Lecture Four

1. Representing Meaning with Predicate Logic

Predicate Logic retains almost all the rules of propositional logic with the exception that it represents propositions differently by introducing internal structures. That is, it can get inside a proposition. To start with, a formula in predicate logic contains predicates and arguments. While an atomic proposition is represented simply as P in propositional logic, in predicate logic, it is represented in the form of (1):

\[ \mathcal{P}(a_0 \ldots a_n) \], where \( \mathcal{P} \) stands for a predicate, and \( a_i \) stands for an argument in i position.

When translating a natural language sentence into a predicate logical formula, use the main predicate to replace the \( \mathcal{P} \) in (1) and use the subject and the object to replace the “\( a_0 \ldots a_n \)” in (1). This is not the whole picture yet, but it serves as a convenient starting point.

Take a look at the following sentences, which should be represented as (4) and (5) respectively. Here, no arguments are involved. So we only use the predicates.

(2) It rains.
(3) 起 (qi3, start)風 (feng1, wind)了 (le, sentence-final particle introducing an event) "The wind blows".
(4) Rain.
(5) BLOW-WIND. [Note that (3) is conventionally treated as a subjectless sentence. Just like (6)]
(6) 下 (xia4, fall)雨 (yu3, rain)了 "It rains".

The following involves natural language sentences with only one argument, i.e. the subject. Their logical formulae in predicate logic are given by the side:

(7) John smiles. SMILE (john)
(8) Peter sleeps. SLEEP (peter)
(9) Mary is happy. HAPPY (mary)

Some conventions:

a. Write the predicate before the argument(s).
b. Write the predicate in small capitals.
c. Write arguments in small letters, even if names are involved.
d. Put arguments in a pair of parenthesis so as to draw the boundary between the predicate and the arguments. (Other logic textbooks may differ.)
e. Treat "be + predicative adjective" as a single predicate.
f. Put a comma between arguments. (We come to this point presently.)
g. When doing translation, leave out the grammatical suffixes, because what gets translated is meaning, not grammatical specifications. Likewise, at this initial level, an active sentence and its transformed passive construction get translated into exactly the same logical formula.

Exercise 1:

(10) Sam cries.
(11) Joan blushes.
(12) Mt. Everest is white.
(13) London is huge.
(14) Sam is British.
Turning to natural language sentences containing predicates with two arguments. Such a predicate should have two slots reserved. That is, it takes the form of \( P(a_1, a_2) \). A predicate having two places for arguments is also called a two-place predicate. It is also called a two-arity predicate. Now we can attempt to translate the following sentences into predicate logic:

\[
\begin{align*}
(15) & \quad \text{John loves Mary.} & \quad \text{LOVE}(\text{john, mary}). \\
(16) & \quad \text{Peter visited Shanghai.} & \quad \text{VISIT}(\text{peter, shanghai}).
\end{align*}
\]

It is in the lexicon that the arity of a predicate is specified. If a predicate can either take one argument or two, it is two predicates in disguise. That is, it means two predicates happen to share the same form. Sometimes, a two-place predicate may take only one argument in the sentence because the other argument is omitted. That is a case of ellipsis.

Sentences containing predicates with three arguments can also be found. Such a predicate is called a three-place predicate, taking the form of \( P(a_1, a_2, a_3) \). Some examples are: introduce, give, place, tell, teach (note that some of the three-place verbs have other uses, i.e. two-place or intransitive uses).

\[
\begin{align*}
(17) & \quad \text{Bill introduced Peter to Mary} & \quad \text{INTRODUCE}(\text{bill, peter, mary}) \\
(18) & \quad \text{Sam give the book to Jill} & \quad \text{GIVE}(\text{sam, book, jill}) \\
(19) & \quad \text{John told Dan the story.} & \quad \text{TELL}(\text{john, story, dan}) \\
(20) & \quad \text{Susan taught him “I Ching”.} & \quad \text{TEACH}(\text{susan, i-ching, a})
\end{align*}
\]

Questions for discussion:

1. When translating a di-transitive sentence, which argument order should be followed? [1]
2. How to translate definite noun phrases? [2]
3. How to translate pronouns? [3]

