**METODA INFERENSI**

**5**

JUMLAH PERTEMUAN : 1 PERTEMUAN

TUJUAN INSTRUKSIONAL KHUSUS :

1. Memahami fungsi dan manfaat inferensi dalam kecerdasan buatan
2. Mengenal metoda-metoda yang digunakan dalam inferensi
3. Menerapkan metoda-metoda inferensi dalam menyelesaikan masalah

**Materi :**

* 1. **Pengenalan**

**inferensi** merupakan suatu proses untuk menghasilkan informasi  dari  fakta  yang diketahui.  Inferensi  adalah  konklusi  logis  atau  implikasi berdasarkan informasi yang tersedia. Dalam sistem pakar,  proses inferensi dialakukan dalam suatu modul yang disebut inference  engine. Ketika representasi pengetahaun pada bagian knowledge base  telah lengkap, atau paling tidak telah berada pada level yang cukup  akurat, maka representasi pengetahuan tersebut telah siap digunakan.

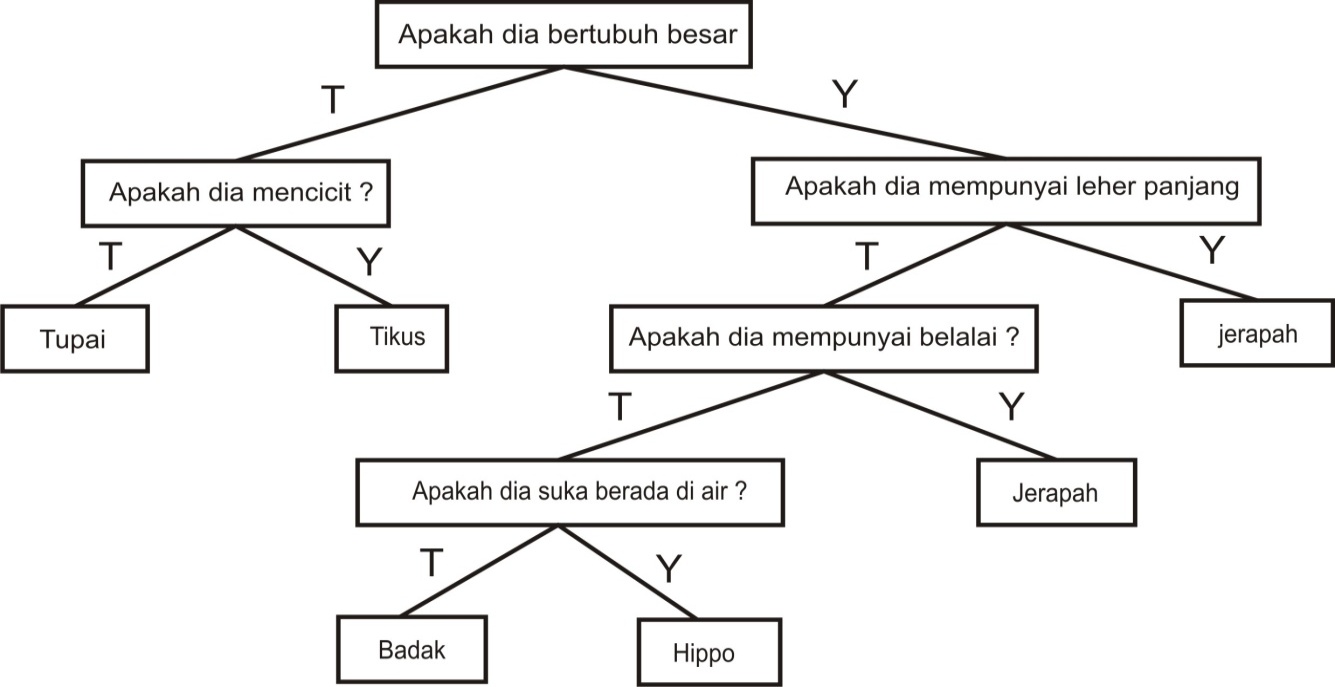
**Inference  engine**  merupakan  modul  yang  berisi  program  tentang  bagaimana mengendalikan proses reasoning.  Ada dua metode inferensi yang penting dalam sistem pakar,  yaitu  runut  maju  (forward  chaining)  dan  runut  balik  (backward  chaining).

