Artificial Intelligence (CS280)

Noushin Saba

Lecture 3
Reasoning and Knowledge Representation: A Theoretical Analysis

Outline

Introduction to Knowledge and Reasoning

Formal Foundations of Knowledge Representation

- Facts, Models, Entailment
- Knowledge based Agents

Knowledge Representation Techniques

- Logic (Prepositional & First Order)
- Semantic Networks, Frames, Rules, Ontologies

Reasoning Approaches

- Deductive, Inductive, Abductive
- Inference Procedures
- Soundness & Completeness

Modern Views of KR

- Large Language Models (LLMs)
- Probabilistic Reasoning

Outline

Introduction to Knowledge and Reasoning

Formal Foundations of Knowledge Representation
- Facts, Models, Entailment
- Knowledge based Agents

Knowledge Representation Techniques
- Logic (Prepositional & First Order)
- Semantic Networks, Frames, Rules, Ontologies

Reasoning Approaches
- Deductive, Inductive, Abductive
- Inference Procedures
- Soundness & Completeness

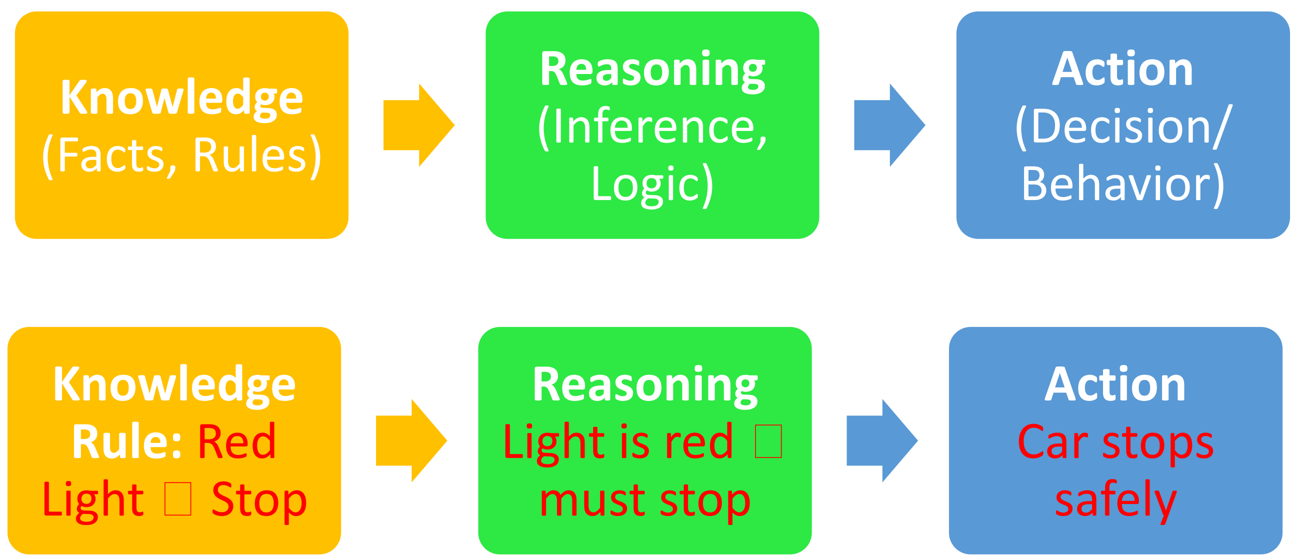
Modern Views of KR
- Large Language Models (LLMs)
- Probabilistic Reasoning

Introduction to Knowledge and Reasoning

- **What is Knowledge?**
  - Knowledge = facts + relationships + rules about the world.
  - Structured, contextual, and meaningful information.
    - Fact: “It is raining.”
    - Knowledge: “If it rains, the ground becomes wet.”
- **What is Reasoning?**
  - The process of using knowledge to draw conclusions or make decisions.
    - Knowledge: “If it rains, the ground becomes wet.”
    - Fact: “It is raining.”
    - Reasoning → “The ground is wet.”

Introduction to Knowledge and Reasoning (cont..)

- **Why Important in AI?**
  - Forms the foundation of intelligent agents.
  - Separates data (knowledge) from processing (reasoning).
  - Enables decision-making, problem-solving, and planning.



Outline

Introduction to Knowledge and Reasoning

Formal Foundations of Knowledge Representation

- Facts, Models, Entailment
- Knowledge based Agents

Knowledge Representation Techniques

- Logic (Prepositional & First Order)
- Semantic Networks, Frames, Rules, Ontologies

Reasoning Approaches

- Deductive, Inductive, Abductive
- Inference Procedures
- Soundness & Completeness

