**What is knowledge representation?**

Humans are best 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**. Hence we can describe Knowledge representation as following:

* Knowledge representation and reasoning (KR, KRR) is the part of Artificial intelligence which concerned with AI agents thinking and how thinking contributes to intelligent behavior of agents.
* It is responsible for representing information about the real world so that a computer can understand and can utilize this knowledge to solve the complex real world problems such as diagnosis a medical condition or communicating with humans in natural language.
* It is also a way which describes how we can represent knowledge in artificial intelligence. Knowledge representation is not just storing data into some database, but it also enables an intelligent machine to learn from that knowledge and experiences so that it can behave intelligently like a human.

**What to Represent:**

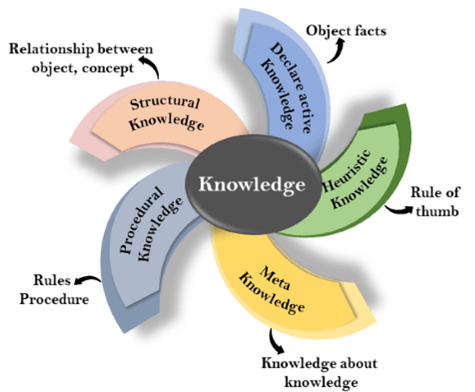
Following are the kind of knowledge which needs to be represented in AI systems:

* **Object:** All the facts about objects in our world domain. E.g., Guitars contains strings, trumpets are brass instruments.
* **Events:** Events are the actions which occur in our world.
* **Performance:** It describe behavior which involves knowledge about how to do things.
* **Meta-knowledge:** It is knowledge about what we know.
* **Facts:** Facts are the truths about the real world and what we represent.
* **Knowledge-Base:** The central component of the knowledge-based agents is the knowledge base. It is represented as KB. The Knowledgebase is a group of the Sentences (Here, sentences are used as a technical term and not identical with the English language).

AD**Knowledge:** Knowledge is awareness or familiarity gained by experiences of facts, data, and situations. Following are the types of knowledge in artificial intelligence:

Types of knowledge

Following are the various types of knowledge:



**1. Declarative Knowledge:**

* Declarative knowledge is to know about something.
* It includes concepts, facts, and objects.
* It is also called descriptive knowledge and expressed in declarativesentences.
* It is simpler than procedural language.

**2. Procedural Knowledge**

* It is also known as imperative knowledge.
* Procedural knowledge is a type of knowledge which is responsible for knowing how to do something.
* It can be directly applied to any task.
* It includes rules, strategies, procedures, agendas, etc.
* Procedural knowledge depends on the task on which it can be applied.

**3. Meta-knowledge:**

* Knowledge about the other types of knowledge is called Meta-knowledge.

**4. Heuristic knowledge:**

* Heuristic knowledge is representing knowledge of some experts in a filed or subject.
* Heuristic knowledge is rules of thumb based on previous experiences, awareness of approaches, and which are good to work but not guaranteed.

**5. Structural knowledge:**

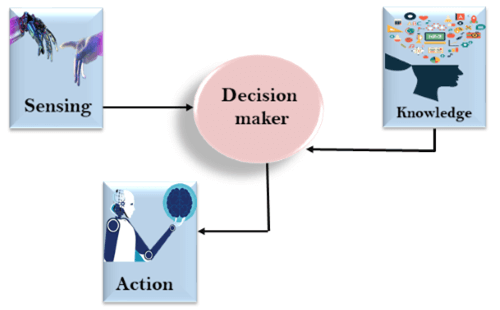
* Structural knowledge is basic knowledge to problem-solving.
* It describes relationships between various concepts such as kind of, part of, and grouping of something.
* It describes the relationship that exists between concepts or objects.

**The relation between knowledge and intelligence:**

Knowledge of real-worlds plays a vital role in intelligence and same for creating artificial intelligence. Knowledge plays an important role in demonstrating intelligent behavior in AI agents. An agent is only able to accurately act on some input when he has some knowledge or experience about that input.

Let's suppose if you met some person who is speaking in a language which you don't know, then how you will able to act on that. The same thing applies to the intelligent behavior of the agents.

As we can see in below diagram, there is one decision maker which act by sensing the environment and using knowledge. But if the knowledge part will not present then, it cannot display intelligent behavior.

