Knowledge representation

May 2025

Topics to be discussed

Part I. Introduction

1. Introduction to Artificial intelligence

Part II. Search and optimisation.

2. Search - basic approaches
3. Search - optimisation
4. Two-player deterministic games
5. Evolutionary and genetic algorithms

Part III. Machine learning and data analysis.

6. Regression, classification and clustering (Part I & II)
8. Artificial neural networks
9. Bayesian models
10. Reinforcement Learning

Part IV. Logic, Inference, Knowledge Representation

11. Propositional logic and predicate logic
12. Knowledge Representation

Part V. AI in Action: Language, Vision

13. AI in Natural language processing
14. AI in Vision and Perception

Part VI. Summary

15. AI engineering, Explainable AI, Ethics

Outline

1. Introduction to Knowledge representation in AI
   - Some views on Knowledge Representation
2. Knowledge Representation in Logic
   - Types of reasoning in logic
   - Modal Logics in AI
   - Epistemic Logic
   - ...
3. Selected Other Symbolic Methods
   - Rough Sets
   - Rule-based Systems
   - Argumentation
   - Semantic Networks
   - Ontologies
   - ...
4. Symbolic vs. Subsymbolic
   - Key characteristics
   - Examples
5. Subsymbolic Methods
   - Artificial Neural Networks and Deep Learning
   - Genetic and evolutionary algorithms
   - Reinforcement learning
   - ...

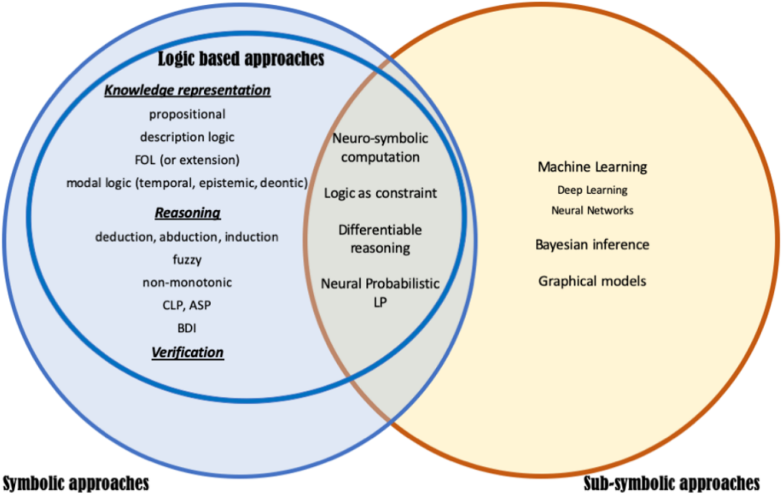
Note: Today we will focus mainly on symbolic knowledge representation (other are covered during remaining lectures on this course).

Introduction to Knowledge representation in AI

- What is Knowledge?
- What is Knowledge Representation?
- What are methods for Knowledge Representation and Processing?

- Why is Knowledge Representation important in AI?
- What are the challenges in Knowledge Representation?

Introduction to Knowledge representation in AI(Cont.)



https://www.mdpi.com/2078-2489/11/3/167

What is Knowledge Representation?

- Knowledge representation is a field of artificial intelligence that focuses on the design and implementation of formal systems for representing and manipulating knowledge.
- The goal of knowledge representation is to enable computers to reason about the world in the same way that humans do, by representing knowledge in a structured and organized manner.
- There are many different approaches to knowledge representation, including:
  - Logic-based approaches, which use symbolic logic to represent knowledge in a precise and unambiguous way.
  - Semantic networks, which represent knowledge as a network of interconnected concepts or nodes.
  - Frames, which represent knowledge as a collection of interrelated attributes and values.
  - Ontologies, which provide a formal, shared vocabulary for representing knowledge in a particular domain.
  - Subsymbolic approaches, which represent knowledge using distributed representations that are not tied to specific symbols or concepts.
- These approaches differ in terms of their underlying assumptions about the nature of knowledge and the best way to represent it.

What is Knowledge Representation? (Cont.)

- Examples of subsymbolic approaches to knowledge representation include:
  - Artificial neural networks, which learn to recognize patterns in data and can be used for tasks such as image classification and speech recognition.
  - Fuzzy logic, which allows for the representation of uncertainty and vagueness in knowledge.
  - Evolutionary algorithms, which use principles of natural selection to evolve solutions to problems.
  - Swarm intelligence, which models the collective behavior of groups of agents to solve complex problems.
  - Bayesian networks, which represent uncertain knowledge using probabilistic models.
- One important aspect of knowledge representation is the ability to reason about uncertainty and incomplete information.
- Many real-world problems involve uncertain or incomplete information, and the ability to reason about this uncertainty is essential for making accurate predictions and decisions.
- Knowledge representation is a critical component of many artificial intelligence systems, and has important applications in fields such as natural language processing, robotics, and expert systems.