Modern Views of KR

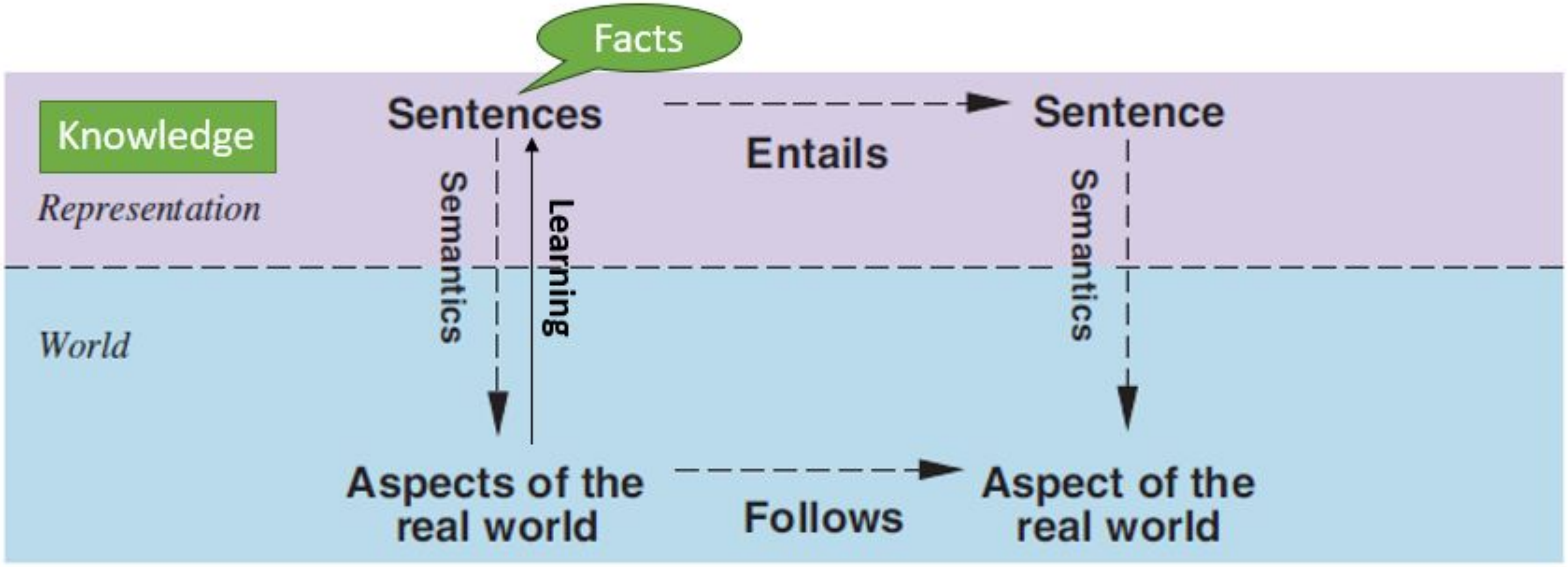
- Large Language Models (LLMs)
- Probabilistic Reasoning

Formal Foundations of Knowledge Representation

- **Facts**
  - Sentences that are known to be true.
- **Models (Possible Worlds)**
  - A **model** = a complete assignment of truth values to all facts.
  - **Possible worlds** = all models consistent with what we know.
  - Learning new facts reduces the number of possible worlds.
    - Fact: “It is raining.”
    - Model 1: Raining = True, Ground wet = True.
    - Model 2: Raining = False, Ground wet = False.
    - *Only Model 1 is consistent with the fact.*

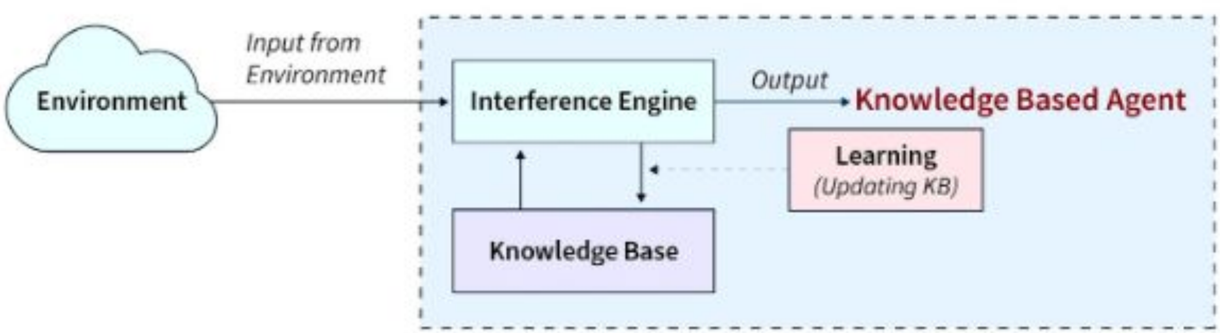
Formal Foundations of Knowledge Representation (cont..)

- **Entailment (⊨)**
  - A sentence logically follows from what we already know.
  - If KB ⊨ α, then in every model where KB is true, α is also true.
    - KB = “It is raining → The ground is wet.”
    - Fact = “It is raining.”
    - Entailment: “The ground is wet.”



