Universiti Malaysia PAHANG
Engineering • Technology • Creativity

# CHAPTER 3
# Knowledge Representation System

ARTIFICIAL INTELLIGENCE (BSD2513)

DR. KU MUHAMMAD NA'IM KU KHALIF

5-STAR WORLD CLASS TECHNOLOGICAL UNIVERSITY

UMPMalaysia

# Content

Chapter 3.1 Concepts of Knowledge Representation System

Chapter 3.2 Logic and Inference

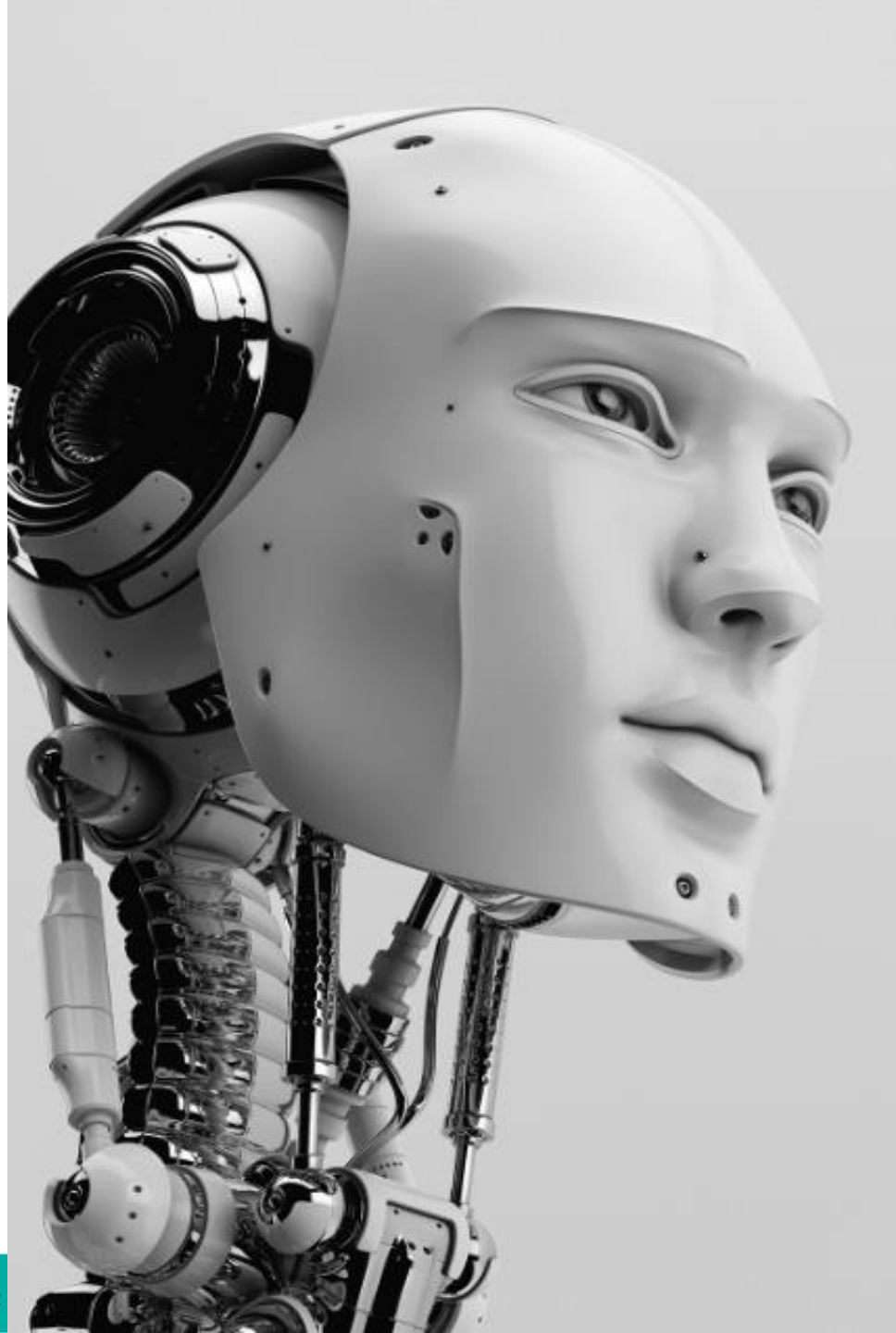
Chapter 3.3 Rule Based Expert System

Chapter 3.4 Case Based Reasoning System

# Chapter 3.1: Concepts of Knowledge Representation System

By the end of this topic, you should be able to:

- understand the concepts knowledge representation system in artificial intelligence.
- understand the needs of knowledge representation system.



# An Overview

- Humans are good at understanding, reasoning, and interpreting knowledge.
- Human knows things, which is knowledge and as per their knowledge they perform various actions in the real world. But how machines do all these things comes under knowledge representation and reasoning.
- No intelligent without knowledge representation system.

## What is Knowledge Representation?

- Knowledge Representation in AI describes the representation of knowledge.
- Basically, it is a study of how the beliefs, intentions, and judgments of an intelligent agent can be expressed suitably for automated reasoning. One of the primary purposes of knowledge representation includes modeling intelligent behavior for an agent.
- Knowledge Representation and Reasoning (KR, KRR) represents information from the real world for a computer to understand and then utilize this knowledge to solve complex real-life problems like communicating with human beings in natural language.
- Knowledge Representation in AI is not just about storing data in a database, it allows a machine to learn from that knowledge and behave intelligently like a human being.

