

Pertemuan 2

Representasi Pengetahuan dan Penalaran

(Knowledge Representation & Reasoning)

A. Konsep Pengetahuan Eksplisit dan Implisit

1. Pengantar

Dalam Kecerdasan Buatan, pengetahuan (*knowledge*) merupakan inti utama dari kecerdasan itu sendiri. Sebuah sistem AI tidak akan mampu berpikir, menalar, atau mengambil keputusan tanpa memiliki pengetahuan yang terstruktur dan dapat diakses oleh mesin. Representasi pengetahuan adalah proses mengubah pengetahuan manusia ke dalam bentuk yang dapat dipahami dan diolah oleh komputer.

2. Pengertian Pengetahuan (*Knowledge*)

Pengetahuan dapat didefinisikan sebagai informasi yang telah diproses dan diorganisasi sedemikian rupa sehingga dapat digunakan untuk memahami, menalar, dan membuat keputusan.

Dalam konteks AI:

- Data → fakta mentah tanpa konteks.
- Informasi → data yang telah diproses sehingga bermakna.
- Pengetahuan (Knowledge) → informasi yang diinterpretasikan untuk membuat keputusan atau menyelesaikan masalah.

Contoh:

| Level | Contoh | Keterangan |
|-------------|--|---|
| Data | “Suhu = 38°C” | Fakta mentah |
| Informasi | “Tubuh pasien demam” | Hasil interpretasi data |
| Pengetahuan | “Jika pasien demam >38°C, mungkin terkena infeksi” | Aturan yang dapat digunakan untuk penalaran |



3. Jenis Pengetahuan dalam AI

Pengetahuan yang digunakan dalam sistem cerdas dibedakan menjadi dua kategori utama:

A. Pengetahuan Eksplisit (*Explicit Knowledge*)

Pengetahuan eksplisit adalah pengetahuan yang terdokumentasi, tersusun secara formal, dan mudah dikomunikasikan atau dikodekan ke dalam sistem.

Ciri-ciri Pengetahuan Eksplisit:

- Dapat ditulis, disimpan, dan ditransfer dengan mudah.
- Dinyatakan dalam bentuk aturan, fakta, tabel, diagram, atau kode program.
- Dapat langsung digunakan oleh mesin inferensi dalam sistem pakar.
- Bersifat terstruktur dan logis.

Contoh Pengetahuan Eksplisit:

1. Aturan IF–THEN dalam sistem pakar:
2. IF suhu tubuh > 38°C THEN kondisi = "demam"
3. Database gejala penyakit:

| Gejala | Penyakit |
|---------------|----------|
| Batuk + Demam | ISPA |

4. Manual Prosedur Operasional (SOP): langkah-langkah kerja yang terdokumentasi.

Dalam AI, pengetahuan eksplisit direpresentasikan melalui:

- *Rule-based system* (aturan logika)
- *Semantic networks*
- *Frames*
- *Ontologi*
- *Predicate logic (logika predikat)*



B. Pengetahuan Implisit (*Tacit / Implicit Knowledge*)

Pengetahuan implisit (atau tacit) adalah pengetahuan yang tersimpan dalam pengalaman, intuisi, atau wawasan seseorang, namun sulit untuk dijelaskan atau diformalkan secara langsung.

Ciri-ciri Pengetahuan Implisit:

- Bersumber dari pengalaman dan intuisi manusia.
- Sulit diekspresikan dalam bentuk aturan formal.
- Biasanya diperoleh melalui praktik, observasi, dan pembelajaran tidak langsung.
- Bersifat subjektif dan kontekstual.

Contoh Pengetahuan Implisit:

1. Dokter berpengalaman dapat mengenali penyakit hanya dari ekspresi wajah pasien — tanpa aturan tertulis.
2. Seorang sopir berpengalaman “merasakan” kapan harus mengerem, bukan berdasarkan perhitungan jarak.
3. Guru dapat “mengetahui” siswa mana yang mengalami kesulitan belajar hanya dengan mengamati perilakunya.