When it comes to translating compound formulae, we still follow the conventions of propositional calculus. Conjoining (15) and (17) will give us (21), and (8) and (20) can also be joined together by a biconditional, yielding (22):

\[
\begin{align*}
(21) & \quad \text{LOVE}(\text{john, mary}) \land \text{INTRODUCE}(\text{bill, peter, mary}) \\
(22) & \quad \text{SLEEP}(\text{peter}) \leftrightarrow \text{TEACH}(\text{susan, i-ching, peter}) \quad \text{[What have I done to the pronoun?]}
\end{align*}
\]

Up to now, the language of predicate logic seems very limited in its expressive power, because only names or equivalents can serve as arguments. Think about propositional logic. There, any statement can be represented. But predicate logic is certainly more expressive. The arguments can be taken by both constants and variables. Logical constants have fixed denotata (singular form: denotatum =df. the actual object referred to by a linguistic expression.). But they can only be used to refer to named or definite individuals. In order to talk about unnamed or indefinite individuals, we need to use variables as arguments. Let us take a look at the following

\[
\begin{align*}
(23) & \quad \text{SLEEP}(x) \land \text{DRINK}(x) \\
(24) & \quad \text{SLEEP}(x) \land \text{DRINK}(y)
\end{align*}
\]

(23) says some individual sleeps and drinks. But it is not specified how many such individual is being considered. (24) says some individual sleeps, and someone else drinks. Again, it is not stated how many such individuals are involved. In both these cases, we have variables as arguments. What is more, each variable is free, meaning that we do not know how many individuals each variable stands for. But once the quantifiers are introduced, things become clearer. In predicate logic, one can

\[\text{df.}\]
logic, there are two standard logical quantifiers, \( \forall \) and \( \exists \). \( \forall \) is the universal quantifier. It can be pronounced as “(for) all” and taken as a capital “a” letter turned upside-down. \( \exists \) is the existential quantifier often pronounced as “(there) exists” and can be taken as an inversed capital “e” letter.

\[
(25) \quad \forall x \, \text{SLEEP}(x). \quad \text{[For all } x, x \text{ sleeps (including human individuals).]}
\]

\[
(26) \quad \exists x \, \text{SLEEP}(x). \quad \text{[There exists at least one } x \text{ such that } x \text{ sleeps.]}\]

Obvious, more qualifications are in order, so we need to give more complex formulae to express more precise thoughts:

\[
(27) \quad \forall x \, (\text{BOY}(x) \rightarrow \text{SLEEP}(x)). \quad \text{[For all } x, \text{ if } x \text{ is a boy, then } x \text{ sleeps.]}\]

\[
(28) \quad \exists x \, (\text{BOY}(x) \land \text{SLEEP}(x)). \quad \text{[There exists at least one boy such that he sleeps.]}\]

An existentially quantified formula comes with an existential presupposition. That is, the mentioned individual is assumed to be in existence. Hence, the main connective associated with existential quantification is conjunction. 2 Think of the truth-table of the conjunction, if one conjunct is false, the whole formula is false. If the existence of the individual turns out to be false, the whole formula is false.

But a universally quantified formula does not have existential presupposition. In (27), if \( x \) is a boy, then he sleeps. The whole formula is true. But if \( x \) is not a boy, the whole formula is still true, according to the truth table of the conditional. Here, the main connective is the conditional sign. If we replaced it with a conjunctive operator, it will give us the wrong meaning, as shown by (29):

\[
(29) \quad \forall x \, (\text{BOY}(x) \land \text{SLEEP}(x)). \quad \text{[For all } x, x \text{ is a boy and } x \text{ sleeps.]}\]

(29) gives the wrong representation because it says everything (in the world) is both a boy and sleeps. When an individual is not a boy, the formula collapses. On the other hand, (30) is also wrong because it lacks the existential presupposition: if no boy exists, the formula is still true. So (30) gives us the wrong prediction on existential quantification.

\[
(30) \quad \exists x \, (\text{BOY}(x) \rightarrow \text{SLEEP}(x))
\]

2. Model-theoretic Interpretation in Predicate Logic

The semantics of predicate logic differs from that of propositional logic. In the latter, the atomic propositions need to have their truth values verified in the real world first. Then the truth values of the compound propositional formulae can be mechanically determined using truth-tables of the logical connectives that are involved in the formulae. If the truth values of the atomic propositions are not known, we can work out all the combinatorial possibilities of the compound formulae by giving each atomic proposition two values, 0, 1, one at a time. The result is a list of exhaustive assignments of truth values and comprehensive computations of their combinatorial results.

