CS-208:Artificial Intelligence
Lectures-12
Knowledge Representation
Using
Propositional Logic & Predicate Logic

Knowledge Representation

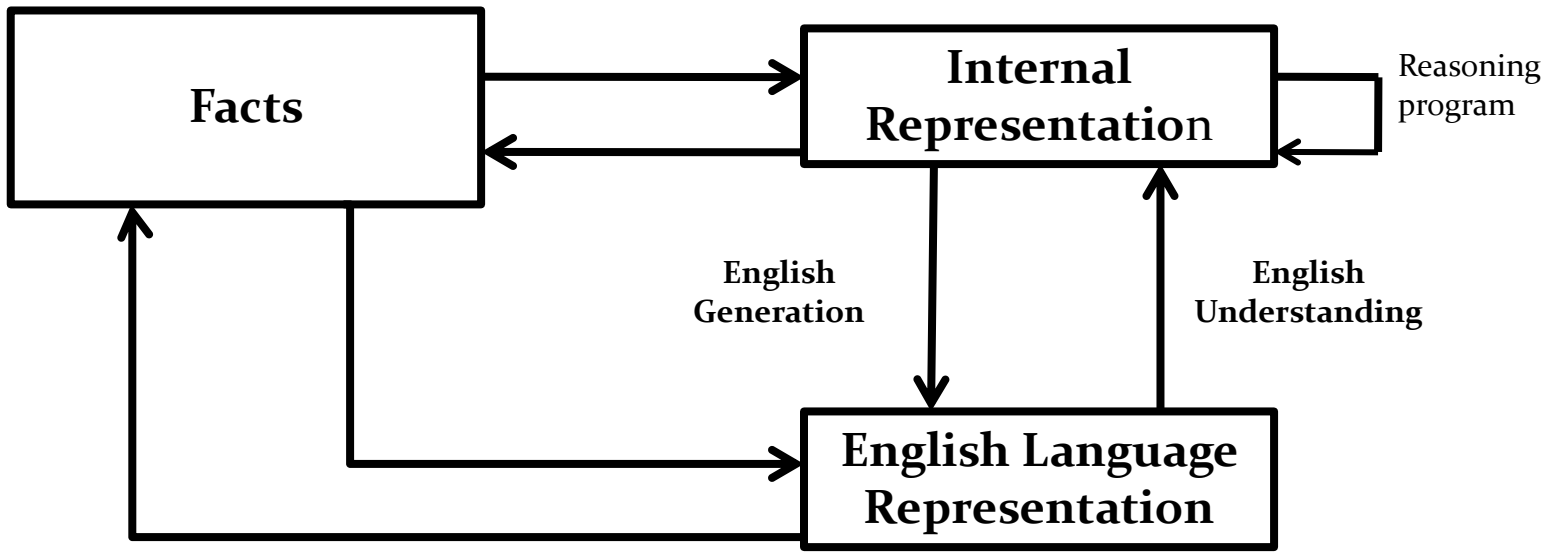
In the previous lectures discussed about the general methods of manipulating the knowledge using search. Now we will look the various ways of representing the knowledge.

There are two entities that are pertained to all kinds of knowledge representations.

- **Facts**: Truth in some relevant world (these are the things we want to represent)
- **Representation of Facts** in some chosen formalism(these are things we actually able to manipulate)

Each representation must associates with some Mapping functions. These functions map from facts to representation and from representation to facts.

An illustrative for Mapping Funtions:



An Illustrative Example

Knowledge Representation Using Mathematical Logic

1. Spot is a dog — *dog(Spot)*
2. All dogs have tails — $\forall x: dog(x) \rightarrow hastail(x)$

A new representation can be generated using the deduction mechanism of logic

hastail(Spot)

An English sentence can be generated from hastail (Spot) using an appropriate mapping function

Spot has a tail

It is important to note that available mapping functions are not always one to one.

- All dog have tails: implies *Every dog has at least one tail* or *Each dog has several tails*
- Every dog has a tail: implies *Every dog has at least one tail* or *There is a tail that every dog has*
  - Here both represent same facts *Every dog has at least one tail*

Knowledge representation using Propositional logic

Knowledge representation using propositional logic is appealing for two reasons.