Dalam AI, pengetahuan implisit biasanya diekstraksi melalui:

- *Machine Learning* (AI belajar pola dari data secara otomatis)
- *Neural Networks* (AI belajar hubungan kompleks tanpa aturan eksplisit)
- *Case-Based Reasoning* (AI belajar dari pengalaman kasus sebelumnya)

4. Perbandingan Pengetahuan Eksplisit dan Implisit

| Aspek | Pengetahuan Eksplisit | Pengetahuan Implisit |
|-------------------|---|------------------------------------|
| Sifat | Terbuka, terdokumentasi | Tersembunyi, intuitif |
| Bentuk | Aturan, fakta, prosedur | Pengalaman, persepsi, intuisi |
| Cara Diperoleh | Melalui pembelajaran formal / dokumentasi | Melalui pengalaman langsung |
| Dapat Ditransfer? | Mudah (bisa ditulis, dibaca) | Sulit (butuh pelatihan, mentoring) |



| | | |
|------------------------|-----------------------------------|--|
| Contoh AI yang Relevan | Expert System, Rule-Based System | Machine Learning, Neural Network |
| Kelebihan | Mudah dijelaskan dan diverifikasi | Lebih adaptif dan kontekstual |
| Kekurangan | Kurang fleksibel untuk kasus baru | Sulit dijelaskan dan dipertanggungjawabkan |

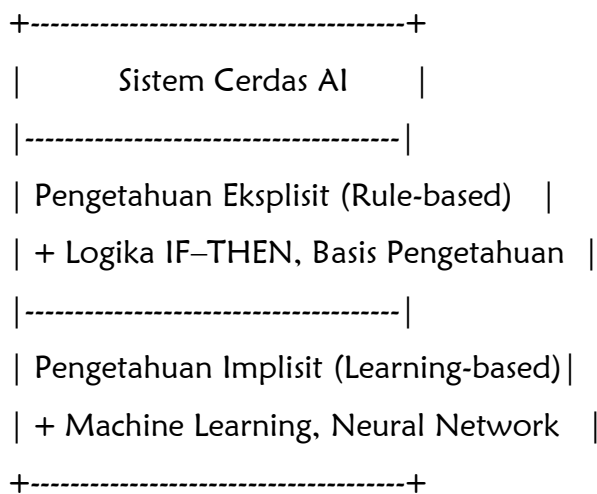
5. Keterkaitan Keduanya dalam Sistem AI

Dalam sistem cerdas modern, pengetahuan eksplisit dan implisit tidak dipisahkan secara tegas, melainkan dikombinasikan untuk mencapai hasil terbaik.

Contoh Kombinasi

- Sistem pakar medis menggunakan aturan eksplisit untuk diagnosa umum, tetapi juga *machine learning* (implisit) untuk mempelajari pola penyakit baru.
- Mobil otonom memiliki aturan eksplisit (rambu lalu lintas) sekaligus pengetahuan implisit dari pembelajaran pengalaman mengemudi.

Model Integratif AI:



B. Model representasi: Propositional Logic, Predicate Logic

1. Pengantar

Agar sistem AI dapat berpikir dan menalar seperti manusia, pengetahuan yang dimilikinya harus diubah ke dalam bentuk formal yang bisa diolah oleh



komputer. Proses ini disebut representasi pengetahuan (*knowledge representation*).

Knowledge representation = cara menyimpan fakta, aturan, atau hubungan antar konsep dalam format yang bisa digunakan oleh mesin inferensi (*inference engine*). Sebuah model representasi pengetahuan yang baik harus memenuhi beberapa karakteristik:

Ciri-ciri Representasi Pengetahuan yang Baik

1. Representational Adequacy — mampu menggambarkan berbagai jenis pengetahuan.
2. Inferential Adequacy — memungkinkan sistem untuk melakukan penalaran.
3. Inferential Efficiency — memudahkan proses penarikan kesimpulan.
4. Acquisition Efficiency — pengetahuan mudah diperoleh dan dimasukkan ke sistem.