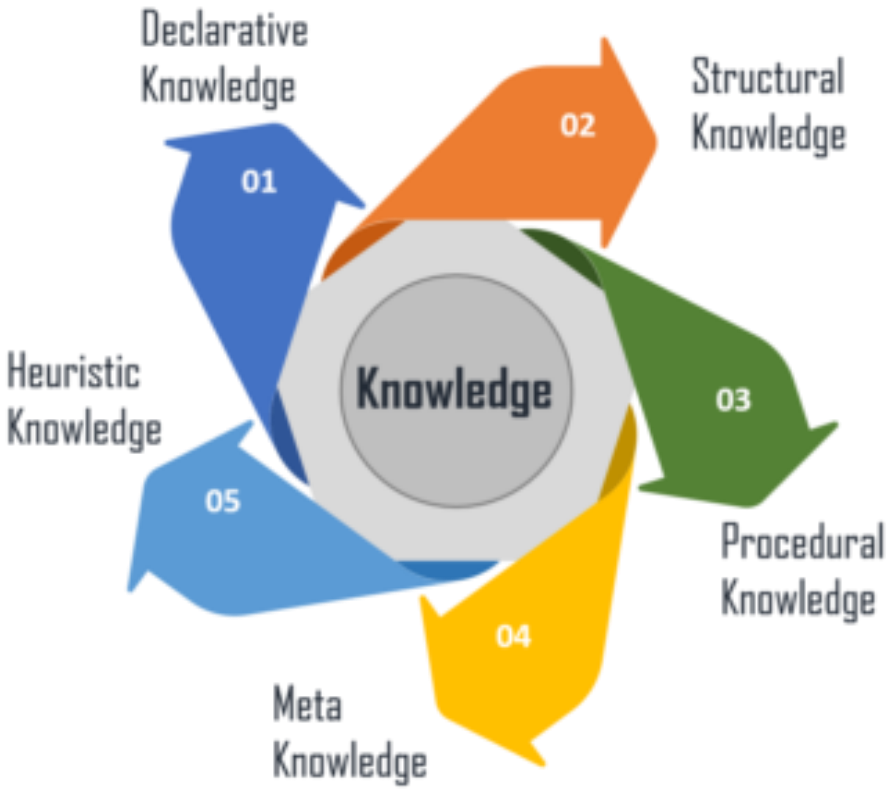



**Figure 3.1:** Types of knowledge.

Declarative Knowledge – It includes concepts, facts, and objects and expressed in a declarative sentence.

Structural Knowledge – It is a basic problem-solving knowledge that describes the relationship between concepts and objects.

Procedural Knowledge – This is responsible for knowing how to do something and includes rules, strategies, procedures, etc.

Meta Knowledge – Meta Knowledge defines knowledge about other types of Knowledge.

Heuristic Knowledge – This represents some expert knowledge in the field or subject.

# Cycle of Knowledge Representation in AI

- Artificial intelligent systems usually consist of various components to display their intelligent behavior.
- Some of these components include:
    1. Perception
    2. Learning
    3. Knowledge Representation & Reasoning
    4. Planning
    5. Execution

- Here is an example to show the different components of the system and how it works:

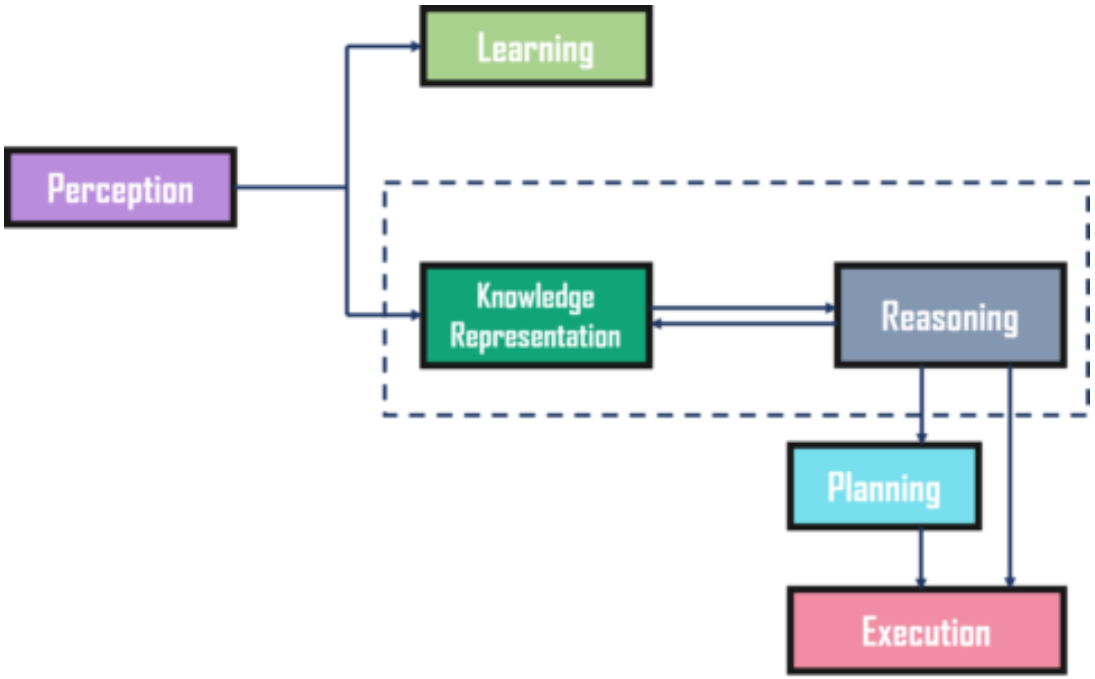



**Figure 3.2:** Different components of the system.

- The above diagram shows the interaction of an AI system with the real world and the components involved in showing intelligence.

- The **Perception Component** retrieves data or information from the environment. with the help of this component, you can retrieve data from the environment, find out the source of noises and check if the AI was damaged by anything. Also, it defines how to respond when any sense has been detected.
- Then, there is the **Learning Component** that learns from the captured data by the perception component. The goal is to build computers that can be taught instead of programming them. Learning focuses on the process of self-improvement. In order to learn new things, the system requires knowledge acquisition, inference, acquisition of heuristics, faster searches, etc.

- The main component in the cycle is **Knowledge Representation and Reasoning** which shows the human-like intelligence in the machines. Knowledge representation is all about understanding intelligence. Instead of trying to understand or build brains from the bottom up, its goal is to understand and build intelligent behavior from the top-down and focus on what an agent needs to know in order to behave intelligently. Also, it defines how automated reasoning procedures can make this knowledge available as needed.
- The **Planning and Execution** components depend on the analysis of knowledge representation and reasoning. Here, planning includes giving an initial state, finding their preconditions and effects, and a sequence of actions to achieve a state in which a particular goal holds. Now once the planning is completed, the final stage is the execution of the entire process.

# What is the Relation between Knowledge & Intelligence?

- In the real world, knowledge plays a vital role in intelligence as well as creating artificial intelligence. It demonstrates the intelligent behavior in AI agents or systems. It is possible for an agent or system to act accurately on some input only when it has the knowledge or experience about the input.