Knowledge-Based Agents

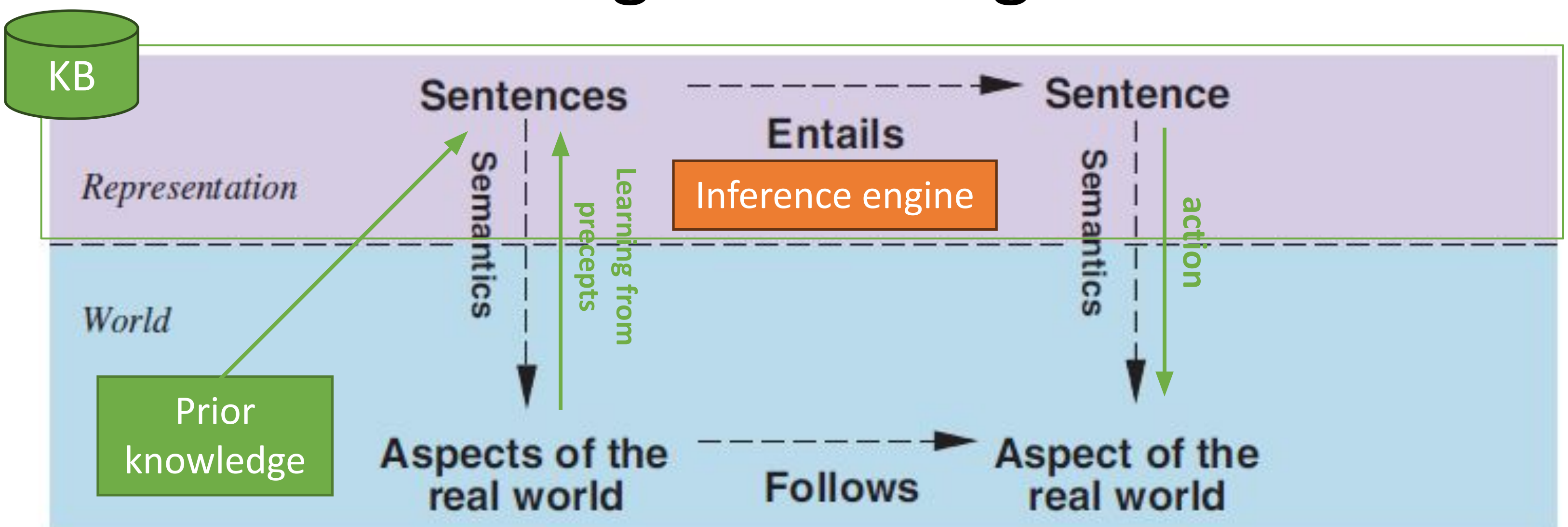
- An agent that reasons and acts using a Knowledge Base (KB).
- KB = Set of facts expressed in a formal language.
- Inference Engine: derives new facts from KB using entailment.
- **Advantages of KB Agents**
  - Separation of Knowledge & Reasoning**:** flexible design.
  - Can learn new facts and adapt actions.
  - Foundation of expert systems.



Knowledge-Based Agents (cont..)

- **Key Components**
  - **Knowledge Base (KB):**
    - Stores facts, rules, and background knowledge.
      - Fact: “It is raining.”
      - Rule: “If it rains, the ground is wet.”
  - **Inference Engine:**
    - Domain-independent algorithms.
    - Derives new information:
      - From above KB → concludes “The ground is wet.”
  - **Agent Program (Decision Maker):**
    - Chooses actions based on knowledge + goals.
      - “If ground is wet → carry umbrella.”

Generic Knowledge-based Agent




```
function KB-AGENT(percept) returns an action
   persistent: KB, a knowledge base
               t, a counter, initially 0, indicating time

   TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
   action ← ASK(KB, MAKE-ACTION-QUERY(t))
   TELL(KB, MAKE-ACTION-SENTENCE(action, t))
   t ← t + 1
   return action
```

Memorize percept at time t

Ask for logical action given an objective

Record action taken at time t

Outline

Introduction to Knowledge and Reasoning

Formal Foundations of Knowledge Representation

- Facts, Models, Entailment
- Knowledge based Agents

Knowledge Representation Techniques

- Logic (Prepositional & First Order)
- Semantic Networks, Frames, Rules, Ontologies

Reasoning Approaches

- Deductive, Inductive, Abductive
- Inference Procedures
- Soundness & Completeness

Modern Views of KR

- Large Language Models (LLMs)
- Probabilistic Reasoning

Logic (Symbolic / Declarative Representation)

- Logic is a formal system for representing knowledge in a precise, unambiguous, and mathematically structured way.
- It allows an intelligent agent to:
  - Represent facts and rules about the world, and
  - Reason by drawing valid conclusions from them.
- It is called declarative because you *declare* what is true, and the reasoning system figures out the consequences automatically.
- **Types of Logic in AI**
  - Propositional Logic (PL)
  - First-Order Logic (FOL)

Propositional Logic (PL)

- Simplest form of logic; deals with *propositions* (statements that can be True or False).
- **Syntax**
  - Atomic propositions (e.g., P, Q, R).
  - Logical connectives ¬ (NOT), ∧ (AND), ∨ (OR), → (IMPLIES), ↔ (IFF).
- **Semantics**
  - Each proposition has a truth value (T/F).
  - The truth of complex statements is determined by truth tables
    - *P: "It is raining."*
    - *Q: "The ground is wet."*
    - *Rule: P → Q*
    - *If P = True → Q must also be True (Entailment).*
- *Interpretation* □ The agent "knows" that if it is raining, the ground becomes wet.

