Knowledge representation is "a science of translating actual knowledge into a format that can be used by the computer".

Outline of this lecture

- Definitions of Knowledge
- Knowledge Pyramid
- Knowledge Representation based on Logic

Definitions of Knowledge

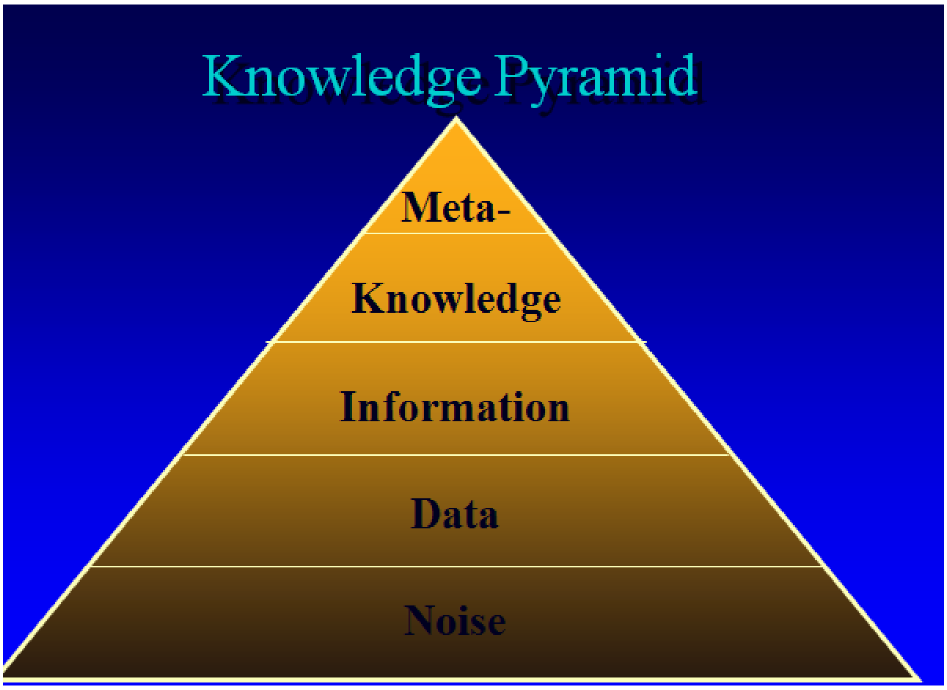
Knowledge is an awareness or understanding of someone or something, such as facts, data, information, descriptions, or skills, which is acquired through experience or education by perceiving, discovering, or learning.

Now, what are differences between Data, Information, and Knowledge?

- Data:
    - Unorganized and unprocessed facts;
    - Accurate and timely,
    - Static;
    - A set of discrete facts about events.
    - Can lead to an increase in understanding and decrease in uncertainty

- Information:
  - Aggregation of data that makes decision making easier
  - Valuable because it can affect behavior, a decision, or an outcome. For example, if a manager is told his/her company's net profit decreased in the past month, he/she may use this information as a reason to cut financial spending for the next month.
- Knowledge
- Derived from information in the same way information is derived from data.
- Knowledge is not something just to be transmitted and stored, it has to be constructed.

Knowledge Pyramid



Knowledge Representation based on Logic

Oldest form of KR in computer, concerned with the truthfulness of a chain of statements. Two kinds of logic:

- Propositional Logic
- Predicate Calculus

Implemented in PROLOG (Programming in Logic) language.

Do not forget: Logical operators

Formal Name	Symbols
Negation (Not)	$\neg$
Conjunction(And)	$\wedge$
Disjunction (Or)	$\vee$
Conditional(Implies)	$\rightarrow$
Biconditional(Equivalent)	$\leftrightarrow$

Propositional Logic

Is based on statements, which have truth values (True or False). The statements in Propositional Logic are formed from "Atomic" symbols.

- **Propositional Logic Symbols**

P,R,Q,T,... English alphabets

True, False Truth alphabets

$\vee$, $\wedge$, ~, ... Connective

- **Propositional Logic Sentences**

1. Every Propositional symbols and Truth symbols are Sentences,

   Ex: Q, P, True.
2. The Negation of a sentence is a sentence. Ex: $\sim$ P, $\sim$False.
3. The Conjunction(AND) of two sentences is a sentence,

   Ex: $P \wedge Q$, $P \wedge \sim R$
4. The Disjunction(OR) of two sentences is a sentence,

   Ex: $P \vee Q$, $P \vee \sim R$
5. The Implication of one sentence from another is a sentence,

   Ex: $P \rightarrow Q$, $R \rightarrow \sim T$
6. The Equivalence of two sentences is a sentence, Ex: $P \vee Q \leftrightarrow R$

Definition: Let R be Propositional Logic constructed by connecting atomic propositions P, Q, ... by operators, The truth value assignment to proposition sentence that keep R true is called" Interpretation".

