KNOWLEDGE REPRESENTATION – LECTURE 3

Knowledge is the information about a domain that can be used to solve problems in that domain. To solve many problems requires much knowledge, and this knowledge must be represented in the computer.
A **representation scheme** is the form of the knowledge that is used in an agent. A representation of some piece of knowledge is the internal representation of the knowledge. A representation scheme specifies the form of the knowledge.
A **knowledge base** is the representation of all of the knowledge that is stored by an agent.
Knowledge-based systems are systems for which intentional stance is grounded by design in symbolic representation.

Qualities of a good representation scheme

- Rich enough to express the knowledge needed to solve the problem.
- Able to be acquired from people, data and past experiences.
- As close to the problem as possible; it should be compact, natural, and maintainable.
- It should be easy to see the relationship between the representation and the domain being represented, so that it is easy to determine whether the knowledge represented is correct. A small change in the problem should result in a small change in the representation of the problem.
- Agreeable to efficient computation - it is able to express features of the problem that can be exploited for computational gain and able to trade off accuracy and computation time.

There are two broad categories of knowledge:-

Explicit Knowledge
This type of knowledge is formalized and codified, and is sometimes referred to as know-what. It is therefore fairly easy to identify, store, and retrieve. The greatest challenge with explicit knowledge is similar to information. It involves ensuring that people have access to what they need; that important knowledge is stored; and that the knowledge is reviewed, updated, or discarded.
Explicit knowledge is found in: databases, memos, notes, documents, etc.

Tacit Knowledge
It is sometimes referred to as know-how and refers to intuitive, hard to define knowledge that is largely experience based. Because of this, tacit knowledge is often context dependent and personal in nature. It is hard to communicate and deeply rooted in action, commitment, and involvement.
Tacit knowledge is also regarded as being the most valuable source of knowledge, and the most likely to lead to breakthroughs in the organization
Tacit knowledge is found in: the minds of human stakeholders. It includes cultural beliefs, values, attitudes, mental models, etc. as well as skills, capabilities and expertise.

Representations

Once you have some requirements on the nature of a solution, you must represent the problem so a computer can solve it.
Computers and human minds are examples of **physical symbol systems**.
A **symbol** is a meaningful pattern that can be manipulated. E.g written words, sentences, gestures, marks on paper, or sequences of bits.
A **symbol system** creates, copies, modifies, and destroys symbols. Essentially, a symbol is one of the patterns manipulated as a unit by a symbol system. The term physical is used, because symbols in a physical symbol system are physical objects that are part of the real world, even though they may be internal to computers and brains. They may also need to physically affect action or motor control.
Much of AI rests on the **physical symbol system hypothesis.**
A **physical symbol system** has the necessary and sufficient means for general intelligent action. It means that any intelligent agent is necessarily a physical symbol system.
An **intelligent agent** can be seen as manipulating symbols to produce action. Many of these symbols are used to refer to things in the world. Other symbols may be useful concepts that may or may not have external meaning. Yet other symbols may refer to internal states of the agent.

The following are two levels that seem to be common to both biological and computational entities:

- The **knowledge level** is a level of abstraction that considers what an agent knows and believes and what its goals are. The knowledge level considers what an agent knows, but not how it reasons. For example, the delivery agent's behavior can be described in terms of whether it knows that a parcel has arrived or not and whether it knows where a particular person is or not. Both human and robotic agents can be described at the knowledge level. At this level, you do not specify how the solution will be computed or even which of the many possible strategies available to the agent will be used.
- The **symbol level** is a level of description of an agent in terms of the reasoning it does. To implement the knowledge level, an agent manipulates symbols to produce answers. Many cognitive science experiments are designed to determine what symbol manipulation occurs during reasoning. Note that whereas the knowledge level is about what the agent believes about the external world and what its goals are in terms of the outside world, the symbol level is about what goes on inside an agent to reason about the external world.

Reasoning and Acting

Reasoning is the act of deriving a conclusion from certain premises using a given methodology. Reasoning is a process of thinking; reasoning is logically arguing; reasoning is drawing inference. Many types of Reasoning have long been identified and recognized, but many questions regarding their logical and computational properties still remain controversial.

The popular methods of Reasoning include **abduction**, **induction**, **model- based** and **confirmation**. All of them are intimately related to problems of belief revision and theory development, knowledge assimilation, discovery and learning

- **Deductive reasoning** is a logical process in which a conclusion is based on the concordance of multiple premises that are generally assumed to be true. Deductive reasoning is sometimes referred to as top-down logic.
- **Inductive reasoning** is a logical process in which multiple premises, all believed true or found true most of the time, are combined to obtain a specific conclusion. Inductive reasoning is often used in applications that involve prediction, forecasting, or behavior.
- In **abductive reasoning**, the premises do not guarantee the conclusion. One can understand abductive reasoning as "inference to the best explanation".
- **Model-based reasoning** refers to an inference method used in expert systems based on a model of the physical world. With this approach, the main focus of application development is developing the model. Then at run time, an "engine" combines this model knowledge with observed data to derive conclusions such as a diagnosis or a prediction.

Example

If we know : Robins are birds. All birds have wings.
Then if we ask : Do robins have wings?
Some reasoning (although very simple) has to go on answering the question.

The manipulation of symbols to produce action is called **reasoning**.

One way that AI representations differ from computer programs in traditional languages is that an AI representation typically specifies **what** needs to be computed, not **how** it is to be computed. We might specify that the agent should find the most likely disease a patient has, or specify that a robot should get coffee, but not give detailed instructions on how to do these things. Much AI reasoning involves searching through the space of possibilities to determine how to complete a task.

In deciding what an agent will do, there are three aspects of computation that must be distinguished:

1. The computation that goes into the design of the agent,
2. The computation that the agent can do before it observes the world and needs to act
3. The computation that is done by the agent as it is acting.

- **Design time reasoning** is the reasoning that is carried out to design the agent. It is carried out by the designer of the agent, not the agent itself.
- **Offline computation** is the computation done by the agent before it has to act. It can include compilation and learning. Offline, the agent takes background knowledge and data and compiles them into a usable form called a **knowledge base**. **Background knowledge** can be given either at design time or offline.

- **Online computation** is the computation done by the agent between observing the environment and acting in the environment. A piece of information obtained online is called an observation. An agent typically must use both its knowledge base and its observations to determine what to do.

Two broad strategies have been pursued in building agents:

- The first is to simplify environments and build complex reasoning systems for these simple environments. For example, factory robots can do sophisticated tasks in the engineered environment of a factory, but they may be hopeless in a natural environment. Much of the complexity of the problem can be reduced by simplifying the environment. This is also important for building practical systems because many environments can be engineered to make them simpler for agents.
- The second strategy is to build simple agents in natural environments. This is inspired by seeing how **insects** can survive in complex environments even though they have very limited reasoning abilities. Researchers then make the agents have more reasoning abilities as their tasks become more complicated.

Any knowledge system must reason, if it is required to do something which has not been told explicitly. For reasoning, the system must find out what it needs to know from what it already knows.

Logic is a language for reasoning. It is a collection of rules called Logic arguments, we use when doing logical reasoning.
Logic reasoning is the process of drawing conclusions from premises using rules of inference.
The study of logic is divided into formal and informal logic.
The **formal logic** is sometimes called **symbolic logic**.
Symbolic logic is the study of symbolic abstractions (construct) that capture the formal features of logical inference by a formal system
Formal system consists of two components, a formal language plus a set of inference rules. The formal system has axioms
Axiom is a sentence that is always true within the system.