In predicate logic, since we can get inside a proposition, we can take a step further by formulating the truth-conditions under which a proposition with given structural properties can be true. And we can test if a given proposition is true or false against the external world in finer ways. To attain this goal, we need to construct a model: a mini-world that contains fewer things than the actual external world so that we can find it more manageable. We could have used a real-world scene, but that would still be unnecessarily complicated, like the following pictures:

2 “main connective” is specified as the connective that must be used together with a quantifier. If the formula is complex, there may be other connectives.
For the purpose of doing semantic interpretation, a much simpler sketch of a model will suffice, like Model 3 below:

\[ M^3 \]

\[
\begin{align*}
& a, b, c, d, e, f, g, h, i, j, k, l \\
& \text{BOY} = \{a, b, c\}, \text{PEOPLE} = \{a, b, c, d, e, f\} \\
& \text{GIRL} = \{d, e\} \\
& \text{HAPPY} = \{a, b, c, d\} \\
& \text{CRY} = \{e\} \\
& \text{LOVE} = \{<a, d>, <b, e>, <e, b>\}
\end{align*}
\]

Given a model \( M^3 \), its domain should contain some individual entities, which are associated with some expressions in a language. It should also contain some sets, which are related to some predicates in a language.

To interpret a formula in predicate logic is to give it a denotational interpretation, which involves computing its truth value. From a denotation-theoretic point of view, a formula in predicate logic can be interpreted according to the following truth-conditions:

(31) \( \text{PRED}_0 = 1 \) or \( \text{PRED}_0 = 0 \) [Since there is no internal structure, such a formula is just like an atomic proposition and is so evaluated.]

(32) \( \text{PRED}_1(\text{arg}_1) = 1 \iff [\text{arg}_1]^{M_i} \in [\text{PRED}_1]^{M_i} \) [The argument here should be a constant. Since a one-place predicate denotes a set of individuals, the argument should be a member of the set to make the formula true.]

(33) \( \text{PRED}_2(\text{arg}_1, \text{arg}_2) = 1 \iff [<\text{arg}_1, \text{arg}_2>]^{M_i} \in [\text{PRED}_2]^{M_i} \) [Again, only constants are being considered. What should be reminded is that a two-place predicate denotes a set whose members are ordered pairs of individuals.]

(34) Now you can work out the truth-condition for a structure containing a three-place predicate.
If arguments are assumed by one or more variables, an additional value assignment function for variables is added to the interpretation process, with the result that now we have $[\alpha]^{M_3g_3}$, where $g$ is the additional variable valuation function. The function $g$ works in the manner illustrated as (36):

$$\begin{align*}
\text{(35)} & \quad \text{Value assignment for variables [valuation function]} \\
\begin{array}{c|c|c|c|c|c|c}
\text{x} & M_3g_3_1 & M_3g_3_2 & M_3g_3_3 & M_3g_3_4 & M_3g_3_5 & \ldots \\
\hline
\text{a} & \text{b} & \text{c} & \text{d} & \text{e} & \ldots & \text{n}
\end{array}
\end{align*}$$

Take the above graph as a process showing how an argument variable in a predicate logical form is given a tentative value. When a value is assigned, a given constant replaces the variable in the logical form as shown by (36) – (38). Then the logical formula can go on with its truth-evaluation process until a truth value is computed. Note that the following interpretations are conducted with reference to the Model 3 as given above.