- Let’s take an example to understand the relationship:

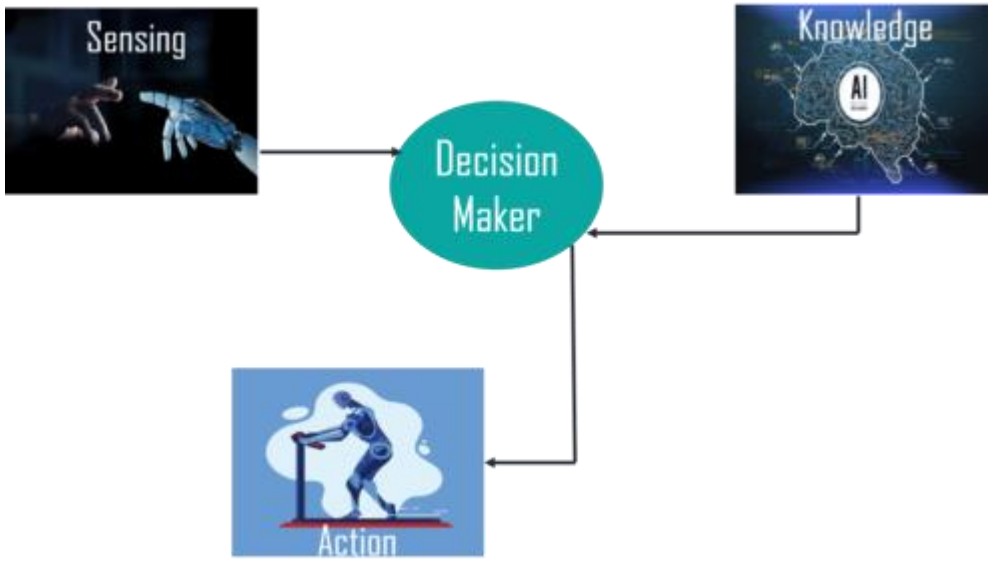



**Figure 3.3:** Decision maker’s relationship.

- In this example, there is one decision-maker whose actions are justified by sensing the environment and using knowledge. But, if we remove the knowledge part here, it will not be able to display any intelligent behavior.

# Techniques of Knowledge Representation in AI

- There are four techniques of representing knowledge such as:


**Figure 3.4:** Techniques of Knowledge Representation in AI**.**

# Logical Representation

- Logical representation is a language with some definite rules which deal with propositions and has no ambiguity in representation. It represents a conclusion based on various conditions and lays down some important communication rules. Also, it consists of precisely defined syntax and semantics which supports the sound inference. Each sentence can be translated into logics using syntax and semantics.

| Syntax | Semantics |
| --- | --- |
| • It decides how we can construct legal sentences in logic.<br>• It determines which symbol we can use in knowledge representation.<br>• Also, how to write those symbols. | • Semantics are the rules by which we can interpret the sentence in the logic.<br>• It assigns a meaning to each sentence. |

Advantages:

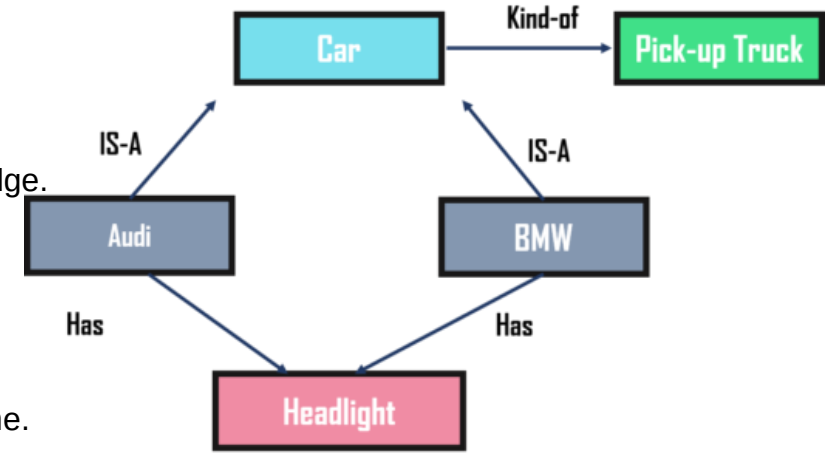
- Logical representation helps to perform logical reasoning.
- This representation is the basis for the programming languages.

Disadvantages:

- Logical representations have some restrictions and are challenging to work with.
- This technique may not be very natural, and inference may not be very efficient.

# Semantic Network Representation

- Semantic networks work as an alternative of predicate logic for knowledge representation. In Semantic networks, you can represent your knowledge in the form of graphical networks. This network consists of nodes representing objects and arcs which describe the relationship between those objects. Also, it categorizes the object in different forms and links those objects.
- This representation consist of two types of relations:
  1. IS-A relation (Inheritance)
  2. Kind-of-relation




Advantages:

- Semantic networks are a natural representation of knowledge.
- Also, it conveys meaning in a transparent manner.
- These networks are simple and easy to understand.

Disadvantages:

- Semantic networks take more computational time at runtime.
- Also, these are inadequate as they do not have any equivalent quantifiers.
- These networks are not intelligent and depend on the creator of the system.

# Frame Representation

- A frame is a record like structure that consists of a collection of attributes and values to describe an entity in the world. These are the AI data structure that divides knowledge into substructures by representing stereotypes situations.
- Basically, it consists of a collection of slots and slot values of any type and size. Slots have names and values which are called facets.

Advantages:

- It makes the programming easier by grouping the related data.
- Frame representation is easy to understand and visualize.
- It is very easy to add slots for new attributes and relations.
- Also, it is easy to include default data and search for missing values.

Disadvantages:

- In frame system inference, the mechanism cannot be easily processed.
- The inference mechanism cannot be smoothly proceeded by frame representation.
- It has a very generalized approach.

# Production Rules

- In production rules, agent checks for the condition and if the condition exists then production rule fires and corresponding action is carried out. The condition part of the rule determines which rule may be applied to a problem. Whereas, the action part carries out the associated problem-solving steps. This complete process is called a recognize-act cycle.
- The production rules system consists of three main parts:
    1. The set of production rules
    2. Working Memory
    3. The recognize-act-cycle

Advantages:

- The production rules are expressed in natural language.
- The production rules are highly modular and can be easily removed or modified.

Disadvantages:

- It does not exhibit any learning capabilities and does not store the result of the problem for future uses.
- During the execution of the program, many rules may be active. Thus, rule-based production systems are inefficient.

# Representation Requirements

A good knowledge representation system must have properties such as:

- Representational Accuracy: It should represent all kinds of required knowledge.
- Inferential Adequacy: It should be able to manipulate the representational structures to produce new knowledge corresponding to the existing structure.
- Inferential Efficiency: The ability to direct the inferential knowledge mechanism into the most productive directions by storing appropriate guides.
- Acquisitional efficiency: The ability to acquire new knowledge easily using automatic methods.