* 1. **Tree, Latice dan Graph**

Tree adalah hierarki struktur yang terdiri node atau simpul yang menyimpan informasi atau pengetahuan serta cabang atau edge yang menghubungkan node. Tree merupakan tipe khusus jaringan semantik yang setiap nodenya, kecuali akar, mempunyai satu node orang tua dan mempunyai nol atau lebih node anak. Tree adalah bagian dari Graph yang memiliki ciri khusus yaitu tidak memiliki edge yang terhubung ke node yang tingkatannya lebih tinggi. Berikut ini beberapa istilah dalam bagian ini, yaitu :

* Graph asiklik adalah graph yang tidak mengandung siklus.
* Graph berarah disebut digraph
* Graph asiklik yang berarah adalah lattice
* Tree yang memiliki satu path untuk satu node disebut degerate tree

Tree dan Lattice diaplikasikan dalam decision tree dan decision lattice. Contoh : decision tree yang menunjukkan pengetahuan tentang hewan



Gambar 5.1 Decision tree pengetahuan tentang hewan

Berdasarkan Gambar 5.1 aturan produksi adalah sebagai berikut :

JIKA pertanyaan = “Apakah dia bertubuh besar ?”

DAN jawaban = “Tidak”

MAKA pertanyaan = “Apakah dia mencicit ?”

JIKA pertanyaan = “Apakah dia bertubuh besar ?”

DAN jawaban = “Ya”

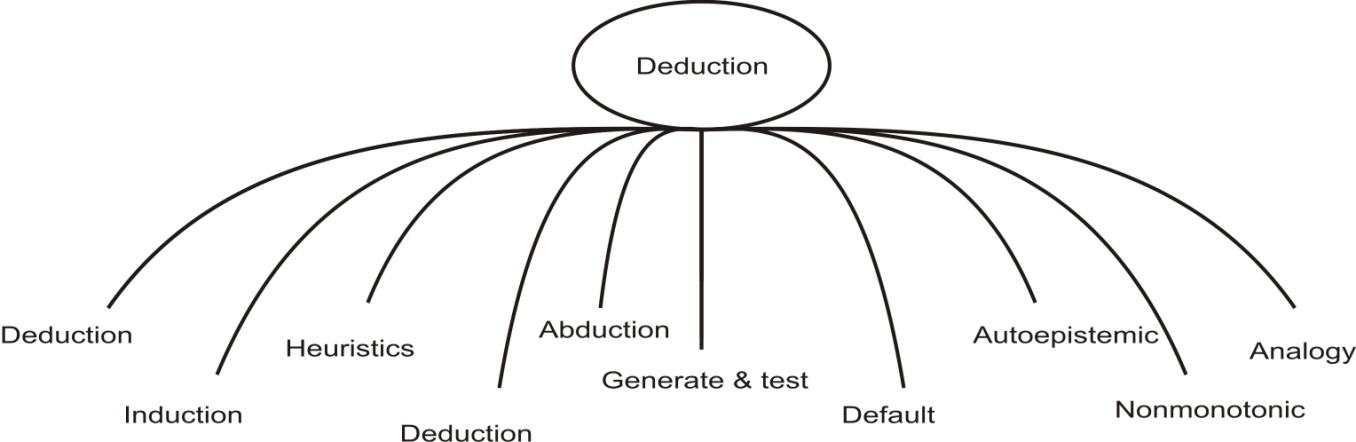
MAKA pertanyaan = “Apakah dia mempunyai leher panjang?”