First-Order Logic (FOL)

- Extends propositional logic by introducing objects, relations, and quantifiers.
- **Why needed?**
  - PL cannot represent general statements like “All humans are mortal.”
- **Components of FOL**
  - Constants: Represent objects (e.g., Socrates).
  - Predicates: Represent relationships/properties (e.g., Human(x), Mortal(x)).
  - Quantifiers
    - Universal (∀): For all
    - Existential (∃): There exists
    - $\forall x\ (Human(x) \rightarrow Mortal(x))$
    - $Human(Socrates)$
    - $\vDash Mortal(Socrates)$ *(Socrates is mortal)*
- *Interpretation* □ We can describe general truths and infer facts about specific individuals.

Logic (Symbolic / Declarative Representation) (cont..)

- **Reasoning in Logic**
  - Deduction: Deriving specific facts from general rules.
  - Applies logical rules to move from general knowledge to specific instances.
    - General Rule: $\forall x\ (\text{Human}(x) \rightarrow \text{Mortal}(x))$ — *All humans are mortal.*
    - Fact: Human(Socrates) — *Socrates is a human.*
    - Deduction: Mortal(Socrates) — *Therefore, Socrates is mortal.*
  - Inference Rules: Formal procedures used to draw conclusions
    - Modus Ponens: If $P \rightarrow Q$ and P, then Q.
      E.g. If it rains, the ground is wet → It rains → The ground is wet.
    - Resolution: Combine clauses to infer new facts ($\neg P \lor Q$ and $P \Rightarrow Q$).
    - Unification: Match variables with constants (x = Socrates).
  - Entailment (⊨): A sentence is entailed if it's true in all models of the KB.
    - KB = {It rains → ground wet, It rains} ⊨ ground wet
  - Logical reasoning ensures that every inferred statement follows necessarily from the knowledge base.

Logic (Symbolic / Declarative Representation) (cont..)

- **Strengths**
  - Mathematically exact; statements have clear and precise meanings.
  - Inference produces correct and exhaustive conclusions.
  - Rules can be combined to infer new knowledge automatically.
  - Forms the basis for expert systems, theorem provers, and planning systems.
- **Limitations**
  - Real-world knowledge is rarely fully true or false, making systems brittle.
  - Cannot represent uncertainty such as “probably” or “likely.”
  - Inference processes can become computationally expensive as knowledge size increases.
  - Requires experts to manually encode facts, making knowledge acquisition difficult.
- **Applications**
  - Expert systems MYCIN for medical diagnosis.
  - Automated reasoning theorem provers, SAT solvers.
  - Knowledge-based planning systems.
  - Semantic web ontologies foundation of OWL logic.

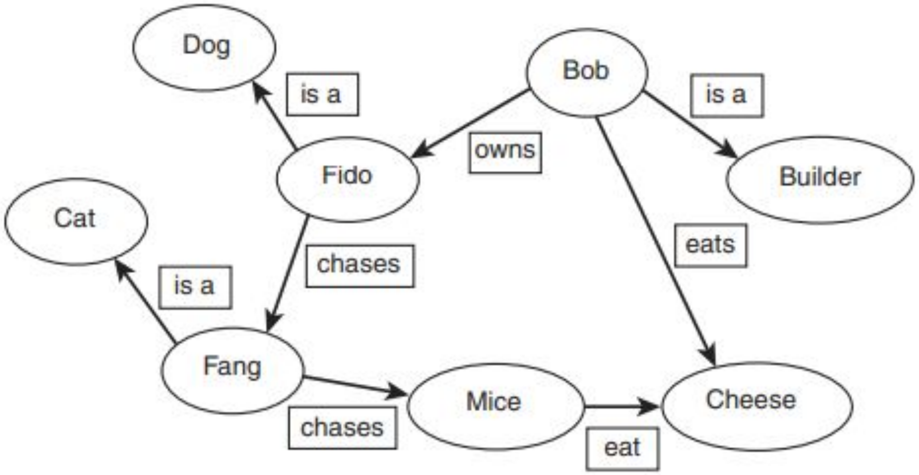
Semantic Networks (Graph-Based Representation)

- A Semantic Network is a graph-based knowledge representation model where
  - Nodes represent concepts or entities, and
  - Edges (links) represent relationships between these concepts.
- It provides a visual and structural way to represent how knowledge is organized and related in the human mind or in an AI system.
- *Interpretation* □ Instead of storing knowledge as text or equations, we represent it as a network of connected meanings — hence the term *"semantic"* (meaning-based)
- **Structure of a Semantic Network**

Element	**Description**
Node	Object, concept, or class
Edge / Link	Relationship between nodes
Inheritance	Lower nodes inherit properties of higher nodes

Semantic Networks (Graph-Based Representation) (cont..)

- This network shows how objects and relationships are represented in AI.
- Nodes represent concepts or entities (e.g., *Bob, Fido, Cheese*).
- Links show relationships (e.g., *is-a, owns, chases, eats*).
- Knowledge flows through these connections, allowing reasoning and inheritance.



Semantic Networks (Graph-Based Representation) (cont..)