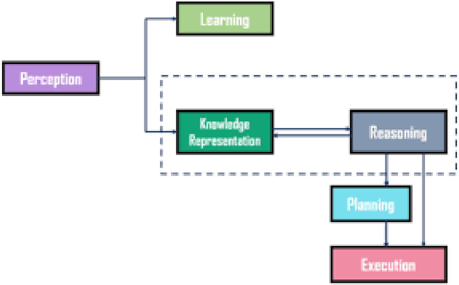
Why is Knowledge Representation important in AI?

- Efficient Problem Solving:
  - Knowledge representation allows AI systems to store and organize large amounts of information efficiently.
  - It provides a structured framework to model complex problems, enabling efficient problem solving.
- Inference and Reasoning:
  - Knowledge representation enables AI systems to perform logical reasoning and make inferences.
  - It allows the system to draw conclusions, derive new knowledge, and make informed decisions.
- Learning and Adaptability:
  - Knowledge representation facilitates learning and adaptation in AI systems.
  - It enables the system to acquire new knowledge, update existing knowledge, and improve its performance over time.
- Interoperability and Collaboration:
  - Knowledge representation provides a common language and format for sharing and exchanging information between AI systems.
  - It enables interoperability, allowing different systems to work together and collaborate effectively.

Cycle of Knowledge Representation in AI

Artificial Intelligent Systems usually consist of various components to display their intelligent behavior.

- **Perception**: The system acquires information from its environment through sensors or data sources.
- **Learning**: The system learns from the acquired data to improve its performance or adapt to new situations.
- **Knowledge Representation & Reasoning**: The system represents and organizes the acquired knowledge in a structured format, allowing for efficient retrieval and reasoning.
- **Planning**: The system uses its knowledge and reasoning capabilities to generate plans or strategies to achieve specific goals.
- **Execution**: The system puts its plans into action by interacting with the environment or other agents, and monitors the outcomes to update its knowledge.



https://www.edureka.co/blog/knowledge-representation-in-ai/

Knowledge Representation in Cognitive Science

Definition: Knowledge representation in cognitive science refers to the process of encoding and organizing information in a format that allows cognitive systems, such as humans or artificial intelligence, to reason, learn, and make decisions.

Selected Approaches:

- **Mental Models:** Mental models are internal representations of the world that individuals construct to understand and interact with their environment. These models capture knowledge about objects, concepts, relationships, and processes.
- **Concepts and Categories:** Concepts are mental representations that group objects, events, or ideas based on shared characteristics. Categories are formed by organizing concepts into hierarchical structures or networks, allowing for efficient cognitive processing.
- **Schemas and Scripts:** Schemas and scripts are cognitive frameworks that represent knowledge about typical patterns of events or situations. Schemas capture general knowledge structures, while scripts provide specific sequences of actions or events.
- **Semantic Networks:** Semantic networks represent knowledge using interconnected nodes and edges, where nodes represent concepts and edges represent relationships between concepts. This network structure enables efficient retrieval and inference of knowledge.
- **Frames and Prototypes:** Frames and prototypes represent knowledge by capturing the typical features and attributes associated with concepts. Frames provide structured representations with slots for specific properties, while prototypes define the central or typical characteristics of a concept.

Logical Representation

- **Propositional Logic**: Propositional logic is a type of logic that deals with propositions, which are statements that can be either true or false. Propositional logic uses logical connectives such as AND, OR, and NOT to combine propositions and form more complex statements. Propositional logic is used for a wide range of tasks, including natural language processing, expert systems, and automated theorem proving.
- **First-order Logic**: First-order logic, also known as predicate logic, is a more expressive type of logic that can deal with complex relationships among objects. It uses quantifiers such as "for all" and "there exists" to express generalizations and constraints on the objects in the domain. First-order logic is used for a wide range of tasks, including knowledge representation, reasoning, and natural language processing.
- **Higher-order Logic**: Higher-order logic is a type of logic that extends first-order logic by allowing functions and predicates to be quantified over. This allows for more expressive and flexible knowledge representation, as functions and predicates can be treated as first-class objects in the logic. Higher-order logic is used for a wide range of tasks, including formal verification, program synthesis, and automated reasoning.

Entailment Reasoning (Semantics, Models) vs. Inference (Syntax) in Propositional Logic

Entailment Reasoning:

- Entailment reasoning is based on the semantics of propositional logic.
- It involves evaluating the truth values of formulas in different models or interpretations.
- It focuses on the relationship between formulas and their truth values.
- It determines whether one formula logically entails another by examining all possible truth value assignments.
- It deals with the validity of the logical relationship between formulas.