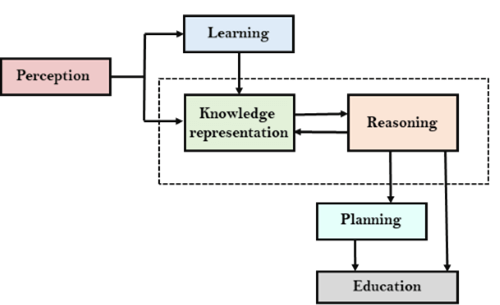


AD

**AI knowledge cycle:**

An Artificial intelligence system has the following components for displaying intelligent behavior:

* Perception
* Learning
* Knowledge Representation and Reasoning
* Planning
* Execution



The above diagram is showing how an AI system can interact with the real world and what components help it to show intelligence. AI system has Perception component by which it retrieves information from its environment. It can be visual, audio or another form of sensory input. The learning component is responsible for learning from data captured by Perception comportment. In the complete cycle, the main components are knowledge representation and Reasoning. These two components are involved in showing the intelligence in machine-like humans. These two components are independent with each other but also coupled together. The planning and execution depend on analysis of Knowledge representation and reasoning.

**Approaches to knowledge representation:**

There are mainly four approaches to knowledge representation, which are givenbelow:

1. Simple relational knowledge:

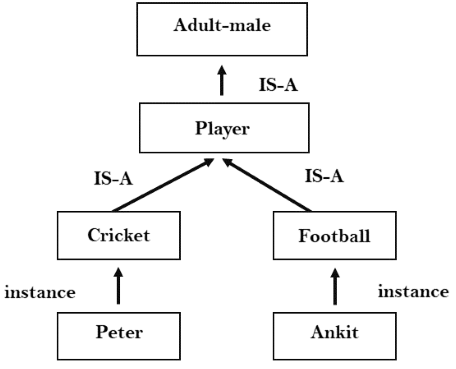
* It is the simplest way of storing facts which uses the relational method, and each fact about a set of the object is set out systematically in columns.
* This approach of knowledge representation is famous in database systems where the relationship between different entities is represented.
* This approach has little opportunity for inference.

**Example: The following is the simple relational knowledge representation.**

|  |  |  |
| --- | --- | --- |
| **Player** | **Weight** | **Age** |
| Player1 | 65 | 23 |
| Player2 | 58 | 18 |
| Player3 | 75 | 24 |

2. Inheritable knowledge:

* In the inheritable knowledge approach, all data must be stored into a hierarchy of classes.
* All classes should be arranged in a generalized form or a hierarchal manner.
* In this approach, we apply inheritance property.
* Elements inherit values from other members of a class.
* This approach contains inheritable knowledge which shows a relation between instance and class, and it is called instance relation.
* Every individual frame can represent the collection of attributes and its value.
* In this approach, objects and values are represented in Boxed nodes.
* We use Arrows which point from objects to their values.
* **Example:**



3. Inferential knowledge:

* Inferential knowledge approach represents knowledge in the form of formal logics.
* This approach can be used to derive more facts.
* It guaranteed correctness.
* **Example:** Let's suppose there are two statements:
  1. Marcus is a man
  2. All men are mortal  
     Then it can represent as;  
       
     **man(Marcus)  
     ∀x = man (x) ----------> mortal (x)s**

4. Procedural knowledge:

* Procedural knowledge approach uses small programs and codes which describes how to do specific things, and how to proceed.
* In this approach, one important rule is used which is **If-Then rule**.
* In this knowledge, we can use various coding languages such as **LISP language** and **Prolog language**.
* We can easily represent heuristic or domain-specific knowledge using this approach.
* But it is not necessary that we can represent all cases in this approach.

Requirements for knowledge Representation system:

A good knowledge representation system must possess the following properties.

1. **1. Representational Accuracy:**  
   KR system should have the ability to represent all kind of required knowledge.
2. **2. Inferential Adequacy:**  
   KR system should have ability to manipulate the representational structures to produce new knowledge corresponding to existing structure.
3. **3. Inferential Efficiency:**  
   The ability to direct the inferential knowledge mechanism into the most productive directions by storing appropriate guides.
4. **4. Acquisitional efficiency-** The ability to acquire the new knowledge easily using automatic methods.

**Representations and Mappings**

In order to solve complex problems encountered in artificial intelligence, one needs both a large amount of knowledge and some mechanism for manipulating that knowledge to create solutions.

Knowledge and Representation are two distinct entities. They play central but distinguishable roles in intelligent system.

Knowledge is a description of the world. It determines a system’s competence by what it knows.

Representation is the way knowledge is encoded. It defines a system’s performance in doing something.

Different types of knowledge require different kinds of representation.

The knowledge Representation models/mechanisms are often based on:

* Logic
* Rules
* Frames
* Semantic Net

[Knowledge is categorized into two major types.](https://medium.com/@dpthegrey/knowledge-is-categorized-into-two-major-types-b3d50f42b34e)

A variety of ways of representing knowledge have been exploited in AI programs.

There are two different kinds of entities, we are dealing with.