# Approaches to Knowledge Representation in AI

There are different approaches to knowledge representation such as:

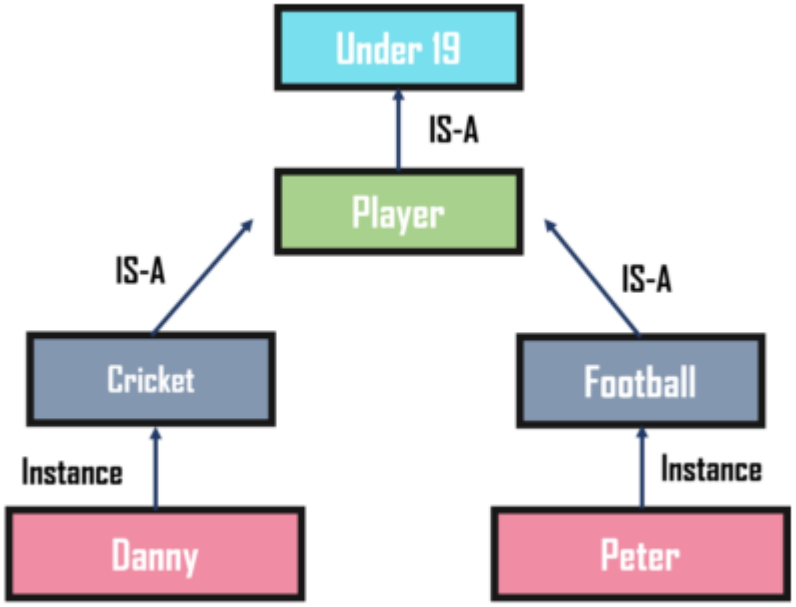
**Simple Relational Knowledge**

It is the simplest way of storing facts which uses the relational method. Here, all the facts about a set of the object are set out systematically in columns. Also, this approach of knowledge representation is famous in database systems where the relationship between different entities is represented. Thus, there is little opportunity for inference.

| Name | Age | Emp ID |
|---|---|---|
| John | 25 | 100071 |
| Amanda | 23 | 100056 |
| Sam | 27 | 100042 |

## Inheritable Knowledge

In the inheritable knowledge approach, all data must be stored into a hierarchy of classes and should be arranged in a generalized form or a hierarchal manner. Also, this approach contains inheritable knowledge which shows a relation between instance and class, and it is called instance relation. In this approach, objects and values are represented in Boxed nodes.



## Inferential Knowledge

The inferential knowledge approach represents knowledge in the form of formal logic. Thus, it can be used to derive more facts. Also, it guarantees correctness.

**Statement 1**: John is a cricketer.

**Statement 2**: All cricketers are athletes.

Then it can be represented as;

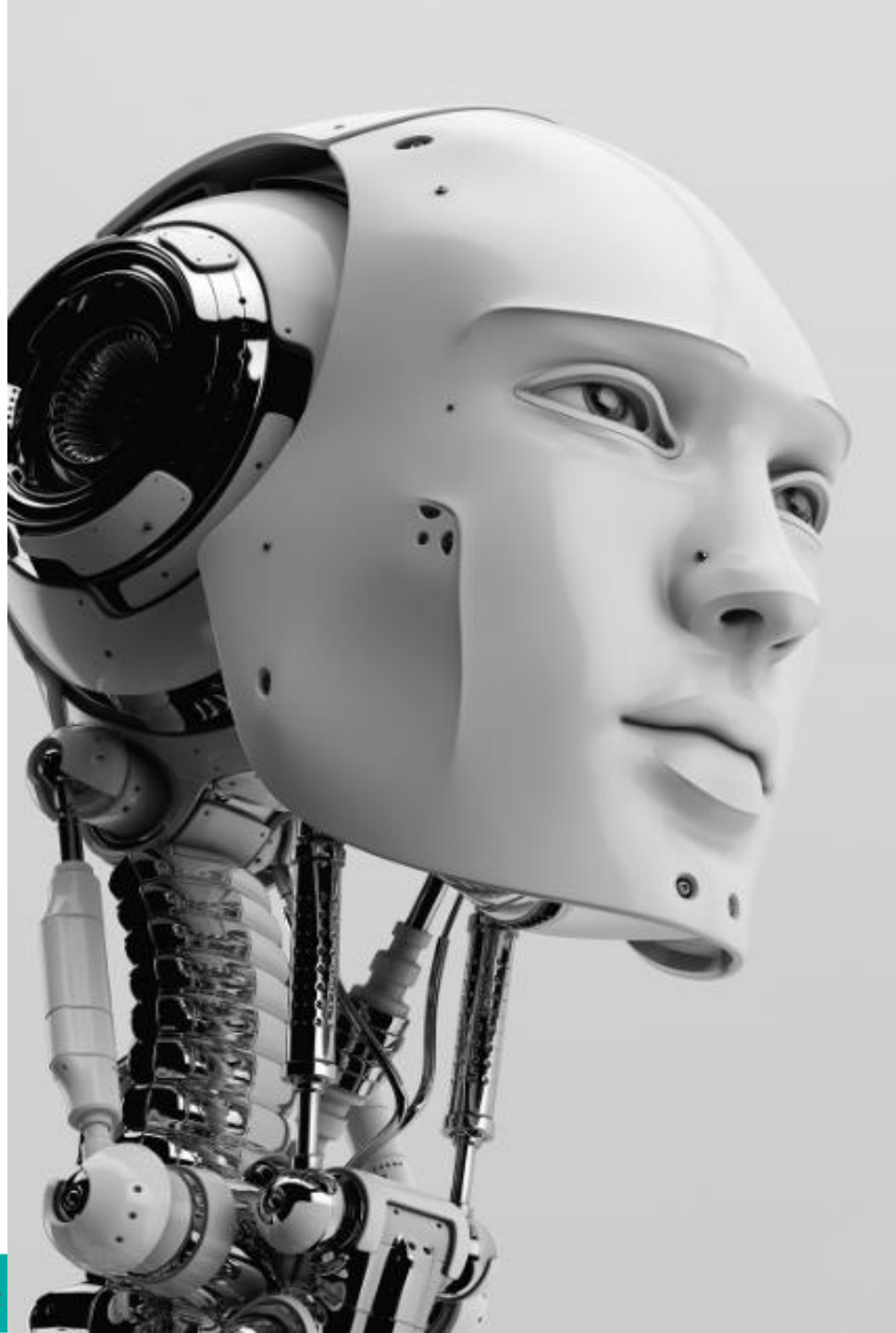
**Cricketer(John)**

**∀x = Cricketer (x) ———-> Athelete (x)s**

# Chapter 3.2:
# Logic and Inference

By the end of this topic, you should be able to:

- understand the concepts logic in artificial intelligence.
- understand the needs of logic and inference in AI.
- understand the proposional logic and first order logic



# Logic is Everywhere

- Logic has been called the *calculus of computer science*, because logic plays a fundamental role in computer science, starting from the construction of computers, to the computing devices beyond our ability of construction, such as computer architecture (digital gates, hardware verification), software engineering (specification, verification), programming languages (semantics, type theory, abstract data types, object-oriented programming), databases (relational algebra), artificial intelligence (automated theorem proving, knowledge representation), algorithms and theory of computation (complexity, computability), etc.
- Logic also plays important roles in other fields, such as mathematics and philosophy. In mathematics, logic includes both the mathematical study of logic and the applications of formal logic to other areas of mathematics.