- **Reasoning in Semantic Networks**
- Semantic networks support reasoning through inheritance and associative links:
  - Inheritance: Lower-level nodes automatically acquire properties from higher-level nodes.
  - *Since Fido is a Dog, and Dogs are Animals, Fido inherits all animal properties (e.g., needs food, can move).*
  - *Fido → Dog → Animal → inherits properties like "can move" and "needs food."*
  - Spreading Activation: When one concept is activated, related concepts are also activated.
  - *Activating Bob in the network may trigger related nodes like Fido (his pet), Builder (his occupation), or Cheese (something he eats).*

Semantic Networks (Graph-Based Representation) (cont..)

- **Types of Semantic Networks**
  - **Definitional Networks**
    - Represent “is-a” hierarchies and taxonomies.
    - Dog □ Mammal □ Animal
  - **Assertional Networks**
    - Represent specific facts about instances.
    - Fido □ instance-of □ Dog
  - **Inheritance Networks**
    - Represent property inheritance and exceptions.
      - Bird □ can fly
      - Penguin □ Bird, but *cannot fly* (exception handling)
  - **Conceptual Dependency Networks**
    - Represent actions and relations (used in NLP).
    - “John eats an apple” □(ACTOR: John, ACTION: Eat, OBJECT: Apple)

Semantic Networks (Graph-Based Representation) (cont..)

- **Strengths**
  - Intuitive and visual; easy to understand for both humans and machines.
  - Supports inheritance by automatically transferring properties down hierarchy.
  - Efficient for retrieval and quick association of related concepts.
  - Useful in applications like semantic search and knowledge graphs.
- **Limitations**
  - Relationships between nodes may lack clear or formal semantics.
  - Cannot easily represent complex logical constraints.
  - Does not handle uncertainty; all relationships are assumed definite.
  - Scalability issues arise as large networks become dense and difficult to maintain.
- **Applications**
  - WordNet □ Large lexical database linking words by meaning (used in NLP).
  - Google Knowledge Graph □Represents entities and their relations for semantic search.
  - Expert Systems □ Conceptual understanding in diagnosis or classification.
  - ConceptNet □ Open-source common-sense reasoning network.

Frames (Structured Representation)

- Frames are data structures that represent structured knowledge about objects, situations, or concepts in terms of attributes (called slots) and their corresponding values.
  Each frame acts like a template or schema that describes an entity and its properties.
- It allows an intelligent system to:
  - Store and organize knowledge in a hierarchical way.
  - Use default reasoning and inheritance to fill missing details.
- *Interpretation □ Frames are similar to "objects" in object-oriented programming, where each object has attributes, default values, and possibly relationships to other objects.*
- **Structure of a Frame** A frame consists of
  - Slots □ Attributes or properties of the object.
  - Values □ Specific information or default values for each slot.
  - Inheritance Links □ Connect frames in a hierarchy, allowing lower-level frames to inherit properties from higher-level frames.

Frames (Structured Representation) (cont..)

- The Vehicle frame defines general attributes (slots).
- The Car frame inherits all attributes from Vehicle, adds new ones (Type, Color), and can override defaults.
- The ElectricCar frame further inherits and overrides (Engine ⟶ Electric).
- When a query is made for a property, the system uses:
  - **Instance value**, if available.
  - **Default value**, if not specified.
  - **Inherited value**, if neither of the above is defined.

Vehicle (Frame)

- Wheels: 4
- Engine: Generic
- HasDoors: Yes

Car (Frame)

- Inherits from: Vehicle
- Engine: Petrol
- Type: Sedan
- Color: White

Electric Car (Frame)

- Inherits from: Car
- Engine: Electric (Overrides)
- Battery: Lithium-Ion
- Range: 350 km

Frames (Structured Representation) (cont..)

- **Reasoning with Frames**
- Reasoning in frames allows AI systems to reuse, extend, and infer knowledge efficiently through inheritance and default logic.
  - Inheritance: Lower-level frames automatically acquire attributes from higher-level frames.
  - The Car frame inherits Wheels: 4 and HasDoors: Yes from Vehicle.
  - Default Reasoning: If a slot has no specific value, the default is used.
  - The Car frame uses Color: White unless another value is specified.
  - Procedural Attachment: Frames can trigger actions when a slot is accessed or updated.
  - Accessing the Engine slot in ElectricCar may call a procedure to compute Battery Range.

Frames (Structured Representation) (cont..)

- **Strengths**
  - Organizes knowledge in a structured and hierarchical way.
  - Supports default reasoning and inheritance.
  - Captures common-sense and contextual knowledge.
  - Easy to understand and maintain due to object-like structure.
- **Limitations**
  - Difficult to handle uncertainty or changing information.
  - Can become complex in large hierarchies.
  - Not ideal for representing procedural or relational knowledge (e.g., time-based or logical relationships).
- **Applications**
  - Expert systems □Medical or engineering diagnosis.
  - Computer vision □ Scene and object understanding.
  - Natural language processing □ Context and meaning representation.
  - Knowledge-based robotics □ Representing structured world models.