- It is simple to deal with
- There exist a deduction procedure for it

	English Sentence	**Propositions**
1	It is raining	**Raining**
2	It is sunny	**Sunny**
3	It is windy	**Windy**
4	If it is raining then it is not sunny	**Raining** $\rightarrow \neg$**Sunny**

Limitations-1

Amit is a student **AmitStudent**

Anand is student **AnandStudent**

These two are completely separate assertions and it is not possible to draw any conclusion about similarities between Amit and Anand

Limitations-2

It is also very difficult to represent a simple fact using propositional logic like

"***All men are mortal***"

Knowledge Representation using Predicate Logic

The prime limitations of representing knowledge by using propositional logic can be overcome by using predicate logic. For examples

English Sentence	**Predicate Statement**
Amit is a student	**Student(Amit)**
Anand is student	**Student(Anand)**
All men are mortal	$\forall x$: **Man**$(x) \rightarrow$ **mortal**(x)

- Here the structure of representation reflects the structure the knowledge itself

Convert the following English sentences to Predicate logic statements

1. Marcus was a man
2. Marcus was a Pompeian
3. All Pompeian were Romans
4. Caesar was a ruler
5. All Romans were either loyal to Caesar or hate him
6. Everyone is loyal to someone
7. People only try to assassinate ruler they are not loyal to
8. Marcus tried to assassinate Caesar

Consider the statement:

All Romans were either loyal to Caesar or hate him

$\forall x$: Roman(x)$\rightarrow$ loyalto(x, Caesar) $\vee$ hate(x, Caesar)

Here the use of inclusive-or is not a correct mapping (or interpretation) because both loyalto(x, Caesar) and hate(x, Caesar) can not be true simultaneously. So we need to use Exclusive-OR instead of Inclusive-OR.

Exclusive-OR Operations on A and B:

$A \oplus B = (A \vee B) \wedge \neg(A \wedge B)$

$A \oplus B = (A \wedge \neg B) \vee (\neg A \wedge B)$

Predicate logic statements

1. Man(Marcus)
2. Pompeian(Marcus)
3. $\forall x$: Pompeian(x) $\rightarrow$ Roman(x)
4. Ruler(Caesar)
5. $\forall$x: Roman(x) $\rightarrow$[(loyalto(x, Caesar) $\vee$hate(x,Caesar))$\wedge$ $\neg$(loyalto(x, Caesar) $\wedge$hate(x,Caesar))]

   or

   $\forall$x: Roman(x) $\rightarrow$[(loyalto(x, Caesar) $\wedge$ $\neg$hate(x,Caesar))$\vee$ ($\neg$loyalto(x, Caesar) $\wedge$hate(x,Caesar))]
6. $\forall$x: $\exists$y: loyalto(x, y)
7. $\forall$x: $\forall$y: Person(x) $\wedge$Ruler(y) $\wedge$trytoassassinate(x, y) $\rightarrow$ $\neg$loyalto(x, y)
8. trytoassassinate(Marcus, Caesar)
9. $\forall$x: Man(x) $\rightarrow$ Person(x)

Suppose, we want to prove by using these predicate statements to answer the question ***Was Marcus loyal to Caesar?*** *Here we will be able to prove Marcus was not loyal to Caesar by using the statements 7 & 8*

Formal Proof

(by reasoning backward from the desired goal)

¬loyal(Marcus, Caesar)

↓ *(substitution-7)*

Person(Marcus)∧ Ruler(Caesar)∧ trytoassassinate(Marcus, Caesar)

↓ *(substitution-4)*

Person(Marcus)∧ trytoassassinate(Marcus, Caesar)

↓ *(substitution-8)*

Person(Marcus)

↓ *(substitution-9)*

Man(Marcus)

↓ **(***substitution-1)*

Nil

In order to prove the goal we need to use **Rule of Inference (***A given goal is transformed in to another goal that in turn to be transformed until no unsatisfied goals remains*).

In order to complete the proof we need to add another fact

9. All men are people

After adding this fact, the last goal has been satisfied and able to produce the proof that Marcus was not loyal to Caesar

Three important issues related to the process of converting the English sentences to logical statements and then using those statements to deduce new ones

1. Many English sentences are ambiguous and choosing the correct interpretation may be difficult.

2. There are often choice of ways of representing the knowledge. Simple representations are desirable but they may preclude certain kinds of reasoning.

3. Even in very simple situations, a set of sentences is unlikely to contain all information necessary to reason about the topic at hand. In order to be able to use set statements effectively, it is usually necessary to have access to another set of statements that represents the fact that people consider too obvious to mention.

Adoption of the Strategies used in Producing the Formal Proof

It is very difficult to know in advance which sentence is to be deduced for the question **Was Marcus loyal to Caesar?** Either ¬**loyalto**(*Marcus, Caesar*) or **loyalto**(*Marcus, Caesar*).

- Forward reasoning (start reasoning from given goal) or Backward reasoning (staring reasoning from the desired goal)
- Using some sort of heuristic rules can be used to decide which answer is more likely.
- Trying to prove simultaneously both the answers and stop when one effort becomes successful.
- Trying both to prove and disprove one answer by using the information gained one of the process to guide the other.