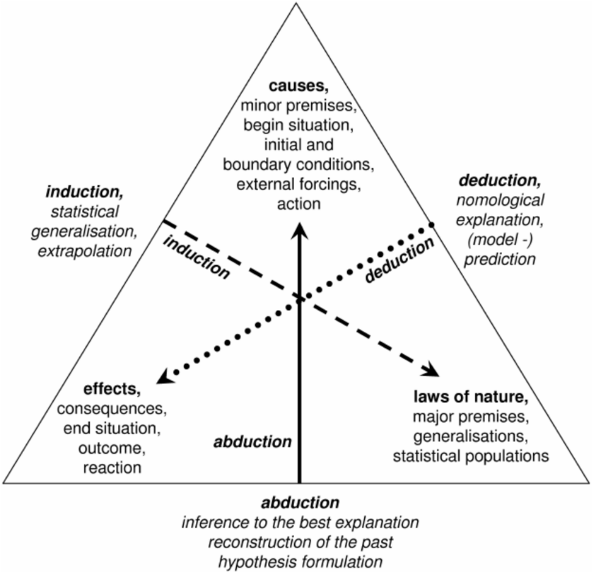
Inference:

- Inference refers to the process of deriving new formulas or conclusions based on existing formulas and logical rules.
- There are different types of inference in propositional logic:
    - **Abduction:** It involves generating explanations or hypotheses to account for given observations or facts. Abductive inference aims to find the most plausible explanation given the available evidence.
    - **Deduction:** It uses deductive reasoning rules, such as modus ponens and modus tollens, to derive valid conclusions from given premises. Deductive inference follows strict rules of logical implication.
    - **Induction:** It involves inferring general patterns or rules based on specific observations or instances. Inductive inference aims to generalize from specific cases to make probabilistic predictions or hypotheses.

Types of reasoning's: Deduction, Abduction, and Induction

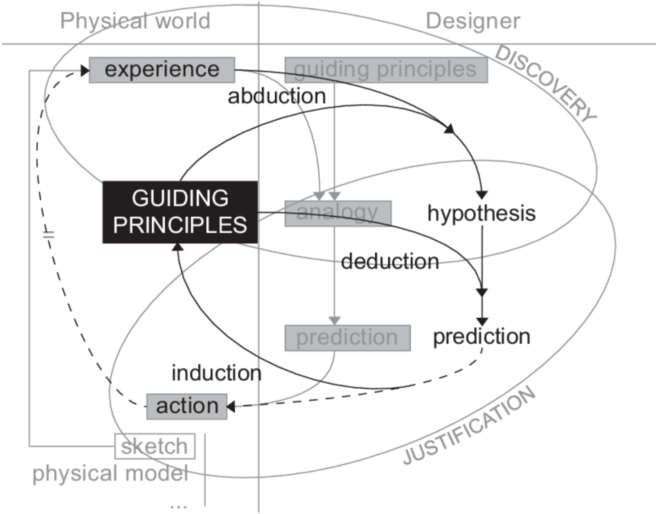
- **Deduction**: Deductive reasoning involves starting with a set of premises (known facts) and deriving a conclusion using logical rules of inference.
  - **Example**: given the premises "All humans are mortal" and "Socrates is human", we can deduce the conclusion "Socrates is mortal".
  - Deduction is used in applications such as expert systems and theorem proving.
- **Abduction**: Abductive reasoning involves starting with an observation or set of observations and generating a set of hypotheses that might explain the observations. The goal is to find the most likely explanation or set of explanations.
  - **Example**: if we observe smoke, we might generate the hypothesis "there is a fire".
  - Abduction is used in applications such as diagnosis and natural language understanding.
- **Induction**: Inductive reasoning involves starting with a set of observations and generating a general rule or hypothesis that explains the observations. The rule or hypothesis is not guaranteed to be true, but it is supported by the evidence.
  - **Example**: given a set of data points that represent the height and weight of a group of people, we might generate the hypothesis "taller people tend to weigh more".
  - Induction is used in applications such as machine learning and data mining.

Deduction, Abduction, and Induction (Cont.)



https://www.researchgate.net/publication/48326326

Deduction, Abduction, and Induction in Action



https://www.researchgate.net/publication/264595635

Examples of Other Types of Logics in AI

- **Modal Logic**: deals with modalities like necessity and possibility, and includes operators like $\Box$ (necessarily) and $\Diamond$ (possibly).
- **Temporal Logic**: deals with time and includes operators like X (next), F (eventually), and G (always).
- **Description logic**: a family of logics used in knowledge representation and reasoning in the semantic web
- **Epistemic logic**: deals with knowledge and belief of agents in a multi-agent system
- **Nonmonotonic Logic**: deals with reasoning under uncertainty and incomplete information, and includes default logic, circumscription, and autoepistemic logic.
- **Probabilistic Logic**: deals with uncertainty and includes Bayesian networks, Markov logic networks, and probabilistic soft logic.

Modal Logics in AI

- Modal logics are an extension of propositional logic that deal with modalities like necessity and possibility.
- In AI, modal logics are used to reason about knowledge, belief, obligation, and other modal notions.
- Modal logic includes operators like $\Box$ (necessarily) and $\Diamond$ (possibly) to represent modalities.
- A formula $\Box\phi$ is true in a model $\mathcal{M}$ if and only if ϕ is true in all worlds accessible from the current world in $\mathcal{M}$.
- A formula $\Diamond\phi$ is true in a model $\mathcal{M}$ if and only if ϕ is true in at least one world accessible from the current world in $\mathcal{M}$.