Production Rules (Rule-Based Systems)

- Production rules represent procedural knowledge — what actions to take when specific conditions are met.
- They are expressed as IF–THEN statements, describing a cause–effect or condition–action relationship.
- *Interpretation* Rules simulate human decision-making logic and are widely used in expert systems.
- **Structure of a Rule**
  - A production rule has two parts:
  - Condition (IF) □ The situation or fact to be checked.
  - Action (THEN) □ The conclusion or response triggered when the condition is satisfied.

Production Rules (Rule-Based Systems) (cont..)

- **Rule 1:** IF fever AND cough THEN possible flu
- **Rule 2:** IF temperature > 38°C THEN fever = True
- **Known Facts**
  - temperature = 39°C
  - cough = True
- **Reasoning Flow**
  - The system checks Rule 2 → since temperature > 38°C → infer fever = True.
  - Now Rule 1 applies → since fever = True and cough = True → infer possible flu.

Production Rules (Rule-Based Systems) (cont..)

- **Reasoning in Rule-Based Systems**
  - There are two main reasoning strategies:
  - Forward Chaining (Data-Driven): Starts with known facts and applies rules to derive new facts.
    - Given *temperature > 38°C*, infer *fever = True*, then infer *possible flu*.
  - Backward Chaining (Goal-Driven): Starts with a goal and works backward to determine if facts support it.
    - To test “Does the patient have flu?”, the system checks rules that conclude flu.

Production Rules (Rule-Based Systems) (cont..)

- **Strengths**
  - Simple and intuitive to design (“if–then” format).
  - Supports step-by-step reasoning.
  - Easy to modify or add new rules.
  - Transparent reasoning process — easy to explain conclusions.
- **Limitations**
  - Large systems can contain thousands of rules that are hard to manage.
  - Multiple rules may trigger simultaneously.
  - Cannot easily handle **probabilistic or uncertain reasoning**.
- **Applications**
  - Expert systems MYCIN (medical diagnosis), DENDRAL (chemical analysis).
  - Decision support systems Business and industrial rule engines.
  - Control systems Troubleshooting, maintenance, configuration.
  - Game AI Strategy-based action triggering.

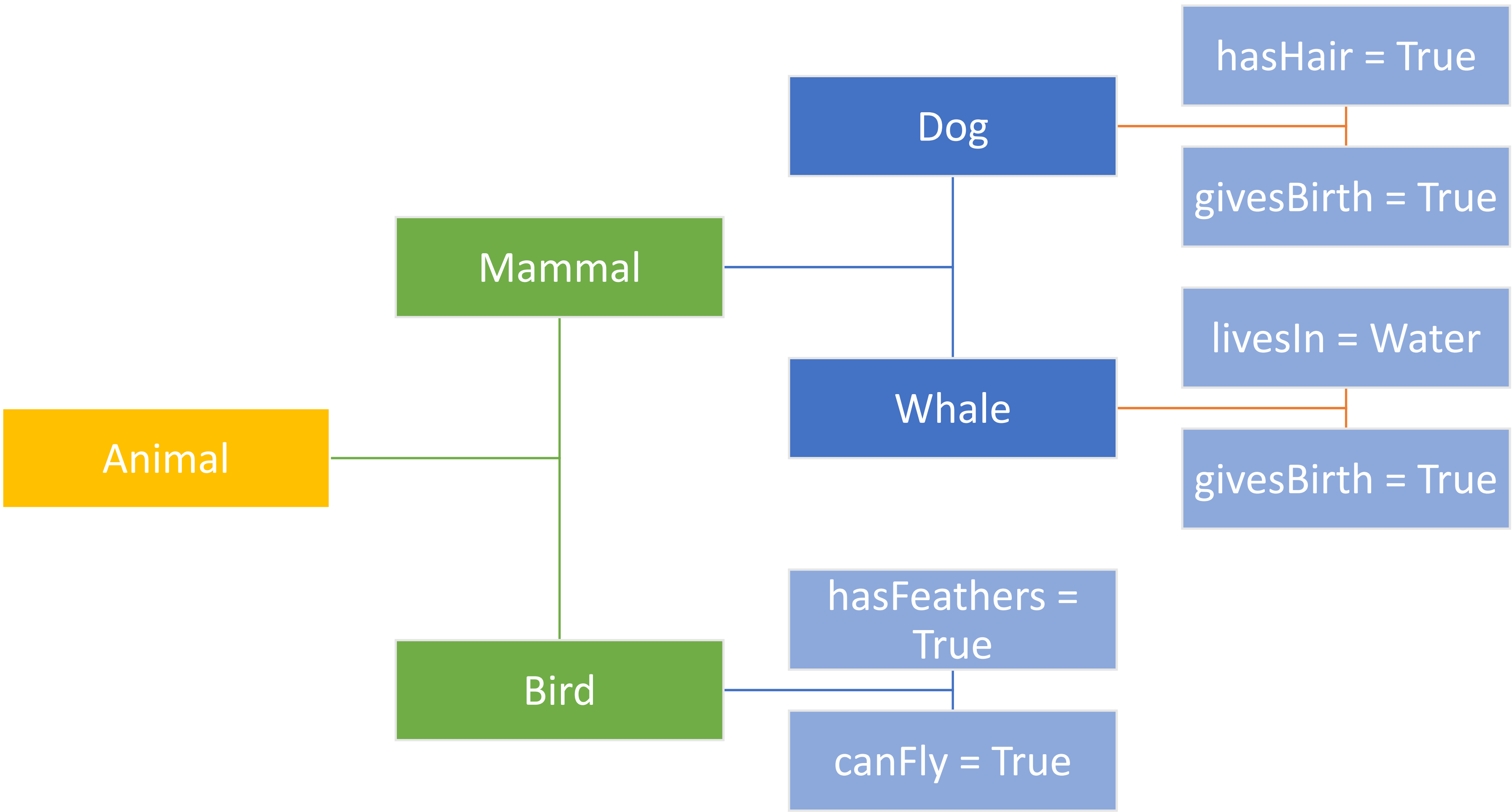
Ontologies (Hierarchical Representation)