2. Jenis-jenis Model Representasi Pengetahuan

Berikut empat model representasi pengetahuan utama yang umum digunakan dalam kecerdasan buatan:

1. Logika Proposisional (Propositional Logic)
2. Logika Predikat (Predicate Logic / First-Order Logic)
3. Semantic Network
4. Frame-Based Representation

2.1 Logika Proposisional (*Propositional Logic*)

Logika proposisional adalah bentuk representasi pengetahuan paling dasar yang menyatakan fakta-fakta sederhana dalam bentuk pernyataan (proposisi) yang bernilai benar (True) atau salah (False).

Contoh proposisi:

- P : “Hujan turun.”
- Q : “Jalanan basah.”

Dapat digabungkan dengan operator logika:



- NOT ($\neg P$) → Negasi
- AND ($P \wedge Q$) → Konjungsi
- OR ($P \vee Q$) → Disjungsi
- IMPLIES ($P \rightarrow Q$) → Implikasi
- IFF ($P \leftrightarrow Q$) → Biimplikasi

Contoh:

Jika hujan turun, maka jalanan basah.

($P \rightarrow Q$)

Jika diketahui (P) benar (hujan turun), maka sistem dapat menalar (Q) (jalanan basah) juga benar. Inilah dasar inferensi sederhana dalam sistem berbasis aturan (*rule-based system*).

2.2 Logika Predikat (Predicate Logic / First-Order Logic)

Logika predikat memperluas logika proposisional dengan **menggunakan variabel dan kuantor** untuk menggambarkan hubungan antar objek.

Bentuk Umum:

Predikat(Objek, ...)

Contoh:

- Manusia(Ali)
- Menyukai(Ali, Kopi)

Kuantor

| Simbol | Arti | Contoh | Makna |
|-----------|-----------------------|--|---------------------------|
| \forall | Untuk semua (for all) | $\forall x$ (Manusia(x) \rightarrow Bernapas(x)) | Semua manusia bernapas |
| \exists | Ada (exists) | $\exists x$ (Mahasiswa(x) \wedge Pintar(x)) | Ada mahasiswa yang pintar |

Contoh dalam Sistem Pakar

$\forall x$ (Penyakit(x) \wedge Menular(x) \rightarrow Butuh_Isolasi(x))

Artinya: *Semua penyakit yang menular membutuhkan isolasi.*

Kecerdasan Buatan (Artificial Intelligence)



Keunggulan: dapat menggambarkan pengetahuan kompleks (hubungan antar objek).

Kelemahan: lebih sulit diimplementasikan secara efisien dibanding logika proposisional.

2.3 Semantic Network (Jaringan Semantik)

Semantic Network adalah model representasi pengetahuan menggunakan node (simpul) dan edge (relasi) untuk menggambarkan hubungan antar konsep.

- Node = konsep / objek.
- Edge = hubungan antar konsep (misalnya “adalah”, “memiliki”, “bagian dari”).

Contoh:

[Burung] —is-a—> [Hewan]

[Burung] —has—> [Sayap]

[Burung] —can—> [Terbang]

Artinya:

Burung adalah hewan, memiliki sayap, dan dapat terbang.

Manfaat Semantic Network:

- Representasi relasional dan hierarkis.
- Memungkinkan pewarisan sifat (inheritance):
Jika “Burung adalah Hewan” dan “Hewan dapat bernapas”, maka sistem dapat menyimpulkan “Burung dapat bernapas.”

Contoh dalam AI:

Digunakan dalam *Natural Language Processing*, *Ontologi Web (OWL)*, dan *Chatbot* berbasis relasi konsep.