Modal Operators in AI

- Modal operators are used in AI to represent modalities like necessity and possibility.
- The $\Box$ operator is used to represent necessity, meaning that the statement inside the box must be true in all possible worlds.
- The $\Diamond$ operator is used to represent possibility, meaning that the statement inside the diamond may be true in at least one possible world.
- Modal operators can be combined to represent more complex modalities, such as $\Box\Diamond p$ (p is contingently necessary) or $\Diamond\Box p$ (p is possibly necessary).
- For example, $\Box(p \to q)$ means that in all accessible worlds if p is true, then q must be true.
- Another example is $\Diamond(p \land q)$, which means that it is possible for both p and q to be true in at least one accessible world.

Epistemic Logic

Overview:

- Epistemic logic is a branch of modal logic that formalizes reasoning about knowledge and beliefs in multi-agent systems.
- It provides a framework to model and reason about what agents know, what they believe, and how their knowledge and beliefs change over time.
- Epistemic logic plays a crucial role in designing intelligent systems and multi-agent systems by capturing and analyzing the properties of knowledge and beliefs.

Key Concepts:

- **Epistemic Operators**: Epistemic logic introduces operators such as K_i (agent i knows) and B_i (agent i believes) to represent different states of knowledge and belief of agents.
- **Modalities of Knowledge**: Epistemic logic deals with modalities of knowledge, such as *necessarily* (represented by $\Box$) and *possibly* (represented by $\Diamond$). For example, $K_i p$ means "Agent i knows that p is true," and $\Box p$ means "It is necessarily true that p."
- **Epistemic Models**: Epistemic logic employs epistemic models, which are structures that represent possible worlds, agents, and their knowledge and belief states. These models provide a formal semantics for interpreting and evaluating epistemic formulas.

Epistemic Logic (cont.)

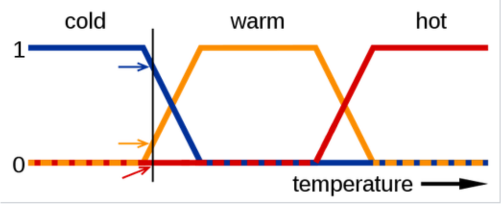
Epistemic logic provides a formal and AI-centric approach to modeling and reasoning about knowledge and beliefs, enabling the development of intelligent systems and multi-agent systems with advanced reasoning capabilities.

Applications in AI:

- **Multi-Agent Systems**: Epistemic logic is widely used in modeling and reasoning about knowledge and beliefs in multi-agent systems, enabling agents to reason about the knowledge and beliefs of other agents, communicate effectively, and make informed decisions.
- **Artificial Intelligence Planning**: Epistemic logic is applied in automated planning to represent and reason about knowledge and belief states of intelligent agents, helping them generate plans that account for uncertainty and changing knowledge.
- **Intelligent Agents**: Epistemic logic plays a crucial role in designing intelligent agents that reason about their own knowledge and beliefs, allowing them to exhibit sophisticated decision-making capabilities and adapt to dynamic environments.

Fuzzy Logic: Introduction

- **Fuzzy Set:** A fuzzy set is a set where elements have degrees of membership between 0 and 1. It allows for partial membership.
- **Membership Function:** A membership function determines the degree of membership of an element in a fuzzy set. It maps elements to membership values.
- **Fuzzy Logic:** Fuzzy logic is a form of multi-valued logic that deals with reasoning under uncertainty and imprecision. It allows for degrees of truth between true and false.
- **Fuzzy Rule:** A fuzzy rule defines the relationship between input variables and output variables in a fuzzy logic system. It consists of an antecedent and a consequent.
- **Fuzzy Inference:** Fuzzy inference is the process of deriving fuzzy outputs from fuzzy inputs using fuzzy rules and fuzzy reasoning mechanisms.



https://en.wikipedia.org/wiki/Fuzzy logic

Fuzzy Logic: Some Formal Definitions of Key Elements

- **Fuzzy Set:** Let X be a universal set and $\mu_A : X \to [0, 1]$ be a membership function of a fuzzy set A. The fuzzy set A is defined as $A = \{(x, \mu_A(x)) \mid x \in X\}$.
- **Fuzzy Intersection:** The fuzzy intersection of two fuzzy sets A and B is given by $A \cap B = \{(x, \min(\mu_A(x), \mu_B(x))) \mid x \in X\}$.
- **Fuzzy Union:** The fuzzy union of two fuzzy sets A and B is given by $A \cup B = \{(x, \max(\mu_A(x), \mu_B(x))) \mid x \in X\}$.
- **Fuzzy Complement:** The fuzzy complement of a fuzzy set A is given by $\bar{A} = \{(x, 1 - \mu_A(x)) \mid x \in X\}$.