Ex: $P \wedge Q$, the possible interpretation is (P true, Q true).

$P \vee Q$, they may be more than one interpretation.

Exercise: 1: Try $(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$

2: Write Propositional Logic for the English sentence:

- Freed is a rich.
- Ali is a teacher.
- The sky is cloudy, it will rain.

• Rules in Propositional Logic

1. $\sim(\sim P) \equiv P$
2. Commutative Rules:

   $P \wedge Q \equiv Q \wedge P$

   $P \vee Q \equiv Q \vee P$
3. Associative Rules:

   $(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$

   $(P \vee Q) \vee R \equiv P \vee (Q \vee R)$
4. Distributive Rules:

   $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$

   $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$
5. Morgan's Rules:

   $\sim(P \wedge Q) \equiv \sim P \vee \sim Q$

   $\sim(P \vee Q) \equiv \sim P \wedge \sim Q$
6. $P \to Q \equiv \sim P \vee Q$
7. $\sim P \to Q \equiv P \vee Q$
8. Contra Positive Rule:

   $P \to Q \equiv \sim Q \to \sim P$

Predicate Logic (Predicate Calculus) called First Order Logic FOL

It is similar formalism like Propositional logic, but the capability of reasoning and knowledge representation using Predicate Logic is higher than Propositional Logic. It include two more quantifiers, namely, the Universal $\forall$ (that a sentence is true for all values of variable X) , and the Existential $\exists$ (that a sentence is true for at least one value of a variable X).

Ex: Ali is a man. man(ali). or is_a(ali , man). or ali_is_man.

Ex: Boys like football. like(Boys , football).

Ex: All boys like football. $\forall_X$ (boy(X) → like(X, football)).

Ex: Some boys like basketball. $\exists_X$ (boy(X) → like(X, basketball)).

Ex: All basketball players are tall.

$\forall_X$ (basketball_player(X) → tall(X)).

Ex: Nobody like taxes. ~ $\exists_X$ (like(X, taxes)).

- ## Alphabets of FOL

1. Constants a,b,c,...
2. Variables A,B,C,...
3. Functions f,g,h,...
4. Operators ∨ , ∧ , ~ , ...
5. Quantifiers ∀ , ∃
6. Predicates P,Q,R,...

Definition: These sentences of FOL are Well Form Formulas WFF defined as follow:

1. If $P(t_1,t_2,\ldots t_n)$ is any predicate, then P is a WFF.
2. If P and Q are WFF, then P ∨ Q , (P ∧ Q), ~ P , P → Q ,... are WFF.

Exercise: Rewrite the following English sentences in FOL:

1. John loves children who take biscuits.
2. John and Bin are friends.

Note: The previous rules in PL can be also applied in Predicate Calculus.

- ## Rules of Quantifiers

$\neg\exists x\, P(X) \equiv \forall x \neg P(X)$

$\neg\forall x\, P(X) \equiv \exists x \neg P(X)$

$\exists x\, P(X) \equiv \neg\forall x \neg P(X)$

$\forall x\, P(X) \equiv \neg\exists x \neg P(X)$

$\forall x\, (P(X) \wedge Q(x)) \equiv \forall x\, P(X) \wedge \forall x\, Q(x)$

$\exists x\, (P(X) \vee Q(x)) \equiv \exists x\, P(X) \vee \exists x\, Q(x)$

Then, the **Representing knowledge in FOL:**

- **Objects:** people

  $John\ , Mary\ , Jane\ , \ldots$

- **Properties:** gender

  $Male\ (x), Female\ (x)$

- **Relations:** parenthood, brotherhood, marriage

  $Parent\ (x, y), Brother\ (x, y), Spouse\ (x, y)$

- **Functions:** mother-of (one for each person x)

  $MotherOf\ (x)$

Relations between predicates and functions

How relate to each other:

- Male and female are disjoint categories

$$\forall x \; Male(x) \Leftrightarrow \neg Female(x)$$

- Parent and child relations are inverse

$$\forall x, y \; Parent(x, y) \Leftrightarrow Child(y, x)$$

- A grandparent is a parent of parent

$$\forall g, c \; Grandparent(g, c) \Leftrightarrow \exists p \; Parent(g, p) \wedge Parent(p, c)$$

- A sibling is another child of one's parents

$$\forall x, y \; Sibling(x, y) \Leftrightarrow (x \neq y) \wedge \exists p \; Parent(p, x) \wedge Parent(p, y)$$

- And so on ….

Next Lecture

Continue Predicate logic