- An ontology is a formal, explicit, and hierarchical representation of knowledge within a domain.
  It defines concepts (classes), relationships, properties, and instances to enable machines to understand and reason about domain information.
- *Interpretation* □ Ontologies provide a shared vocabulary and structure that allows knowledge to be reused and exchanged between systems.
- **Components of an Ontology**

Component	**Description**
Classes	Abstract categories of objects
Subclasses	More specific classes within a category
Relations	Define how classes connect
Properties	Attributes or characteristics
Instances	Concrete examples of a class

Ontologies (Hierarchical Representation) (cont..)

- Dog and Whale inherit properties from Mammal and Animal.
- Ontology allows reasoning such as:
  - If an entity is a Dog, it is a Mammal and therefore an Animal.
  - If Mammals give birth, we can infer that Dogs give birth too.



Ontologies (Hierarchical Representation) (cont..)

- **Reasoning with Ontologies**
- Enables logical inference, error detection, and automatic classification across related concepts.
  - Inheritance: Subclasses inherit properties from parent classes.
  - Since “Dog” is a subclass of “Mammal,” it inherits givesBirth = True and hasHair = True.
  - Classification: Determines where new instances belong in the hierarchy.
  - If “Tweety” has hasFeathers = True and canFly = True, it is classified as a Bird.
  - Consistency Checking: Ensures that relationships don’t contradict each other.
  - An instance cannot be both a Bird and a Fish if those classes are defined as disjoint.
  - Semantic Inference: Discovers new knowledge from existing relationships.
  - If “All mammals are animals” and “Dogs are mammals,” the system infers that Dogs are animals.

Ontologies (Hierarchical Representation) (cont..)

- **Strengths**
  - Provides shared and reusable domain knowledge.
  - Enables semantic interoperability between systems.
  - Supports reasoning, classification, and data integration.
  - Essential for Semantic Web and NLP.
- **Limitations**
  - Requires significant domain expertise to design.
  - Complex and time-consuming to maintain and update.
  - Limited scalability for very large or dynamic domains.
- **Applications**
  - Semantic Web technologies RDF, OWL, SPARQL.
  - Healthcare ontologies SNOMED CT, Gene Ontology.
  - Knowledge graphs Google Knowledge Graph, DBpedia.
  - Natural Language Processing Concept linking, entity recognition.

Outline

Introduction to Knowledge and Reasoning

Formal Foundations of Knowledge Representation

- Facts, Models, Entailment
- Knowledge based Agents

Knowledge Representation Techniques

- Logic (Prepositional & First Order)
- Semantic Networks, Frames, Rules, Ontologies

Reasoning Approaches

- Deductive, Inductive, Abductive
- Inference Procedures
- Soundness & Completeness

Modern Views of KR

- Large Language Models (LLMs)
- Probabilistic Reasoning

Reasoning Approaches in AI

- Reasoning is the process through which an intelligent system derives conclusions or makes decisions based on existing knowledge and evidence.
- Different reasoning approaches define how conclusions are reached — from general to specific, or from effects to causes.
- **Types of Reasoning**
  - Deductive Reasoning (Top-Down)
    - Draws specific conclusions from general rules.
    - **Form:** If A $\rightarrow$ B, and A is true, then B is true.
    - **Rule**: All birds can fly.
      Fact: Parrot is a bird.
      $\Rightarrow$ Conclusion: Parrot can fly.
    - *Used for logical, certain conclusions.*

Reasoning Approaches in AI (cont..)

- Inductive Reasoning (Bottom-Up)
  - Generalizes rules or patterns from observed examples or data.
  - Observation: $Swan_1$, $Swan_2$, $Swan_3$ are white.
    ⇒ Conclusion: All swans are white.
  - *Used in machine learning — patterns inferred from data.*
- Abductive Reasoning (Best Explanation)
  - Infers the most likely cause from an observation.
  - Observation: Ground is wet.
    Possible causes: It rained, sprinkler was on, someone spilled water.
    ⇒ Most likely: It rained.
  - *Used in diagnostic systems and hypothesis generation.*

Reasoning Approaches in AI (cont..)