Fuzzy Logic: Examples in AI

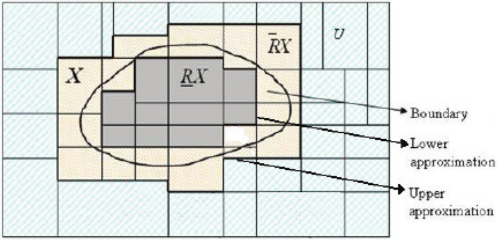
- **Fuzzy Control Systems:** Fuzzy logic is used in AI for designing fuzzy control systems. These systems model and control complex and imprecise processes by using fuzzy rules and linguistic variables.
- **Fuzzy Classification:** Fuzzy logic can be applied to classification tasks where instances may have multiple class memberships. Fuzzy classification assigns degrees of membership to different classes based on feature similarity.
- **Fuzzy Clustering:** Fuzzy logic is used in clustering algorithms to assign degrees of membership to data points for each cluster. Fuzzy clustering allows for partial membership of data points to multiple clusters.
- **Fuzzy Expert Systems:** Fuzzy logic is employed in expert systems to handle imprecise and uncertain knowledge. Fuzzy expert systems use fuzzy rules and fuzzy inference to make decisions and provide expert-like reasoning.

Rough Sets Theory: Introduction

- **Motivation:** In many real-world applications, data may be incomplete, inconsistent, or imprecise. Traditional methods struggle to handle such data effectively, leading to limited analysis and decision-making capabilities.
- **Key Idea:** The key idea behind Rough Sets theory is to characterize sets by their lower and upper approximations, which capture the definite and possible membership of objects, respectively.
- **Justification:** Rough Sets theory provides a mathematical framework to deal with uncertainty, vagueness, and incompleteness in data. It offers a flexible and intuitive approach for analyzing and reasoning about imperfect information.
- **Flexibility and Interpretability:** Rough Sets theory provides a flexible and interpretable framework, allowing domain experts to gain insights and make informed decisions based on uncertain or incomplete data.
- **Applications in AI methods and tools:** In the context of AI methods, Rough Sets theory finds applications in various tasks such as data preprocessing, feature selection, decision rule extraction, pattern recognition, and knowledge representation.
- **Domain and Systems Applications:** Rough Sets theory has been applied in fields such as data mining, expert systems, risk analysis, medical diagnosis, and information retrieval etc..

Rough Sets Theory (cont.)

- **Information System:** An information system is a collection of objects described by attributes. It consists of a set of objects and a set of attributes associated with these objects.
- **Indiscernibility Relation:** The indiscernibility relation defines the notion of similarity between objects in an information system. It groups together objects that have the same values for a given set of attributes.
- **Lower Approximation:** The lower approximation of a set approximates the set from below by including all objects that definitely belong to the set.
- **Upper Approximation:** The upper approximation of a set approximates the set from above by including all objects that possibly belong to the set.
- **Boundary Region:** The boundary region represents the objects that are neither definitely in nor definitely out of the set. It includes the objects in the difference between the upper and lower approximations.



https://www.researchgate.net/publication/261306027

Rule-Based Systems in AI

Definition:

- Rule-based systems, also known as production systems, are a fundamental AI technique used for knowledge representation and reasoning.
- These systems rely on a collection of **if-then** rules, where each rule consists of a condition (antecedent) and an action (consequent).
- When applied to a specific problem or situation, rule-based systems evaluate the conditions of the rules and execute the actions associated with the rules that match.

Components of a Rule-Based System:

- **Working Memory**: Represents the current state of the system and contains facts and data about the problem domain.
- **Rule Base**: Contains the collection of rules that define the knowledge and actions of the system.
- **Inference Engine**: Performs the matching of rules against the working memory and executes the actions of the matched rules.

Inference in Rule-Based Production System:

- The inference process involves applying the rules in the knowledge base to derive new conclusions or actions.

Knowledge Base and Inference in Rule-Based Production System

Knowledge Base (KB):

- A knowledge base is a repository of facts and rules that the rule-based production system uses for reasoning.
- **Facts**: Represent the current state of the system and are typically represented as logical statements or assertions.
- **Rules**: Capture the knowledge and logic of the problem domain and are written in the form of if-then statements.

Example Knowledge Base:

- **Facts**:
  - `WeatherIsSunny`
  - `TemperatureIsHigh`
- **Rules**:
  - `Rule 1: If WeatherIsSunny and TemperatureIsHigh, Then GoForSwimming.`
  - `Rule 2: If WeatherIsSunny, Then WearSunscreen.`

Argumentation Theory and AI

Argumentation theory is a branch of AI that focuses on modeling and analyzing reasoning and decision-making processes based on arguments. It provides a formal framework for constructing, evaluating, and reasoning with arguments to support or challenge claims.

Components of Argumentation:

- **Argument:** An argument is a set of statements, where one statement (the conclusion) is supported by a set of other statements (the premises).
- **Claim:** A claim is a statement that is proposed to be true or false.
- **Premise:** A premise is a statement that provides support or evidence for a claim.
- **Inference:** Inferences represent the logical steps used to derive conclusions from premises.

Argumentation Systems:

- **Formal Argumentation:** Formal argumentation frameworks, such as Dung's argumentation frameworks, provide a mathematical model for representing arguments and their relationships.
- **Assumption-Based Argumentation:** Assumption-based argumentation frameworks allow for reasoning with incomplete information by incorporating assumptions and identifying conflicts among them.
- **Probabilistic Argumentation:** Probabilistic argumentation frameworks introduce uncertainty into the argumentation process by assigning probabilities to arguments and evaluating their strengths based on probabilistic measures.