Dst…

* 1. **Tipe-tipe Inferensi**

1. Deduction

Penarikan kesimpulan dari keadaan yang umum atau penemuan yang khusus dari yang umum. Metode deduksi akan membuktikan suatu kebenaran baru berasal dari kebenaran-kebenaran yang sudah ada dan diketahui sebelumnya (berkesinambungan ). Cirinya adalah jika sebuah pernyataan yang dianggap mewakili sebuah kebenaran atau setidaknya sesuatu yang dianggap benar yang memiliki implikasi-implikasi tertentu yang dapat diturunkan menjadi sebuah atau beberapa buah pernyataan yang lebih spesifik dan khusus. Pemberian alasan dalam logika dimana kesimpulan harus mengikuti premis.



Gambar 5.2 tipe-tipe inferensi

1. Induction

Salah bentuk inferensi dari khusus ke umum.

1. Intuition

Tipe inferensi yang terjadi tanpa dilandasi oleh teori. Kesimpulan muncul karena pola yang ada secara tidak disadari.

1. Heuristic

Kesimpulan yang diambil berdasarkan pengalaman

1. Generate & Test

Kesimpulan ditentukan dengan trial dan error. Solusi dibuat kemudian diuji untuk melihat apakah solusi yg diajukan memenuhi semua persyaratan. Jika solusi memenuhi maka berhenti yang lain membuat solusi yang baru kemudian test lagi dst.

1. Abduction

Dari kesimpulan yang benar ke premis. Metodenya mirip dengan modus ponens

|  |  |  |
| --- | --- | --- |
| Abduction |  | Modus ponens |
| p 🡪 q |  | p 🡪 q |
| q |  | p |
| p |  | q |

* Bukan argument deduksi yang valid
* Berguna untuk kaidah inferensi heuristik
* Analogi,generate and test, abduction adalah metode bukan deduksi. Dari premise yg benar, metode ini tidak dapat membuktikan kesimpulan yg benar

1. Default

Diasumsikan pengetahuan umum sebagai default

1. Autoepistemic

Logika autoepistemic adalah logika formal untuk representasi dan reasoning pengetahuan tentang pengetahuan. Saat logika proposisi hanya dapat mengekspresikan fakta, autoepistemic dapat mengekspresikan pengetahuan dan kebocoran pengetahuan tentang fakta.

1. Nonmonotonic

* Pengetahuan yang sebelumnya mungkin tidak benar jika bukti baru didapatkan. Adanya tambahan aksioma baru pada sistem logika berarti akan banyak teorema yang dapat dibuktikan.
* Peningkatan teorema dengan peningkatan aksioma dikenal dengan ***sistem monotonik***
* Suatu masalah dapat terjadi, jika diperkenalkan aksioma parsial atau komplit baru yang kontradikasi dengan aksioma sebelumnya.
* Pada **sistem nonmonotonik**, tidak perlu adanya peningkatan teorema yang sejalan dengan peningkatan aksioma.

1. Analogy

* Kesimpulan yang berdasarkan pada persamaan untuk situasi yang lain. Mencoba dan menghubungkan situasi lama sebagai penuntun ke situasi baru.
* Contoh : diagnosis medical (gejala penyakit yang diderita oleh seorang pasien ternyata sama dengan gejala yang dialami pasien lain).
* Pemberian alasan analogis berhubungan dgn induksi. Bila induksi membuat inferensi dari spesifik ke umum pada situasi yang sama, maka analogy membuat inferensi dari situasi yang tidak sama.
  1. **Silogisme**

Silogisme adalah salah satu jenis logika argumen. Logika argumen adalah kumpulan pernyataan-pernyataan yang dinyatakan untuk dibenarkan sebagai dasar dari rantai penalaran.

Contoh :

Premis : siapapun yang dapat membuat program adalah pintar.

Premis : John dapat membuat program

Konklusi : oleh karenanya John adalah pintar.