- In artificial intelligence, we need intelligent computers which can create new logic from old logic or by evidence, so generating the conclusions from evidence and facts is termed as Inference. Inference rules are the templates for generating valid arguments. Inference rules are applied to derive proofs in artificial intelligence, and the proof is a sequence of the conclusion that leads to the desired goal.
- The unifying themes in mathematical logic include the study of the expressive power of formal systems and the deductive power of formal proof systems.
- Fields of logic: a) Set theory, b) Model theory, c) Proof theory, d) Computability theory
- Formal logic - formal logic have tended to be termed either philosophy of logic or philosophical logic.
- Philosophical logic dealt with arguments in the natural language used by humans.

  Example. Given the premises that (a) “all men are mortal” and (b)“Socrates is a man”, we may draw the conclusion that (c) “Socrates is mortal”, by the inference rule of Modus Ponens.

# Statement or Proposition

- Logic comes from natural languages as most sentences in natural languages are statements or propositions.
- A proposition is a statement or assertion that expresses a judgment or opinion. Here statement, assertion, and proposition are synonyms.
- Here are some examples:
  - That apple is delicious.
  - Today is either Monday or Tuesday.
  - He has good memory and runs fast.
- Every statement can be either true or false. True or false are called the truth values of a statement. Some sentences in natural languages are not statements, such as commands or exclamatory sentences.

For example, “Run fast!” Coach shouts. The first part of the sentence, “Run fast!”, is not a statement; the second part, “Coach shouts”, is a statement. In fact, a sentence is a statement if and only if it has a truth value.

- The unifying themes in mathematical logic include the study of the expressive power of formal systems and the deductive power of formal proof systems.
- Fields of logic: a) Set theory, b) Model theory, c) Proof theory, d) Computability theory
- Formal logic - formal logic have tended to be termed either philosophy of logic or philosophical logic.
- Philosophical logic dealt with arguments in the natural language used by humans.

  Example. Given the premises that (a) “all men are mortal” and (b)“Socrates is a man”, we may draw the conclusion that (c) “Socrates is mortal”, by the inference rule of Modus Ponens.

- In natural languages, we can combine, or relate statements with words such as “not” (negation), “and” (conjunction), “or” (disjunction), “implies” (implication), “if-then”, etc. That is, a statement can be obtained from other statements by these words. In logic, these words are called logical operators, or equivalently, logical connectives.
- A statement is composed if it can be expressed as a composition of several simpler statements; otherwise it is simple. In the above three examples of statements, the first statement is simple, but the other two are composed. That is, “today is either Monday or Tuesday” is the composition of “today is Monday” and “today is Tuesday”, using the logical operator “or”. We often regard “that apple is not delicious” as a composed statement: It is the negation of a simple statement.
- Logical connectives are indispensable for expressing the relationship between statements.

- For example, the following statement is a composition of several simple statements.

*Since I don't have a good memory, I don't know that today is either Monday or Tuesday.

- To see this clearly, we need to introduce symbols, such as p, q, r, s, and t, to denote them:

1. *p*: I have a good memory.
2. *q*: Today is Monday.
3. *r*: Today is Tuesday.
4. *s*: q or r.
5. *t*: I know s.

To express that the statement (*) is composed, let us use these symbols for logical operators: : ¬ for negation, ∨ for "OR", ∧ for "AND", and → for implication.

Then (*) becomes the following formula:

$$(\neg p) \rightarrow (\neg t)$$

- Here each of the symbols *p*, *q*, *r*, *s*, and t is called a propositional variable, which denotes a statement, either simple or composed. Naturally, these propositional variables, like statements, can take on only the truth values, i.e., true and false.
- Propositional variables are also called Boolean variables after their inventor, the nineteenth century mathematician George Boole.
- Boolean logic includes any logic in which the considered truth values are true and false. The study of propositional variables with logic operators is called Propositional logic, which is the most simple one in the family of Boolean logics.

# Proposional Logic

- In artificial intelligence, we need intelligent computers which can create new logic from old logic or by evidence, so generating the conclusions from evidence and facts is termed as Inference.

- Inference rules are the templates for generating valid arguments. Inference rules are applied to derive proofs in artificial intelligence, and the proof is a sequence of the conclusion that leads to the desired goal.

- For example, the following statement is a composition of several simple statements.

*Since I don't have a good memory, I don't know that today is either Monday or Tuesday.

- To see this clearly, we need to introduce symbols, such as p, q, r, s, and t, to denote them:

1. *p*: I have a good memory.
2. *q*: Today is Monday.
3. *r*: Today is Tuesday.
4. *s*: q or r.
5. *t*: I know s.

To express that the statement (*) is composed, let us use these symbols for logical operators: : ¬ for negation, ∨ for "OR", ∧ for "AND", and → for implication.

Then (*) becomes the following formula:

$$(\neg p) \rightarrow (\neg t)$$

# First Order Logic

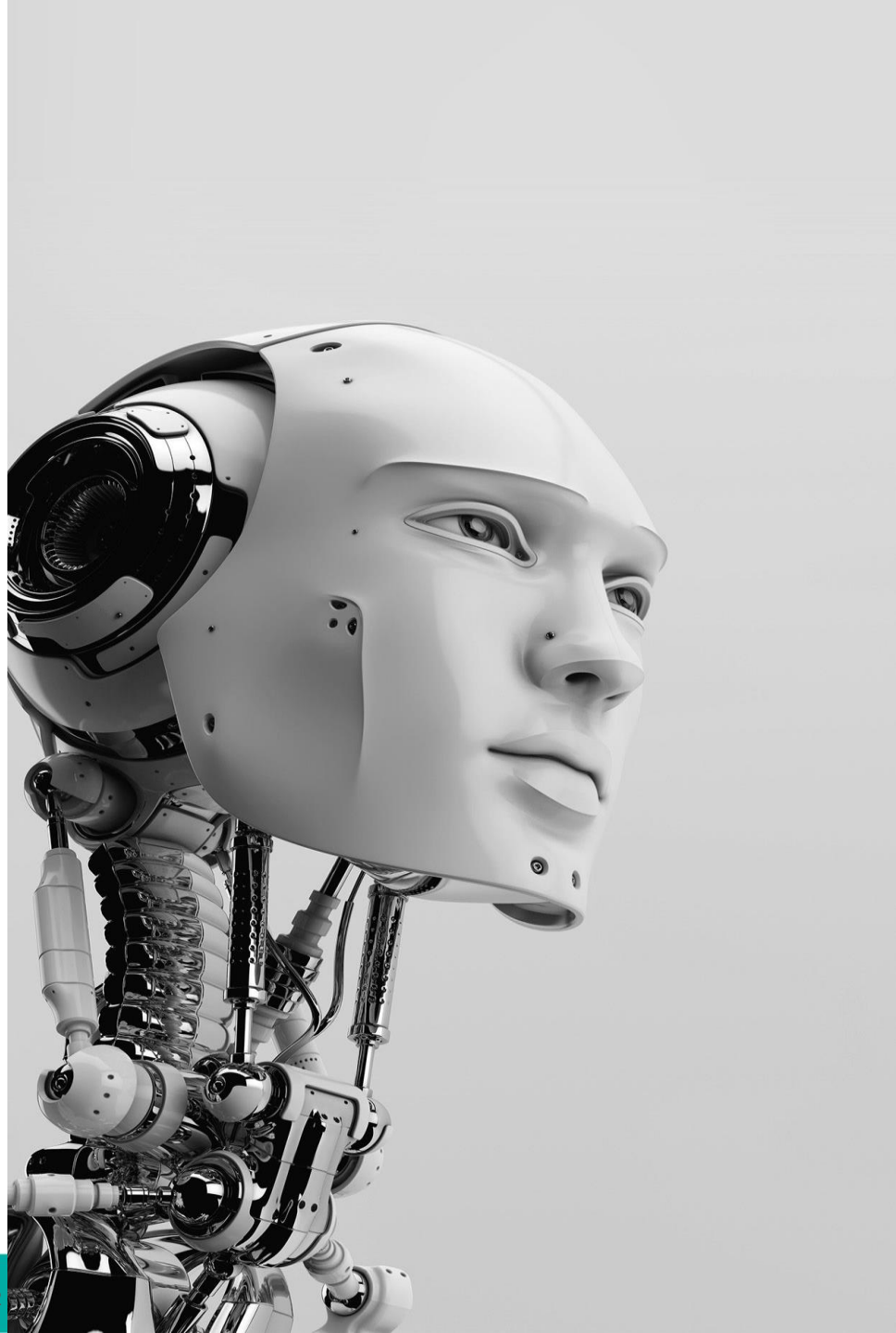
- Propositional logic provides a good start at describing the general principles of logical reasoning, but it is not possible to express general properties of many important sets, especially when a set is infinite. For example, how to specify that every natural number is either even or odd, a natural number is even iff its successor is odd, or the relation < is transitive over the natural numbers?
- The weak expressive power of propositional logic accounts for its relative mathematical simplicity, but it is a very severe limitation, and it is desirable to have more expressive logics. Mathematics and some other disciplines such as computer science often consider sets of elements in which certain relations and operations are singled out.