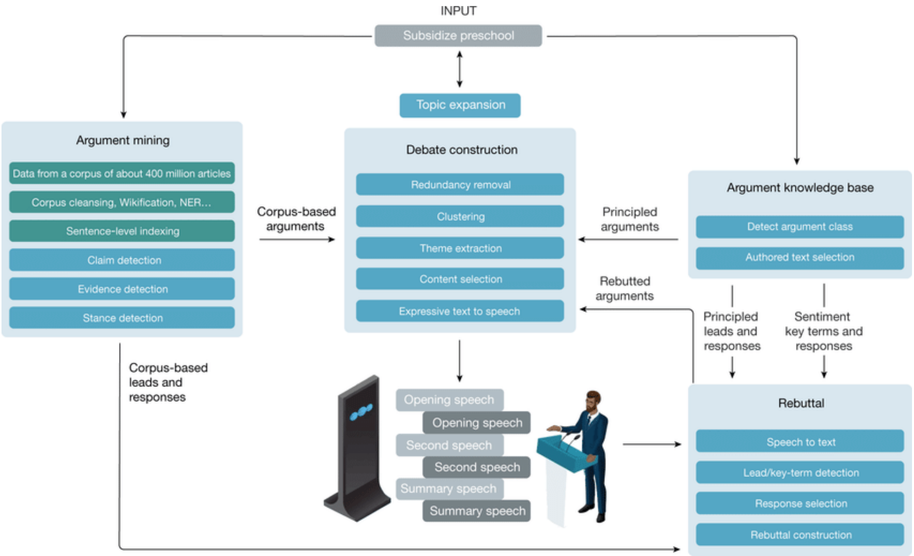
Argumentation Theory and AI (Cont'd)

Applications of Argumentation in AI:

- **Legal Reasoning:** Argumentation theory finds application in legal reasoning systems, where arguments are used to support legal claims, evaluate evidence, and assist in decision-making.
- **Dialogue Systems:** Argumentation-based dialogue systems enable natural language interactions by modeling and reasoning about arguments presented in a conversation. They help in reaching agreements, resolving conflicts, and providing explanations.
- **Decision Support Systems:** Argumentation theory plays a crucial role in decision support systems by providing a structured approach to analyze and evaluate different options, supporting decision-making processes.
- **Explainable AI:** Argumentation frameworks are used to explain the decision-making process of AI systems, providing justifications and evidence for the conclusions reached.
- **Semantic Web:** Argumentation theory is applied in the context of the Semantic Web to handle conflicts and inconsistencies among knowledge sources, enabling more reliable and coherent information integration.

Argumentation theory in AI provides a valuable framework to model, analyze, and reason with arguments, contributing to intelligent systems' decision-making, explainability, and interaction capabilities.

Argumentation - Debater architecture



https://www.researchgate.net/publication/350133414

Semantic Networks in AI

- Semantic networks are a knowledge representation technique used in AI.
- They represent knowledge in the form of interconnected nodes and edges.
- Nodes represent concepts or entities, while edges represent relationships between them.
- Semantic networks capture semantic associations and hierarchical relationships in a graphical structure.
- Example of a simple semantic network:

https://www.sciencedirect.com/science/article/abs/pii/S0957417419307122

Inference in Semantic Networks

Semantic networks are a type of knowledge representation that uses nodes and links to represent concepts and relationships between them. Inference in semantic networks involves using the network to answer queries or make predictions about new information based on the existing knowledge.

- **Forward Chaining**: Forward chaining is a type of inference that starts with the existing knowledge and uses it to make new conclusions. It involves traversing the semantic network from the starting nodes and following the links to derive new information.
- **Backward Chaining**: Backward chaining is a type of inference that starts with a goal or conclusion and works backward to find the necessary conditions that must be true to reach that conclusion. It involves traversing the semantic network in reverse from the conclusion and following the links to find the supporting information.
- **Constraint Satisfaction**: Constraint satisfaction is a type of inference that involves finding a solution that satisfies a set of constraints or rules. It involves applying the constraints to the semantic network and searching for a solution that meets all of the constraints.
- **Probabilistic Inference**: Probabilistic inference is a type of inference that involves assigning probabilities to different nodes or links in the semantic network based on the available evidence. It involves using probability theory to calculate the likelihood of different outcomes and make predictions about new information.

Frames in AI

- A frame is a data structure used to represent a collection of related concepts or objects.
- Frames provide a way to organize and represent knowledge in a structured way.
- Frames are used in AI to model complex concepts and relationships between them.

Structure and Elements

- Frames consist of slots and fillers.
- A slot is a named attribute of a frame, and a filler is the value associated with that slot.
- Slots and fillers can be hierarchical and can have default values.
- Frames can also have inheritance relationships, where child frames inherit slots and values from parent frames.

Example

- Let's say we want to represent the concept of a car in a frame.
- We might have slots for make, model, year, color, and price, with corresponding fillers.
- We could also have a hierarchical slot for engine, with sub-slots for horsepower, fuel type, and displacement.
- We could create a child frame for a specific make and model of car, which would inherit slots and fillers from the parent car frame.

