh : Silogisme tidak valid berbentuk AEE-1

Semua M adalah P

Tidak S adalah M

∴ Tidak S adalah P

Semua mikrokomputer adalah computer

Bukan mainframe adalah mikrokomputer

∴ Bukan mainframe adalah computer

- Diperlukan prosedur keputusan (*decision procedure*) untuk pembuktian validitas.

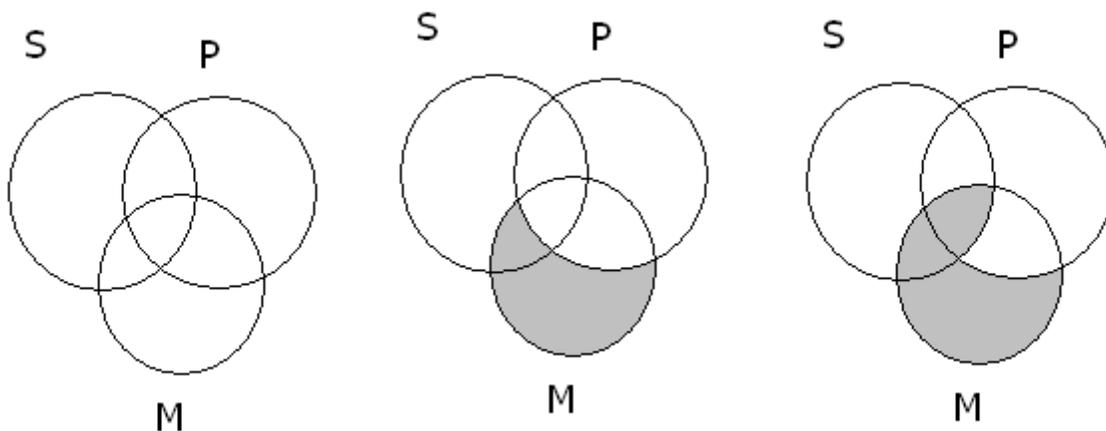
- Prosedur keputusan untuk silogisme dapat dilakukan menggunakan diagram venn tiga lingkaran yang saling berpotongan yang merepresentasikan S,P, M.

- Contoh : Prosedur Keputusan untuk AEE-1

Semua M adalah P

Tidak S adalah M

∴ Tidak S adalah P

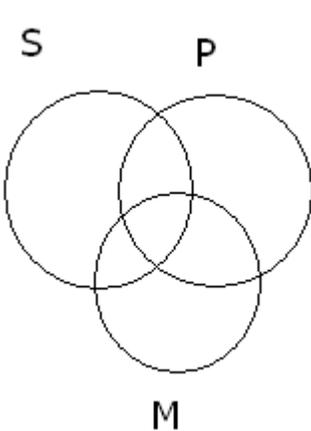


a. Diagram Venn

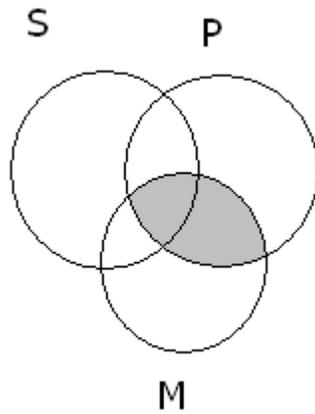
b. Setelah Premis Mayor

c. Setelah Premis Minor

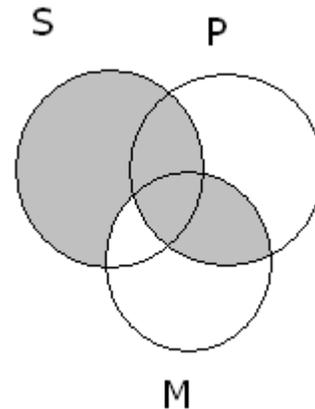
- Contoh : Prosedur Keputusan untuk EAE-1
 Tidak M adalah P
Semua S adalah M
 \therefore Tidak S adalah P



a. Diagram Venn



b. Setelah Premis Mayor



c. Setelah Premis Minor

Kaidah dari Inferensi

- Diagram Venn tidak sesuai untuk argumen yang lebih kompleks karena sulit dibaca pada decision tree untuk silogisme.
- Logika proposisi memberikan pengertian lain dari penggambaran argumen.
- Contoh :

Jika ada daya listrik, komputer akan bekerja

Ada daya

\therefore Komputer akan bekerja

A = ada daya listrik

B = komputer akan bekerja

Sehingga dapat ditulis :

$$\frac{A \rightarrow B}{A} \therefore B$$

- Bentuk umum *Ponens* / direct reasoning / law of detachment / assuming the antecedent

$$\frac{p \rightarrow q}{p} \therefore q \quad \text{atau} \quad p \rightarrow q, p; \therefore q$$

Bentuk tersebut valid, karena argumen tersebut dapat ditunjukkan sebagai suatu *tautologi*.