2.4 Frame-Based Representation

Frame adalah struktur data yang menyerupai objek dan digunakan untuk menggambarkan entitas atau situasi tertentu dalam bentuk slot dan nilai (*attribute-value pairs*).

Kecerdasan Buatan (Artificial Intelligence)



Setiap frame merepresentasikan satu konsep, objek, atau situasi.

Struktur Umum Frame:

Frame: Burung

Slot: Jenis = Unggas

Slot: Sayap = Ya

Slot: Warna = Beragam

Slot: Terbang = Ya

Contoh Relasi Frame:

Frame: Elang

ISA: Burung

Slot: Habitat = Pegunungan

Slot: KecepatanTerbang = Tinggi

Artinya: Elang adalah burung yang hidup di pegunungan dan memiliki kecepatan terbang tinggi.

Kelebihan Frame:

- Memudahkan pengorganisasian informasi kompleks.
- Dapat digunakan bersama inheritance (pewarisan) seperti pada *object-oriented programming (OOP)*.
- Cocok untuk *knowledge-based system* dan *object recognition*.

3. Perbandingan Antar Model Representasi Pengetahuan

| Aspek | Logika Proposisional | Logika Predikat | Semantic Network | Frame-Based |
|--------------|-----------------------|---------------------------|-----------------------------|--|
| Struktur | Pernyataan sederhana | Relasi antar objek | Node dan hubungan | Slot dan nilai |
| Kemampuan | Fakta dasar | Fakta kompleks | Hubungan konseptual | Pengetahuan berorientasi objek |
| Kelebihan | Sederhana, efisien | Ekspresif dan fleksibel | Visual dan mudah dipahami | Terstruktur dan modular |
| Kelemahan | Kurang ekspresif | Implementasi rumit | Sulit untuk inferensi besar | Butuh desain detail |
| Contoh Kasus | Aturan logika IF-THEN | Relasi "manusia bernapas" | Ontologi "burung-hewan" | Representasi entitas "mobil", "burung" |

Kecerdasan Buatan (Artificial Intelligence)



4. Kombinasi Model dalam Sistem AI Modern

Sistem AI modern sering menggabungkan beberapa model representasi agar lebih kuat dan fleksibel.

Contoh Integrasi:

- *Rule-based logic* (untuk inferensi deduktif)
- *Frame structure* (untuk menyimpan atribut objek)
- *Semantic network* (untuk koneksi antar konsep)
- *Machine learning* (untuk mengekstrak pengetahuan implisit dari data)

Kasus nyata:

Chatbot cerdas dapat menggunakan:

- NLP + Semantic Network → untuk memahami konteks kalimat
- Rules + Frame → untuk memberikan respon yang sesuai

C. Inferensi Logika dan Aturan Inferensi (modus ponens, modus tollens)

1. Pengantar Inferensi Logika

Inferensi (*Inference*) adalah proses menarik kesimpulan logis berdasarkan pengetahuan atau premis yang sudah diketahui. Dalam sistem kecerdasan buatan, inferensi logika digunakan oleh mesin inferensi (*inference engine*) untuk menghasilkan pengetahuan baru dari fakta dan aturan yang ada dalam basis pengetahuan (*knowledge base*).

Contoh sederhana:

Premis 1: Jika suhu $> 38^{\circ}\text{C}$ maka pasien demam.

Premis 2: Suhu pasien 39°C .

Kesimpulan: Pasien demam.

Proses menarik kesimpulan dari dua premis tersebut disebut penalaran logis (logical reasoning).

2. Jenis Penalaran dalam AI

| Jenis Penalaran | Arah Penalaran | Ciri Utama |
|-----------------|---------------------|---|
| Deduktif | Dari umum ke khusus | Menghasilkan kesimpulan pasti dari premis umum. |

Kecerdasan Buatan (Artificial Intelligence)



| | | |
|----------|------------------------------------|--|
| Induktif | Dari khusus ke umum | Menghasilkan generalisasi dari observasi data. |
| Abduktif | Dari hasil ke kemungkinan penyebab | Digunakan untuk diagnosis atau hipotesis. |

Dalam sistem pakar, penalaran deduktif dan abduktif paling sering digunakan.