- When using the language of propositional logic, our ability to talk about the properties of such relations and operations is very limited. Thus, it is necessary to refine the logic language, in order to increase the expressive power of logic. These are exactly what is provided by a more expressive logical framework known as first order logic, which will be the topic of the next three chapters. First-order logic is a considerably richer logic than propositional logic, but yet enjoys many nice mathematical properties.
- In first-order logic, many interesting and important properties about various sets can be expressed. Technically, this is achieved by allowing the propositional symbols to have arguments ranging over elements of sets, to express the relations and operations in question. These propositional symbols are called predicate.
- This is the reason that first-order logic is often called predicate calculus.

# Chapter 3.3: Rule-Based Expert System

By the end of this topic, you should be able to:

- understand the concepts of rule-based expert system in artificial intelligence.
- understand the step process for CBR
- know the advantages and disadvantages of CBR.



# Rule-Based Expert System

- A rule-based expert system is the simplest form of artificial intelligence and uses prescribed knowledge-based rules to solve a problem.

- The aim of the expert system is to take knowledge from a human expert and convert this into a number of hardcoded rules to apply to the input data. In their most basic form, the rules are commonly conditional statements (if a, then do x, else if b, then do y).

- These systems should be applied to smaller problems, as the more complex a system is, the more rules that are required to describe it, and thus increased difficulty to model for all possible outcomes.

- Rule-based systems (also known as production systems or expert systems) are the simplest form of artificial intelligence. A rule based system uses rules as the knowledge representation for knowledge coded into the system (Springer).
- **Expert systems**, which are system that mimic the reasoning of human expert in solving a knowledge intensive problem.
- Instead of representing knowledge in a declarative, static way as a set of things which are true, rule-based system represent knowledge in terms of a set of rules that tells what to do or what to conclude in different situations.
- Encoding a human expert's knowledge in a fairly narrow area into an automated system.
- Some applications: IKEA Virtual Assistant, Diagnostics Oriented Rockwell Intelligence System (DORIS), Machine for Intelligent Diagnosis (MIND).

- A rule-based system can be simply created by using a set of assertions and a set of rules that specify how to act on the assertion set. Rules are expressed as a set of if-then statements (called IF-THEN rules or production rules):

  IF P THEN Q

  which is also equivalent to:

  P=>Q

- A rule-based system consists of a set of IF-THEN rules, a set of facts and some interpreter controlling the application of the rules, given the facts. The idea of an expert system is to use the knowledge from an expert system and to encode it into a set of rules. When exposed to the same data, the expert system will perform in a similar manner to the expert.
- The requirement is that the knowledge on the problem area can be expressed in the form of if-then rules. The area should also not be that large because a high number of rules can make the problem solver inefficient.

## FEATURES OF RULE-BASED SYSTEMS

Widely used in Artificial Intelligence, Rule-Based Expert System is not just only responsible for modeling intelligent behavior in machines and building expert system that outperform human expert(s) but also helps:

- Composed of combined knowledge of human experts in the problem domain.
- Represent knowledge in a highly declarative way.
- Enables the use of several different knowledge representations paradigms.
- Supports implementation of non-deterministic search and control strategies.
- It helps describe fragmentary, ill-structured, heuristic, judgmental knowledge.
- It is robust and can operate with uncertain or incomplete knowledge.
- Helps with rule-based decision making examples monitoring, control, diagnostics, service, etc.

## Elements of a Rule-Based System

Any rule-based system consists of a few basic and simple elements as follows:

1. Set of facts. These facts are actually the assertions and should be anything relevant to the beginning state of the system.
2. Set of rules. This contains all actions that should be taken within the scope of a problem specify how to act on the assertion set. A rule relates the facts in the IF part to some action in the THEN part. The system should contain only relevant rules and avoid the irrelevant ones because the number of rules in the system will affect its performance.
3. Termination criterion/ Interpreter. This is a condition that determines that a solution has been found or that none exists. This is necessary to terminate some rule-based systems that find themselves in infinite loops otherwise.

| DATA | CONDITIONS | RULES |
|---|---|---|
| **Season winter**<br>temperature<br>wind blushing<br>road<br>weather | <0, >0<br>strongly, gently<br>slippery, not slippery<br>cold, warm, hot | ***Premises***<br>IF temperature<0<br>AND<br>IF wind blushing is strongly<br>OR<br>IF the road is slippery<br>---<br>***Conclusion***<br>THEN the weather is cold |

**Figure 3.4:** An example showing the parts of a rule based systems and the interactions between them.

There are two ways for a rule to set new values for the data:

- by assignment, where the value is directly set, and
- by assertion. The assertion does not in itself assign a value to the data, but the condition acts like a constraint upon the data value, saying it must be the value specified by the condition.