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

Tabel Kebenaran Ponens :

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

- Terdapat argumen yang menyerupai ponens namun perlu dibuktikan validitasnya.

Contoh :

Jika tidak kesalahan maka program dapat mengkompile

Program dapat mengkompile

\therefore Tidak ada kesalahan

$$\frac{p \rightarrow q}{q} \therefore p \quad \text{atau} \quad p \rightarrow q, q; \therefore p$$

Tabel Kebenaran:

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

(Bukan Pones karena tidak bersifat Tautology)

- Skema argumen lain :

$$\begin{array}{l} p \rightarrow q \\ \sim q \\ \hline \therefore \sim p \end{array}$$

Tabel Kebenaran:

p	q	$p \rightarrow q$	$\sim q$	$(p \rightarrow q) \wedge \sim q$	$\sim p$	$((p \rightarrow q) \wedge \sim q) \rightarrow \sim p$
T	T	T	F	F	F	T
T	F	F	T	F	F	T
F	T	T	F	F	T	T
F	F	T	T	T	T	T

Argumen di atas disebut *Tollens* / indirect reasoning / law of contraposition.

- Beberapa hukum Inferensi

Hukum Inferensi	Skema
1. Hukum Detasemen	$\begin{array}{l} p \rightarrow q \\ p \\ \hline \therefore q \end{array}$
2. Hukum Kontrapositif	$\begin{array}{l} p \rightarrow q \\ \hline \therefore \sim q \rightarrow \sim p \end{array}$
3. Hukum Modus Tollens	$\begin{array}{l} p \rightarrow q \\ \sim q \\ \hline \therefore \sim p \end{array}$

4. Aturan Rantai (hukum silogisme)	$p \rightarrow q$ $q \rightarrow r$ $\therefore p \rightarrow r$
5. Hukum Inferensi Disjungsi	$p \vee q$ $p \vee q$ $\sim p$ $\sim q$ $\therefore q$ $\therefore p$
6. Hukum negasi	$\sim(\sim p)$ $\therefore p$
7. Hukum de Morgan	$\sim(p \wedge q)$ $\sim(p \vee q)$ $\therefore \sim p \vee \sim q$ $\therefore \sim p \wedge q$
8. Hukum Simplifikasi	$p \wedge q$ $p \wedge q$ $\therefore p$ $\therefore q$
9. Hukum Konjungsi	p q $\therefore p \wedge q$
10. Hukum Penambahan Disjungtif	p $\therefore p \vee q$
11. Hukum Argumen Konjugtif	$\sim(p \wedge q)$ $\sim(p \wedge q)$ p q $\sim q$ $\sim p$

- Kaidah inferensi dapat digunakan untuk argumen yang mempunyai lebih dari dua premis.

Contoh :

Harga chip naik hanya jika yen naik
 Yen naik hanya jika dollar turun dan
 jika dollar turun maka yen naik
 Karena harga chip telah naik
 \therefore Dollar harus turun

Misal : C = harga chip naik
 Y = Yen naik
 D = Dollar turun

1. $C \rightarrow Y$
 2. $(Y \rightarrow D) \wedge (D \rightarrow Y)$
 3. C
-
- $\therefore D$

- Kondisional $p \rightarrow q$ mempunyai converse, inverse dan kontrapositif

Kondisional	$p \rightarrow q$
Converse	$q \rightarrow p$
Inverse	$\sim p \rightarrow \sim q$
Kontrapositif	$\sim q \rightarrow \sim p$

Jika $p \rightarrow q$ dan $q \rightarrow p$ bernilai benar, maka keduanya adalah ekuivalen.

$p \rightarrow q \wedge q \rightarrow p$ ekuivalen dengan $p \leftrightarrow q$ atau $p \equiv q$.

sehingga argumen untuk contoh di atas, menjadi :

- | | |
|---|------------------|
| 1. $C \rightarrow Y$ | |
| 2. $(Y \rightarrow D) \wedge (D \rightarrow Y)$ | |
| 3. C | $\therefore D$ |
| 4. $Y \equiv D$ | 2 ekuivalen |
| 5. $C \rightarrow D$ | 1 substitusi |
| 6. D | 3,5 modus ponens |