Scripts in AI

Scripts in AI refer to structured representations of knowledge about stereotypical sequences of events or actions in a particular context. They provide a framework for understanding and reasoning about common situations and events.

Key Concepts:

- **Event Sequences:** Scripts capture the temporal order of events or actions that typically occur in a given scenario.
- **Roles and Participants:** Scripts define the roles or participants involved in the events and specify their expected behaviors and interactions.
- **Default Knowledge:** Scripts include default or stereotypical knowledge about how events unfold, including the preconditions, actions, and outcomes associated with each event.
- **Contextual Dependencies:** Scripts account for contextual information that influences the execution and interpretation of events.
- **Abstraction and Generalization:** Scripts allow for abstraction and generalization by capturing common patterns across different instances of the same event.

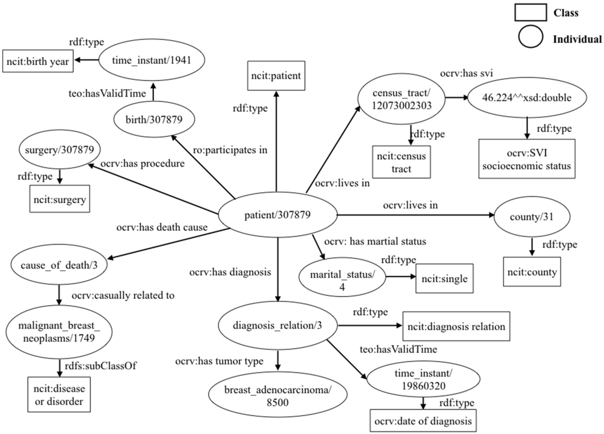
Examples:

- **Restaurant Script:** A script for dining out may include events such as entering the restaurant, ordering food, waiting for the meal, eating, and paying the bill.
- **Travel Script:** A script for air travel may include events such as booking a ticket, going through security, boarding the plane, taking off, in-flight service, landing, and collecting luggage.

The concept of scripts in AI was introduced by Roger Schank and his colleagues in the 1970s.

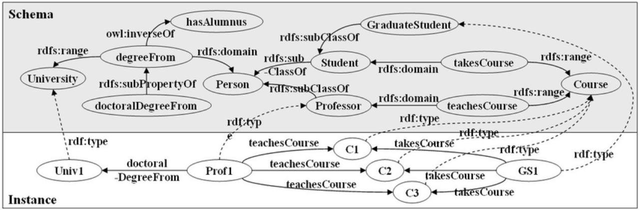
Ontologies in AI

- An ontology is a formal representation of knowledge that defines the concepts and relationships between them. It specifies a set of concepts and the categories, properties, and relations that hold among them.
- Ontologies are used to represent shared knowledge in a structured and machine-readable form (for an example as a graph). They could be also represented using description logics.



https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-018-0636-4

- Ontologies have many applications in AI, including:
  - Semantic search and information retrieval, where ontologies are used to enable more precise and meaningful search results.
  - Natural language processing, where ontologies are used to disambiguate the meaning of words and improve machine understanding of human language.
  - Knowledge management and sharing, where ontologies are used to standardize the representation of knowledge and facilitate sharing across different systems and domains.
- Ontologies are often represented using the Web Ontology Language (OWL), a semantic markup language that is designed for use on the World Wide Web.
- OWL provides a rich set of constructs for representing complex concepts and relationships, such as classes, properties, and restrictions.



https://www.researchgate.net/publication/220788186

Building an Ontology

Building an ontology involves a number of steps that are typically followed in a structured process. It is often an iterative process that involves refining and revising the ontology over time as new knowledge is acquired and the domain evolves.

- **Identify the domain**: The first step is to identify the domain that the ontology will represent. This may involve analyzing existing knowledge sources, consulting with domain experts, and defining the scope and purpose of the ontology.
- **Gather the knowledge**: The next step is to gather the knowledge that will be used to build the ontology. This may involve reviewing existing literature, analyzing data sources, and consulting with experts.
- **Define the concepts**: Once the knowledge has been gathered, the next step is to define the concepts that will be used in the ontology. This involves identifying the key categories and relationships that are relevant to the domain.
- **Specify the relationships**: After the concepts have been defined, the next step is to specify the relationships between them. This involves defining the types of relationships that exist between the concepts, such as part-of, is-a, and has-property.
- **Formalize the ontology**: The final step is to formalize the ontology using a standard ontology language, such as OWL or RDF. This involves specifying the classes, properties, and axioms that make up the ontology, and creating a structured representation of the knowledge in the domain.