Proses deduksi untuk silogisme diatas bergerak dari prinsip umum menuju konklusi khusus. Penalaran deduktif umumnya terdiri dari tiga bagian, yaitu : premis mayor, premis minor dan konklusi. Premis disebut juga antecedent dan konklusi disebut consequent.

Silogisme dapat direpresentasikan ke dalam bentuk aturan IF… THEN…

Contoh :

IF siapapun yang dapat membuat program adalah pintar

AND 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 Afirmative |
| E | Tidak S adalah P | Universal Negative |
| I | Beberapa S adalah P | Particular Afirmative |
| 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

M (middle term) adalah hal yang penting karena silogisme didefinisikan sedemikian sehingga konklusi tidak dapat disimpulkan dengan mengambil salah satu premis.

Mood dari silogisme didefinisikan sebagai tiga huruf yang memberikan bentuk masing-masing premis mayor, minor dan konklusi.

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 |

Contoh :

|  |
| --- |
| Semua M adalah P |
| Semua S adalah M |
| Semua S adalah P |

Menunjukkan suatu mood AAA-1

Argumen yang mempunyai bentuk silogisme tidak selalu valid.

Contoh :

|  |
| --- |
| Semua M adalah P |
| Tidak S adalah M |
| Tidak S adalah P |

Untuk membuktikan validitas suatu argumen diperlukan prosedur keputusan. Prosedur keputusan untuk silogisme dapat dilakukan menggunakan diagram venn tiga lingkaran yang saling berpotongan yang merepresentasikan S,P dan M.

Contoh : Prosedur Keputusan untuk AEE – 1

|  |
| --- |
| Semua M adalah P |
| Tidak S adalah M |
| Tidak S adalah P |
|  |
| A.png | | | B.png | C.png |
| Diagram Venn | | | Setelah Premis Mayor | Setelah Premis Minor |

Contoh : Prosedur keputusan untuk EAE – 1

|  |
| --- |
| Tidak M adalah P |
| Semua S adalah M |
| Tidak S adalah P |

|  |  |  |
| --- | --- | --- |
| D.png | E.png | F.png |
| Diagram Venn | Setelah Premis Mayor | Setelah Premis Minor |

* 1. **Kaidah Inferensi**

Representasi logika dengan diagram venn tidak sesuai untuk argumen yang lebih kompleks karena sulit dibaca pada decision tree untuk silogisme. Jika hal tersebut terjadi, logika proposisi memberi pengertian lain dari penggambaran argumen.

Contoh :

|  |
| --- |
| Jika ada daya listrik, komputer akan bekerja |
| Ada daya |
| komputer akan bekerja |

Jika : A = ada daya listrik

B = komputer akan bekerja

Sehingga dapat ditulis :

|  |
| --- |
| A→B |
| A |
| B |

**MODUS PONEN**

Nama lain Ponen adalah direct reasoning atau law of detachment atau assuming the antecedent.

|  |  |
| --- | --- |
| p→q |  |
| P | Atau p→q , p ; q |
| q |  |

Bentuk ini valid karena argumen tersebut dapat ditunjukkan sebagai tautologi.

((p→q) p) q

Buatlah tabel kebenarannya.

**MODUS TOLLENS**

Tollens atau disebut juga indirect reasoning atau law of contraposition.

|  |
| --- |
| p→q |
| q |
| p |

Buatlah tabel kebenarannya

Bentuk-bentuk lain inferensi seperti pada tabel berikut :

|  |  |  |
| --- | --- | --- |
| **HUKUM INFERENSI** | **SKEMA** | |
| 1. Hukum Detasemen | p q  p  q | |
| 1. Hukum Kontraposisi | p q    qp | |
| 1. Hukum Modus Tollens | p q  q  p | |
| 1. Aturan Rantai (silogisme) | p q  q r  pr | |
| 1. Hukum Disjungsi | pq  p  q | pq  q  p |
| 1. Hukum Negasi Ganda | ~(~p)  p | |
| 1. Hukum De Morgan | ~ (p q)  p~q | ~pq  ~p ~ q |
| 1. Hukum Simplifikasi | p q  p | p q  q |
| 1. Hukum Konjungsi | p  q  p q | |
| 1. Hukum penambahan Disjungtif | p  pq | |
| 1. Hukum Argumen Konjungtif | ~(p q)  p  ~q | ~(p q)  q  ~p |