1. Facts: Truth in some relevant world. Things we want to represent.
2. Representation of facts in some chosen formalism. Things we will actually be able to manipulate.

These entities are structured at two levels:

1. The knowledge level, at which facts are described.
2. The symbol level, at which representation of objects are defined in terms of symbols that can be manipulated by programs.

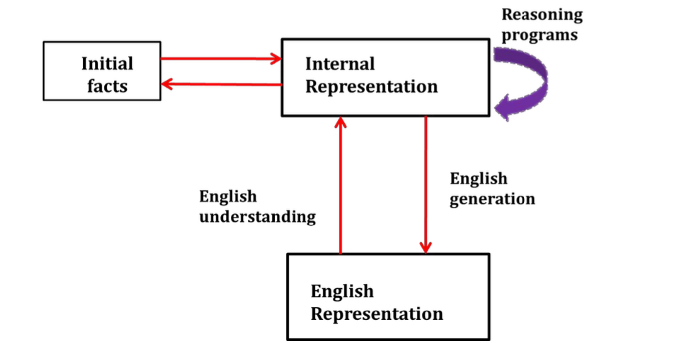


Figure: Mapping between Facts and Representation

**Issues in Knowledge Representation**

The fundamental goal of knowledge Representation is to facilitate inference (conclusions) from knowledge.

The issues that arise while using KR techniques are many. Some of these are explained below.

**Important Attributed:**

Any attribute of objects so basic that they occur in almost every problem domain?

There are two attributed “instance” and “isa”, that are general significance. These attributes are important because they support property inheritance.

**Relationship among attributes:**

Any important relationship that exists among object attributed?

The attributes we use to describe objects are themselves entities that we represent.

The relationship between the attributes of an object, independent of specific knowledge they encode, may hold properties like:

1. Inverse — This is about consistency check, while a value is added to one attribute. The entities are related to each other in many different ways.
2. Existence in an isa hierarchy — This is about generalization-specification, like, classes of objects and specialized subsets of those classes, there are attributes and specialization of attributes. For example, the attribute height is a specialization of general attribute physical-size which is, in turn, a specialization of physical-attribute. These generalization-specialization relationships are important for attributes because they support inheritance.
3. Technique for reasoning about values — This is about reasoning values of attributes not given explicitly. Several kinds of information are used in reasoning, like, height: must be in a unit of length, Age: of a person cannot be greater than the age of person’s parents. The values are often specified when a knowledge base is created.
4. Single valued attributes — This is about a specific attribute that is guaranteed to take a unique value. For example, a baseball player can at time have only a single height and be a member of only one team. KR systems take different approaches to provide support for single valued attributes.

**Choosing Granularity:**

At what level of detail should the knowledge be represented?

Regardless of the KR formalism, it is necessary to know:

* At what level should the knowledge be represented and what are the primitives?
* Should there be a small number or should there be a large number of low-level primitives or High-level facts.
* High-level facts may not be adequate for inference while Low-level primitives may require a lot of storage.

Example of Granularity:

* Suppose we are interested in following facts:

*John spotted Sue.*

*This could be represented as*

**Spotted (agent(John),object (Sue))**

Such a representation would make it easy to answer questions such are:

* Who spotted Sue?

Suppose we want to know:

* Did John see Sue?

Given only one fact, we cannot discover that answer.

We can add other facts, such as

**Spotted(x, y) -> saw(x, y)**

We can now infer the answer to the question.

**Set of objects:**

How should sets of objects be represented?

There are certain properties of objects that are true as member of a set but not as individual;

Example: Consider the assertion made in the sentences:

*“there are more sheep than people in Australia”, and*

*“English speakers can be found all over the world.”*

To describe these facts, the only way is to attach assertion to the sets representing people, sheep, and English.

The reason to represent sets of objects is: if a property is true for all or most elements of a set, then it is more efficient to associate it once with the set rather than to associate it explicitly with every elements of the set.

This is done,

* in logical representation through the use of universal quantifier, and
* in hierarchical structure where node represent sets and inheritance propagate set level assertion down to individual.

**Finding Right structure:**

Given a large amount of knowledge stored in a database, how can relevant parts are accessed when they are needed?

This is about access to right structure for describing a particular situation.

This requires, selecting an initial structure and then revising the choice.

While doing so, it is necessary to solve following problems:

* How to perform an initial selection of the most appropriate structure.
* How to fill in appropriate details from the current situations.
* How to find a better structure if the one chosen initially turns out not to be appropriate.
* What to do if none of the available structures is appropriate.
* When to create and remember a new structure.

There is no good, general purpose method for solving all these problems. Some knowledge representation techniques solve some of these issues.