## COMPONENTS OF RULE-BASED SYSTEMS

The rule-based expert system architecture is an amalgamation of four (4) important components that are focused on different aspects of the problem in hand. From assessing the information to helping machines reach the goal state, these components are integral for the smooth functioning of rule-based systems. These are:

1. Rule Base: This is a list of rules that is specific to a type of knowledge base, which can be rule-based vs. model-based, etc.
2. Semantic Reasoner: Also known as the inference engine, it infers information or takes necessary actions based on input and the rule base in the knowledge base. Semantic reasoner involves a match-resolve-act cycle.
3. Working Memory: Stores temporary information or data.
4. User Interface: It is the connection to the outside world, input and output signals are sent and received.

## CONSTRUCTION OF RULE-BASED SYSTEMS

Before we move on to discuss the types of rule-based systems, we need to understand its construction, as it plays a crucial role in how the system evaluated the information. The construction of rule-based systems is based on a specific type of logic, such as Boolean logic, fuzzy logic, and probabilistic logic and is categorized into:

1. Knowledge-Based Approach: It is a knowledge-based construction that follows a traditional engineering approach, which is domain-independent. Here, it is important to acquire requirements as well as necessary knowledge before identifying the relationships between attributes.
2. Data-Based Approach: This data-based construction follows a machine learning approach, which, like the earlier approach, is domain-independent. This rule-based approach is subdivided into:

    Supervised Learning.

    Unsupervised Learning.

# DIFFERENCE BETWEEN RULE BASED SYSTEM AND MACHINE LEARNING

Machine learning is among the few techniques of artificial intelligence that is time and again compared with Rule-Based Systems to comprehend their uniqueness. Hence, a discussion on the latter cannot be complete without a comparison between Rule-Based Systems and Machine Learning:

| RULE-BASED SYSTEMS | MACHINE LEARNING |
| --- | --- |
| It is a simplified form of Artificial Intelligence. | Machine Learning is an application of Artificial Intelligence. |
| It is based on Facts and Rules. | It is based on models. |
| Labor intensive and hard to implement. | Easy to implement and use. |
| Deliver excellent performance within a narrow domain. | It helps deliver more accurate results. |
| Limited to human coded information. | It effortlessly deals with complex and excessive data. |

## TYPES OF RULE-BASED SYSTEMS

Like expert systems, rule-based systems can also be categorized into:

1. Forward Chaining:

   Also known as data-driven reasoning, forward chaining is a data- driven technique that follows a deductive approach to reach a conclusion.

2. Backward Chaining:

   Often used in formulating plans, backward chaining is an alternative to forward chaining. It is a goal-driven technique that follows an inductive approach or associative reasoning.

## ADVANTAGES OF RULE-BASED SYSTEMS

- Rule-based programming is easy to understand.
- It can be built to represent expert judgment in simple or complicated subjects.
- The cause-and-effect in Rule-Based Systems is transparent.
- It offers flexibility and an adequate mechanism to model several basic mental processes into machines.
- Mechanizes the reasoning process.

## DISADVANTAGES OF RULE-BASED SYSTEMS

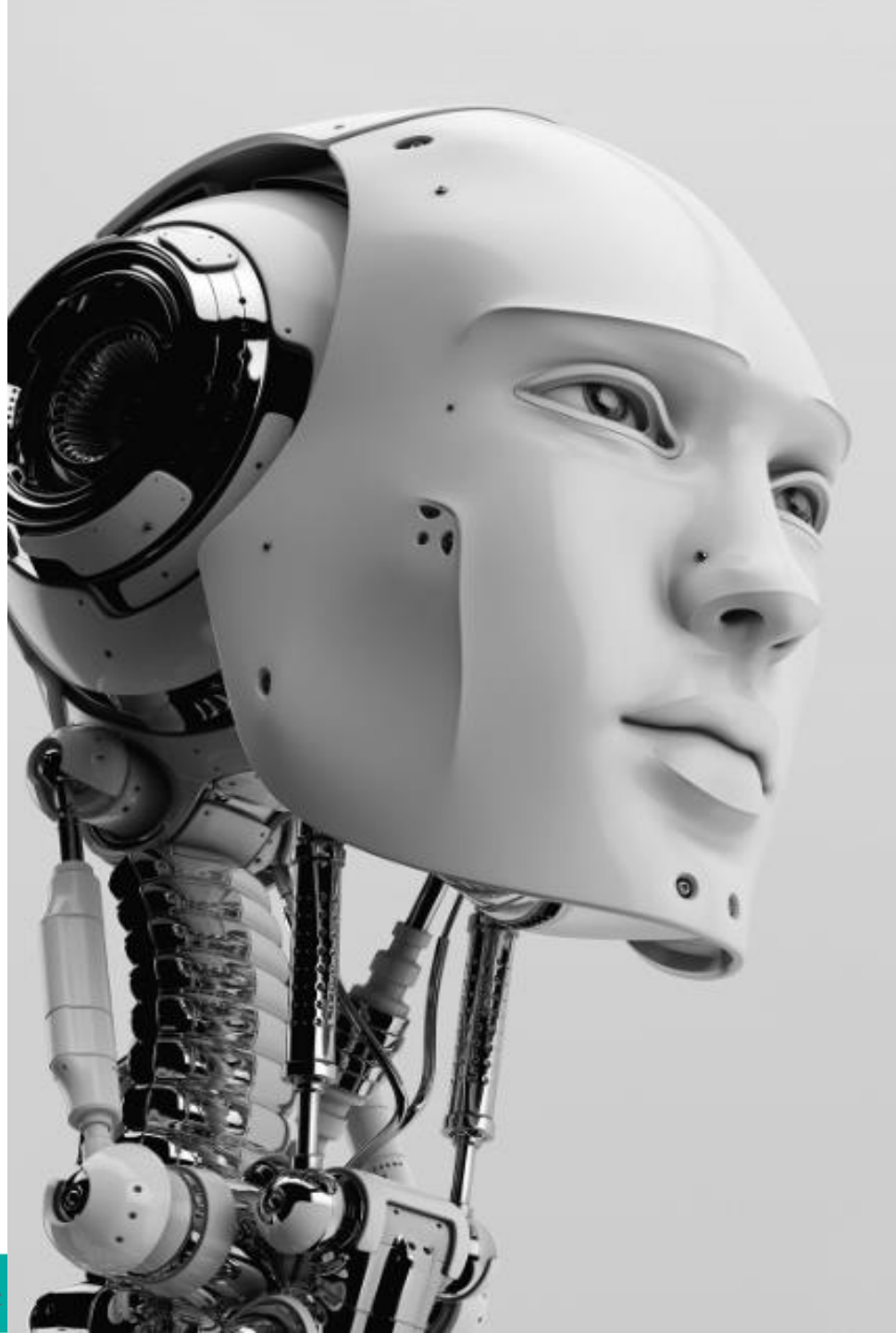
- They require deep domain knowledge and manual work.
- Generating rules for a complex system is quite challenging and time-consuming.
- It has less learning capacity, as it generates results based on the rules.