* 1. **Logika Urutan Pertama (First Order Logic)**

Representasi 4 kategori silogisme menggunakan logika predikat

|  |  |  |
| --- | --- | --- |
| Bentuk | Skema | Representasi Predikat |
| A | Semua S adalah P | (∀x) (S(x)🡪P(x)) |
| E | Tidak S adalah P | (∀x) (S(x)🡪~P(x)) |
| I | Beberapa S adalah P | (∃x) (S(x)🡪P(x)) |
| O | Beberapa S bukan P | (∃x) (S(x)🡪~P(x)) |

Kaidah Universal Instatiation merupakan state dasar, dimana suatu individual dapat digantikan (disubsitusi) ke dalam sifat universal.

Contoh :

Misal, φ merupakan fungsi proposisi :

(∀x) φ(x)

∴ φ(a)

merupakan bentuk yang valid, dimana a menunjukkan spesifik individual, sedangkan x adalah suatu variabel yang berada dalam jangkauan semua individu (universal)

Contoh lain : (∀x) H(x)

∴ H(Socrates)

Berikut ini adalah contoh pembuktian formal silogisme:

All men are mortal

Socrates is a man

Therefore, Socrates is mortal

Misal : H = man, M = mortal, s = Socrates

1. (∀x) (H (x) 🡪 M(x))
2. H(s) / ∴ M(s)
3. H(s) 🡪 M(s) 1 Universal Instatiation
4. M(s) 2,3 Modus Ponens

**5.7 SISTEM LOGIKA**

Sistem logika adalah kumpulan objek seperti kaidah (rule), aksioma, statement dan lainnya yang diatur dalam cara yang konsisten.

Sistem logika mempunyai beberapa tujuan :

1. Menentukan bentuk argumen.

Awalnya argumen logika tidak memiliki arti dalam semantic sense, bentuk yang valid pada dasarnya dapat dicapai jika validitas dari argumen tersebut dapat ditentukan.

Fungsi terpenting dari logika sistem adalah menentukan **well formed formulas (wffs)** dari argumen yang digunakan.

Contoh : All S is P ….. merupakan wffs

tapi…. All

All is S P ….. bukan wffs

Is S all

1. Menunjukkan kaidah inferensi yang valid.
2. Mengembangkan dirinya sendiri dengan menemukan kaidah baru inferensi dan memperluas jangkauan argumen yang dapat dibuktikan.

Sistem logika dibangun melalui Sentential atau kalkulus proposisi, kalkulus predikat dst.

Setiap sistem disandarkan pada **aksioma** atau **postulat**, yang merupakan definisi mendasar dari sistem.

Suatu **aksioma** merupakan fakta sederhana atau assertion yang tidak dapat dibuktikan dalam sistem. Terkadang, kita menerima aksioma dikarenakan ada sesuatu yang menarik atau melalui pengamatan.

Sistem formal membutuhkan :

1. simbol alfabet.
2. suatu set finite string dari simbol tertentu, wffs
3. aksioma, definisi dari sistem
4. kaidah inferensi, yang memungkinkan wffs, A untuk dikurangi sebagai kesimpulan dari set finite Γ wff lain dimana Γ = {A1,A2,…An}. Wffs harus berupa aksioma atau teori lain dari sistem logis. Sebagai contoh : sistem logika dapat didefinisikan menggunakan modus pones untuk diturunkan menjadi teorema baru.