Inference with Ontologies

Ontologies are used in AI to represent knowledge in a structured way, and to support reasoning and inference over that knowledge. Some common techniques for performing inference with ontologies include:

- **Classification**: Ontologies define a set of classes and subclass relationships between them. Classification involves determining which class or classes a given instance belongs to based on its properties and relationships. For example, if an ontology includes classes for "Animal", "Mammal", and "Cat", an instance of a domestic cat would be classified as both a "Mammal" and a "Cat".
- **Reasoning**: Ontologies include a set of axioms and rules that define the relationships between concepts. Reasoning involves applying those rules to derive new knowledge from existing knowledge. For example, if an ontology includes the axiom that "All mammals have hair" and the fact that "Cats are mammals", it can be inferred that "Cats have hair".
- **Querying**: Ontologies can be queried to retrieve information about the concepts and relationships defined in the ontology. Querying involves constructing a query in a query language such as SPARQL or SQL that retrieves the desired information. For example, a query might ask for all instances of the class "Person" that have the property "hasAge" greater than 18.
- **Validation**: Ontologies can be used to validate the consistency and completeness of a set of knowledge. Validation involves checking that the axioms and rules in the ontology do not lead to contradictions or inconsistencies, and that all necessary information is present in the ontology.

Symbolic vs. Subsymbolic AI

Introduction

Symbolic AI:

- Symbolic AI, also known as classical AI or GOFAI (Good Old-Fashioned AI), represents knowledge using symbols, rules, and logical operations.
- It relies on formal logic and symbolic manipulation to perform reasoning and decision-making.
- Knowledge is explicitly represented, and the focus is on the manipulation and inference of these symbolic representations.

Subsymbolic AI:

- Subsymbolic AI, also known as connectionist AI or neural network-based AI, represents knowledge through the activation patterns and weights of interconnected artificial neurons.
- It is based on the principles of distributed processing and learning from examples.
- Knowledge is implicitly represented in the connection weights and activation states of the network, and computation occurs through parallel distributed processing.

Symbolic vs. Subsymbolic AI

Characteristics

Symbolic AI:

- **Explicit knowledge representation:** Knowledge is represented explicitly using symbols, rules, and logic.
- **Rule-based reasoning:** Reasoning is based on applying logical rules and making deductions.
- **Transparency and interpretability:** Symbolic representations allow for human understanding and interpretation of the system's decision-making process.
- **Limited scalability:** Symbolic AI may struggle with large-scale knowledge representation and handling uncertainty.

Subsymbolic AI:

- **Implicit knowledge representation:** Knowledge is represented implicitly through distributed activation patterns and connection weights.
- **Pattern recognition and learning:** Learning occurs through exposure to examples and the adjustment of connection weights.
- **Parallel distributed processing:** Computation happens in parallel across interconnected artificial neurons.
- **Scalability and adaptability:** Subsymbolic AI systems can handle large-scale data and adapt to new situations through learning.

Hybrid Approaches:

- Hybrid AI approaches combine the strengths of symbolic and subsymbolic AI to overcome their limitations.
- By integrating symbolic and subsymbolic techniques, these approaches aim to achieve more robust and flexible AI systems.
- Examples of hybrid approaches include:
  - **Connectionist-Symbolic Integration:** Combining neural networks for pattern recognition with symbolic reasoning and logic-based decision-making.
  - **Knowledge-Guided Learning:** Using symbolic knowledge to guide and constrain the learning process in subsymbolic AI systems.
  - **Integrated Architectures:** Creating AI systems that employ both symbolic and subsymbolic components, with interactions and feedback between them.

Benefits of Hybrid Approaches:

- Enhanced reasoning and learning capabilities by leveraging the strengths of both approaches.
- Improved handling of uncertainty, complex problems, and real-world data.
- Increased interpretability and explainability while maintaining computational efficiency.

Symbolic vs. Subsymbolic AI
Examples of Methods

Symbolic AI:

- Logic-based reasoning (e.g., propositional logic, first-order logic)
- Expert systems and rule-based systems
- Knowledge representation using formal languages (e.g., ontologies, semantic networks)
- Symbolic planning and decision-making algorithms
- Bayesian networks for probabilistic reasoning

Subsymbolic AI:

- Artificial neural networks (e.g., feedforward networks, recurrent networks)
- Deep learning models (e.g., convolutional neural networks, recurrent neural networks)
- Support vector machines and kernel methods
- Genetic algorithms and evolutionary computation
- Reinforcement learning algorithms (e.g. Markov models)

Summary - Knowledge Representation

1. Introduction to Knowledge representation in AI
   - Some views on Knowledge Representation
2. Knowledge Representation in Logic
   - Types of reasoning in logic
   - Modal Logics in AI
   - Epistemic Logic
   - ...
3. Selected Other Symbolic Methods
   - Rough Sets
   - Rule-based Systems
   - Argumentation
   - Semantic Networks
   - Ontologies
   - ...
4. Symbolic vs. Subsymbolic
   - Key characteristics
   - Examples
5. Subsymbolic Methods
   - Artificial Neural Networks and Deep Learning
   - Genetic and evolutionary algorithms
   - Reinforcement learning
   - ...

References

1. S. J. Russell, P. Norvig, Artificial Intelligence: A Modern Approach", Financial Times Prentice Hall, 2019.
2. M. Flasiński, Introduction to Artificial Intelligence", Springer Verlang, 2016
3. M. Muraszkiewicz, R. Nowak (ed.), Sztuczna Inteligencja dla inżynierów", Oficyna Wydawnicza PW, 2022
4. J. Prateek , Artificial Intelligence with Python", Packt 2017