**One of the first most successful AI approaches, Rule-Based Systems, is paving the way to the development of large and complex applications, with limited efforts. These systems are helping programmers as well as machines to tackle problems with numerous pattern nodes and solve tasks at a higher level of abstraction using human-like thinking and reasoning capabilities. Even though it offers a few limitations, there is no doubt with the ever-evolving technology it too will evolve to be more flexible, effective, and suitable.

# Chapter 3.4: Case-Based Reasoning System

By the end of this topic, you should be able to:

- understand the concepts of case-based reasoning (CBR) system in artificial intelligence.
- understand the step process for CBR .
- know the advantages and disadvantages of CBR.



# Case-Based Reasoning Systems

- Case-based reasoning (CBR) is an experience-based approach to solving new problems by adapting previously successful solutions to similar problems.
- Addressing memory, learning, planning and problem solving, CBR provides a foundation for a new technology of intelligent computer systems that can solve problems and adapt to new situations.
- In CBR, the “intelligent” reuse of knowledge from already-solved problems, or cases, relies on the premise that the more similar two problems are, the more similar their solutions will be.

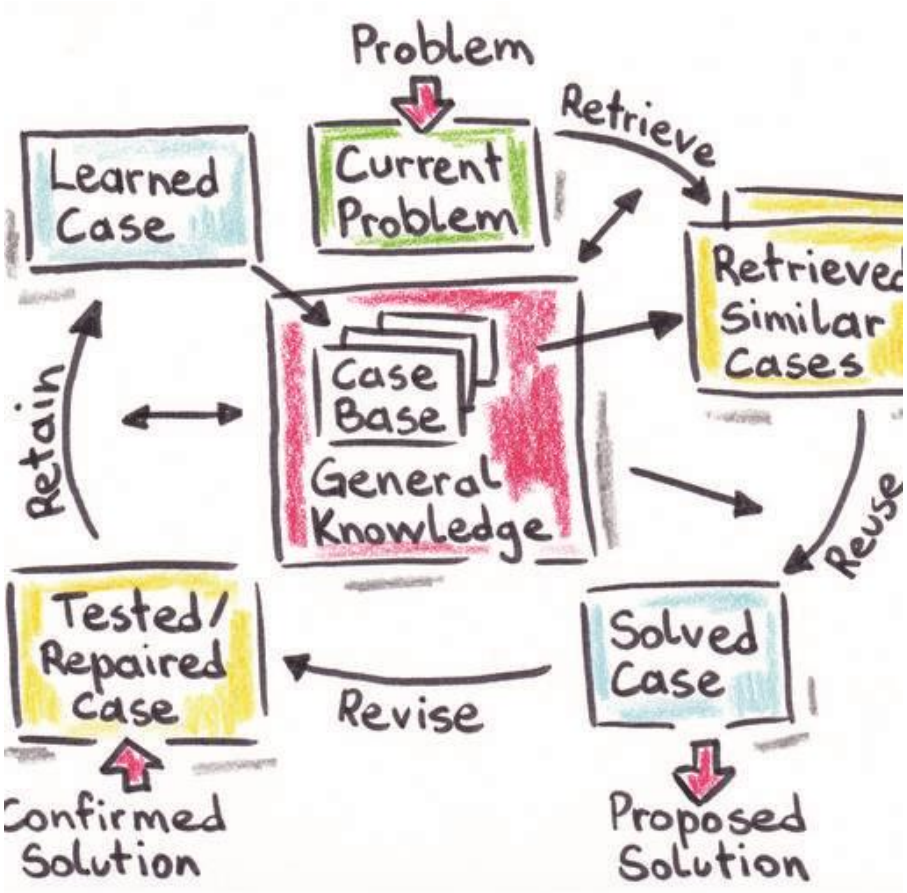
## Comparison to other techniques

- Case-based reasoning has several differences from other AI approaches, such as knowledge-based systems (KBS).
- Rather than relying completely on general knowledge of a problem domain or making associations along generalized relationships between problem descriptors and conclusions, CBR employs the specific knowledge of previously experienced, concrete problem situations.
- CBR also offers incremental, sustained learning in that each time a problem is solved a new experience is retained and can be applied for future problems.

## Four step process for CBR

In general, the case-based reasoning process entails:

- Retrieve- Gathering from memory an experience closest to the current problem.
- Reuse- Suggesting a solution based on the experience and adapting it to meet the demands of the new situation.
- Revise- Evaluating the use of the solution in the new context.
- Retain- Storing this new problem-solving method in the memory system.



## Comparison to other techniques

- Case-based reasoning has several differences from other AI approaches, such as knowledge-based systems (KBS).
- Rather than relying completely on general knowledge of a problem domain or making associations along generalized relationships between problem descriptors and conclusions, CBR employs the specific knowledge of previously experienced, concrete problem situations.
- CBR also offers incremental, sustained learning in that each time a problem is solved a new experience is retained and can be applied for future problems.

## Advantages and disadvantages of CBR

- Remembering past experiences helps learners avoid repeating previous mistakes, and the reasoner can discern what features of a problem are significant and focus on them.
- CBR is intuitive because it reflects how people work. Because no knowledge must be elicited to create rules or methods, development is easier. Another benefit is that systems learn by acquiring new cases through use, and this makes maintenance easier.
- CBR also enables the reasoner to propose solutions to problems quickly. The reasoner can propose solutions in areas that he or she does not fully understand, evaluate solutions when no algorithmic method is available and interpret open-ended and ill-defined concepts.
- On the negative side, critics claim that the main premise of CBR is based on anecdotal/ subjective evidence and that adapting the elements of one case to another may be complex and potentially lead to inaccuracies. However, recent work has enhanced CBR by using a statistical framework. This makes it possible to produce case-based predictions with a higher level of confidence.

# Case-based reasoning Systems

Four step process for CBR
In general, the case-based reasoning process entails:

Retrieve- Gathering from memory an experience closest to the current problem.

Reuse- Suggesting a solution based on the experience and adapting it to meet the demands of the new situation.

Revise- Evaluating the use of the solution in the new context.

Retain- Storing this new problem-solving method in the memory system.

# Links to Read

https://www.javatpoint.com/knowledge-representation-in-ai

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

https://www.mygreatlearning.com/blog/what-is-knowledge-representation/

http://homepage.divms.uiowa.edu/~hzhang/c188/notes/Ch01-10-Hoare.pdf

https://www.professional-ai.com/rule-based-systems.html

https://www.techtarget.com/searchenterpriseai/definition/case-based-reasoning-CBR