- **Inference Procedures**
  - Forward Chaining (Data-Driven)
    - Starts with known facts → applies rules → derives new conclusions.
    - Used in production systems (e.g., MYCIN).
  - Backward Chaining (Goal-Driven)
    - Starts with a goal → works backward → checks supporting facts.
    - Used in systems like Prolog.
  - Resolution
    - A rule-based method to prove logical conclusions using contradictions.
    - Used in automated theorem proving.

Reasoning Approaches in AI (cont..)

- **Soundness & Completeness**
- Soundness
  - If the inference procedure derives a conclusion, that conclusion is logically correct (no false results).
- Completeness
  - If a conclusion is logically entailed, the inference procedure can derive it (no missing truths).
- *Goal in AI reasoning: inference systems should be both sound and complete.*

Outline

Introduction to Knowledge and Reasoning

Formal Foundations of Knowledge Representation

- Facts, Models, Entailment
- Knowledge based Agents

Knowledge Representation Techniques

- Logic (Prepositional & First Order)
- Semantic Networks, Frames, Rules, Ontologies

Reasoning Approaches

- Deductive, Inductive, Abductive
- Inference Procedures
- Soundness & Completeness

Modern Views of KR

- Large Language Models (LLMs)
- Probabilistic Reasoning

Modern Views of KR

- Modern approaches to Knowledge Representation (KR) go beyond symbolic systems like logic and semantic networks.
- Integrate probabilistic models, neural networks, and Large Language Models (LLMs).
- Enable reasoning under uncertainty and incomplete information.
- Provide scalability to handle large, complex real-world data.
- Learn and represent knowledge automatically from data rather than predefined rules.
- Bridge the gap between symbolic reasoning and data-driven learning.
- Traditional KR focused on explicit rules; modern KR emphasizes learning representations automatically from data.

Large Language Models (LLMs)

- LLMs (e.g., GPT, BERT, PaLM) represent knowledge implicitly within high-dimensional vector spaces learned from massive text data.
- They encode semantics, context, and relationships without needing predefined rules.
- Learn from large-scale data (trillions of tokens).
- Represent meaning through embeddings (numerical representations of words/concepts).
- Enable reasoning via contextual understanding and in-context learning.
- Prompt: “If Paris is in France, where is Berlin?” □LLM infers: “Germany” — by associating learned patterns rather than explicit logic rules.

Large Language Models (LLMs) (cont..)

- **Advantages**
  - Capture real-world knowledge implicitly.
  - Handle ambiguity and natural language variability.
  - Continuously adaptable to new data.
- **Limitations**
  - Lack explicit reasoning transparency.
  - May generate plausible but incorrect answers (hallucination).
  - Knowledge not easily inspectable or modifiable.
- *Interpretation* □ LLMs shift KR from symbolic representation to statistical and neural representation, bridging knowledge and language understanding.

Probabilistic Reasoning

- Traditional logic deals with *certainty* (True/False).
- Probabilistic reasoning handles *uncertainty* using mathematical probability to infer likely conclusions.
- **Core Concept**
  Instead of “The patient has flu,” we represent:
  P(Flu | Fever, Cough) = 0.85
- **Common Models**
  - Bayesian Networks Graphical models representing probabilistic dependencies.
  - Hidden Markov Models (HMMs) Used for sequential reasoning (e.g., speech recognition).
  - Markov Decision Processes (MDPs) Used in AI planning and reinforcement learning.

Probabilistic Reasoning (cont..)

- **Example (Bayesian Inference)**
  - If 90% of flu patients have fever, and 10% of non-flu patients have fever, then observing "fever" increases belief in "flu" but doesn't guarantee it.
- **Advantages**
  - Handles noisy or incomplete data.
  - Supports decision-making under uncertainty.
  - Widely used in real-world AI (diagnosis, prediction, robotics).
- **Limitations**
  - Requires large, accurate probability data.
  - Computationally intensive for large networks.
- *Interpretation* Probabilistic reasoning combines logic and statistics to make rational decisions under uncertainty — a foundation for modern intelligent systems.

Integration of Symbolic + Neural KR (Neuro-Symbolic AI)

- Modern AI research seeks to combine structured symbolic reasoning (logic, ontologies) with data-driven learning (LLMs, neural nets).
- **Goal**
  - Achieve systems that can learn, reason, and explain.
- Use an LLM for text understanding and a logic layer for ensuring factual consistency.
- Neuro-symbolic systems represent the future of KR, enabling AI that is both intelligent and interpretable.

Summary

- We explored
  - **Knowledge & Reasoning:** How systems represent, reason, and derive conclusions using facts, models, and entailment.
  - **Representation Techniques:** Logic, Semantic Networks, Frames, Rules, and Ontologies for structured and relational knowledge.
  - **Reasoning Approaches:** Deductive, Inductive, Abductive reasoning; Forward/Backward Chaining; Soundness & Completeness.
  - **Modern KR:** Shift to data-driven models — LLMs for language-based reasoning and Probabilistic methods for uncertainty handling.
  - **Trend:** Integration of **symbolic** and **neural** reasoning — forming the basis of **Neuro-Symbolic AI**.
- *"Traditional AI focused on representing what we know; modern AI learns how to represent knowledge itself — bridging logic, probability, and language to achieve true understanding*."