Jika terdapat argumen :

A1, A2, ……., AN; ∴ A

yang valid, maka A disebut teorema dari sistem logika formal dan ditulis dengan simbol  (metasymbol) yang menunjukkan wff adalah suatu **teorema** .

A1, A2, ……., AN A

Contoh : teorema silogisme tentang Socrates yang

ditulis dalam bentuk logika predikat.

(∀x) (H (x)🡪M(x)), H(s) M(s)

M(s) dapat dibuktikan dari aksioma di sisi kiri, hal tersebut menunjukkan aksioma

Suatu teorema merupakan **tautology**, ditunjukkan melalui Γ sebagai set null dimana wff selalu bernilai null dan tidak tergantung dari aksioma atau teorema yang lain.

Teorema dengan tautology ditulis dengan simbol , misalnya A.

Contoh :

Jika A ≡ p ∨ ~p maka p ∨ ~p

Suatu model adalah interpretasi wff bernilai benar.

Suatu wff disebut **konsisten** atau **satifiable** jika interpretasi yang dihasilkan benar, dan disebut **inkonsisten** atau **unsatisfiable** jika wff menghasilkan nilai yang salah pada semua interpretasi.

**5.8 RESOLUSI**

Diperkenalkan oleh Robinson (1965). Resolusi merupakan kaidah inferensi utama dalam bahasa PROLOG. PROLOG menggunakan notasi “quantifier-free”. PROLOG didasarakan pada logika predikat urutan pertama.

Sebelum resolusi diaplikasikan, wff harus berada dalam **bentuk normal** atau **standard**.

Tiga tipe utama bentuk normal : **conjunctive normal form, clausal form** dan **subset Horn clause**. Resolusi diaplikasikan ke dalam bentuk normal wff dengan menghubungkan seluruh elemen dan quantifier yang dieliminasi.

Contoh :

(A ∨ B) ∧ (~B ∨C) ………… conjunctive normal form

Dimana A ∨ B dan ~B ∨C adalah clause.

Logika proposional dapat ditulis dalam bentuk clause.

**Full clause form** yang mengekspresikan formula logika predikat dapat ditulis dalam **Kowalski clause form**.

A1, A2, ……., AN 🡪 B1, B2, ……., BM

Clause yang ditulis dalam notasi standard :

A1∧ A2, ……., AN 🡪 B1 ∨ B2, ……., BM

Bentuk disjungsinya merupakan disjungsi dari literal menggunakan equivalence :

p 🡪 q ≡ ~p ∨ q

sehingga

A1∧ A2, ……., AN 🡪 B1 ∨ B2, ……., BM

≡ ~( A1∧ A2, …, AN) ∨ (B1 ∨ B2, …., BM)

≡ ~A1∨ ~A2, …, ~AN ∨ B1 ∨ B2, …., BM

Yang merupakan hukum de Morgan : ~(p ∧ q) ≡ ~p ∨ ~q

Dengan Horn clause dapat ditulis : A1, A2, ……., AN 🡪 B

Dalam bahasa PROLOG ditulis : B :- A1, A2, ……., AN

Untuk membuktikan teorema di atas benar, digunakan metode klasik **reductio ad absurdum** atau **metode kontradiksi**.

Tujuan dasar resolusi adalah membuat infer klausa baru yang disebut “**revolvent**” dari dua klausa lain yang disebut **parent clause**.

Contoh :

A ∨ B

A ∨ ~B

∴ A

Premis dapat ditulis : (A ∨ B) ∧ (A ∨ ~B)

Ingat Aksioma Distribusi : p ∨ (q ∧ r) ≡ (p ∨ q) ∧ (p ∨ q)

Sehingga premis di atas dapat ditulis : (A ∨ B) ∧ (A ∨ ~B) ≡ A ∨ (B ∧ ~B) ≡ A

dimana B ∧ ~B selalu bernilai salah.