3. Dasar Inferensi Logika dalam AI

AI sering menggunakan logika proposisional untuk inferensi sederhana karena bentuknya efisien dan mudah diimplementasikan.

Bentuk umum pernyataan logika:

Jika P maka Q

$(P \rightarrow Q)$

Artinya:

Jika kondisi P benar, maka hasil Q juga benar.

4. Aturan Inferensi Utama

Dua aturan inferensi paling mendasar dalam AI adalah:

A. Modus Ponens (Aturan Penegasan)

Bentuk Umum:

1. Jika P maka Q

2. P benar

\therefore Q benar

(\therefore dibaca “maka dapat disimpulkan”)

Contoh 1

Premis 1: Jika hujan turun maka jalanan basah.

Premis 2: Hujan turun.

Kesimpulan: Jalanan basah.

Contoh 2 (dalam sistem pakar medis)

Premis 1: Jika suhu tubuh $> 38^{\circ}\text{C}$ maka pasien demam.

Premis 2: Suhu pasien = 39°C .

Kesimpulan: Pasien demam.



B. Modus Tollens (Aturan Penyangkalan)

Bentuk Umum:

1. Jika P maka Q
 2. Q salah ($\neg Q$)
- \therefore P salah ($\neg P$)

Contoh 1

Premis 1: Jika hujan turun maka jalanan basah.

Premis 2: Jalanan tidak basah.

Kesimpulan: Tidak hujan turun.

Contoh 2 (dalam sistem pakar kesehatan)

Premis 1: Jika pasien menderita influenza maka suhu tubuh tinggi.

Premis 2: Suhu tubuh normal.

Kesimpulan: Pasien tidak menderita influenza.

5. Aturan Inferensi Lain yang Sering Digunakan

Selain *modus ponens* dan *modus tollens*, sistem inferensi dalam AI juga menggunakan aturan-aturan berikut:

| No | Nama Aturan | Bentuk | Penjelasan |
|----|-------------------------------------|---|--|
| 1 | Chain Rule / Hypothetical Syllogism | Jika $P \rightarrow Q$ dan $Q \rightarrow R$ maka $P \rightarrow R$ | Penalaran berantai antar aturan |
| 2 | Conjunction | $P, Q \rightarrow (P \wedge Q)$ | Menggabungkan dua fakta benar |
| 3 | Disjunction Introduction | $P \rightarrow (P \vee Q)$ | Jika P benar, maka "P atau Q" juga benar |
| 4 | Simplification | $(P \wedge Q) \rightarrow P$ | Menarik salah satu fakta dari konjungsi |



| | | | |
|---|------------|--|--|
| 5 | Resolution | $(P \vee Q), (\neg P \vee R) \rightarrow (Q \vee R)$ | Digunakan dalam logika otomatis (<i>automated reasoning</i>) |
|---|------------|--|--|

6. Arah Penalaran dalam Sistem AI

| Jenis Chaining | Arah Logika | Deskripsi |
|-------------------|--------------------------|---|
| Forward Chaining | Dari fakta ke kesimpulan | Jika semua kondisi terpenuhi \rightarrow hasilnya ditarik. (Modus Ponens) |
| Backward Chaining | Dari kesimpulan ke fakta | Sistem memeriksa apakah kesimpulan dapat dibuktikan. (Modus Tollens) |

Contoh Kombinasi:

1. Rule (Aturan):

Jika gejala = batuk dan demam maka penyakit = ISPA.

2. Fakta:

Pasien mengalami batuk dan demam.

3. Forward Chaining:

\rightarrow Sistem menyimpulkan pasien ISPA.

4. Backward Chaining:

\rightarrow Jika ingin membuktikan "Pasien ISPA", sistem mencari apakah batuk & demam benar.