Tabel Klausa dan Resolvent

|  |  |  |
| --- | --- | --- |
| **Parent Clause** | **Resolvent** | **Arti** |
| p 🡪 q , p atau ~p ∨ q, p | Q | Modus Pones |
| p 🡪 q , q 🡪 r atau  ~p ∨ q, ~ q ∨ r | p 🡪 r atau  ~p ∨ r | Chaining atau Silogisme Hipotesis |
| ~p ∨ q, p ∨ q | Q | Penggabungan |
| ~p ∨ ~q, p ∨ q | ~p ∨ p atau ~q ∨ q | TRUE (tautology) |
| ~p, p | Nill | FALSE (kontradiksi) |

**5.9 SISTEM RESOLUSI DAN DEDUKSI**

**Refutation** adalah pembuktian teorema dengan menunjukkan negasi atau pembuktian kontradiksi melalui reductio ad absurdum. Melakukan refute berarti membuktikan kesalahan.

Contoh :

A 🡪 B

B 🡪 C

C 🡪 D

∴A 🡪 D

Untuk membuktikan konklusi A 🡪 D adalah suatu teorema melalui resolusi refutation, hal yang dilakukan : p 🡪 q ≡ ~p ∨ q

sehingga A🡪 D ≡ ~A ∨ D

dan langkah terakhir adalah melakukan negasi ~(~A ∨ D) ≡ A ∧ ~D

Penggunaan konjungsi dari disjunctive form pada premis dan negasi pada konsklusi, memberikan conjuctive normal form yang cocok untuk resolusi refutation.

Dari contoh di atas, penulisannya menjadi :

(~A ∨ B) ∧ (~B ∨ C) ∧ (~C ∨ D) ∧ A ∧ ~D



Akar bernilai nill, menunjukkan kontradiksi. Sehingga melalui refutation dapat ditunjukkan konklusi asli (awal) adalah **teorema** dengan peran kontradiksi.

* 1. **Forward Chaining**

Chain (rantai) : perkalian inferensi yang menghubung-kan suatu permasalahan dengan solusinya.

Forward chaining :

* Suatu rantai yang dicari atau dilewati/dilintasi dari suatu permasalahn untuk memperoleh solusi.
* Penalaran dari fakta menuju konklusi yang terdapat dari fakta.

Backward chaining :

* Suatu rantai yang dilintasi dari suatu hipotesa kembali ke fakta yang mendukung hipotesa tersebut.
* Tujuan yang dapat dipenuhi dengan pemenuhan sub tujuannya.

Contoh rantai inferensi :

gajah(x) 🡪 mamalia (x)

mamalia(x) 🡪 binatang(x)

* Causal (sebab-akibat) Forward chain

gajah(clyde)

gajah(x) mamalia(x)

mamalia(x) binatang(x)

binatang(clyde)

* Explicit Causal chain

gajah(clyde)

unifikasi

implikasi gajah(clyde) mamalia(clyde)

unifikasi

implikasi mamalia(clyde)

Karakteristik Forward dan Backward chaining

|  |  |
| --- | --- |
| **Forward chaining** | **Backward chaining** |
| Perencanaan, monitoring, kontrol | Diagnosis |
| Disajkan untuk masa depan | Disajikan untuk masa lalu |
| Antecedent ke konsekuen | Konsekuen ke antecedent |
| Data memandu, penalaran dari bawah ke atas | Tujuan memandu, penalaran dari atas ke bawah |
| Bekerja ke depan untuk mendapatkan solusi apa yang mengikuti fakta | Bekerja ke belakang untuk mendapatkan fakta yang mendukung hipotesis |
| *Breadth first search* dimudahkan | *Depth first search* dimudahkan |
| *Antecedent* menentukan pencarian | *Konsekuen* menentukan pencarian |
| Penjelasan tidak difasilitasi | Penjelasan difasilitasi |

* Forward Chaining





* Backward Chaining





**LATIHAN**

1. Cari jurnal ilmiah dengan topik kecerdasan buatan dengan menggunakan metoda inferensi. Buat resume jurnal tersebut.