$$\begin{align*}
\text{(36)} & \quad \text{John loves everyone. } \forall x (\text{PEOPLE} (x) \rightarrow \text{LOVE} (j, x)) \\
\text{(37)} & \quad \begin{align*}
[\text{john}]^{M_3} & = f, \\
[\text{x}]^{M_3g_3_1} & = a, \\
\text{PEOPLE} (x/a) & \rightarrow \text{LOVE} (j, x/a) \quad [x/a \text{ means “using a to replace x”}] \\
a & \in \text{[PEOPLE]}^{M_3} \\
<f, a> & \notin \text{[LOVE]}^{M_3} \\
<f, a> & \notin \{<a, d>, <b, e>, <e, b>\} \\
\end{align*}
\end{align*}$$

The question is: how many assignments need to be performed? That is, when does the valuation machine grind to a halt? Answer: it depends on the nature of the quantifier that binds the variable. A universal quantifier requires its valuation to be exhaustive, unless a 0 is reached before the domain is exhausted. An existential quantifier requires its valuation to identify just one true instance and will not stop its valuation until it finds one. Suppose the domain contains infinite members. Then, in the case of universal quantification, if no false case is found, the valuation machinery will not stop, and in the worst case, the truth of the whole formula can never be established. In the case of existential quantification, if no true case is found, the valuation machinery will also keep on grinding, and in the worst case, since a true instance cannot be found and the domain is infinite, the truth of the whole formula can never be established either.

3. More definitions

Summing up, predicate calculus enables us get into the sentence and study its argument structures. In predicate calculus, we need to distinguish between names and variables. Variables can be bound by quantifiers (or other types of operators such as the $\lambda$-operator), or they can be free. There are only two quantifiers in first-order predicate logic, $\forall$ and $\exists$. $\forall$ is called the universal quantifier, and $\exists$ the existential quantifier. $\forall$ means each and every instance ranging over the variable it binds should satisfy a property. $\exists$ means at least one instantiation of the variable it binds should satisfy a property. For complex logical forms, universal quantification and existential quantification take on different canonical patterns:

$$\begin{align*}
(1) & \quad \forall x (Fx \rightarrow Gx) \\
(2) & \quad \exists x (Fx \land Gx)
\end{align*}$$
A formula containing variables but no binding quantifiers is called a propositional function, or an open proposition, in which the unbound variables are called free variables.
6. Exclusive Disjunction

Exclusive disjunction is different from inclusive disjunction in the sense that it does not allow both the disjuncts to be true. Hence it has the following truth-table:

<table>
<thead>
<tr>
<th>P</th>
<th>P \lor Q</th>
<th>Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Examples in natural language are the following:

1. Either you are mad or I am.
2. Either you go or I go.
3. I will have either tea or coffee.
4. 不bu2 是shi4 你ni2 死si3，就jiu4 是shi4 我wo3 亡wang2。 “Either you die or I die.”
   [Contrast it with the superficially illogical 不是你死，就是我活huo2 ] “Either you die or I live(die)”

Is it absolutely necessary for us to define a new exclusive disjunctor？ Not really, because we can represent exclusive disjunction in terms of the following formula:

5. \( (P \lor Q) \land \neg(P \land Q) \)

Since we do not touch on logical proof in this subject, I won’t show you how to prove that (5) and \( P \lor Q \) are equivalent, i.e. from one, we can infer the other by applying rules of deduction. But we can at least show that they are equivalent through truth-tables. The following is the truth-table for (5).

\[
\begin{array}{cccc}
(P \lor Q) & \land & \neg(P \land Q) \\
1 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 1 \\
\end{array}
\]

The final result

7. The Biconditional

The biconditional \( \leftrightarrow \) is not a basic connective, because it can be decomposed to a conjunction of two conditionals, as shown by (6) below. Hence \( \leftrightarrow \) is just a convenient shorthand sign.

6. \( (P \leftrightarrow Q) = (P \rightarrow Q) \land (Q \rightarrow P) \)

\[
\begin{array}{cccc}
(P \rightarrow Q) & \land & (Q \rightarrow P) \\
1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 & 0 \\
\end{array}
\]

- 8 -
8. **Sufficient and Necessary Conditions**

\((A \rightarrow B)\) is equivalent to \((\sim B \rightarrow \sim A)\). This is called **contraposition**.

This, while a sufficient condition is represented as \((P \rightarrow Q)\), we can represent a necessary condition as \((Q \rightarrow P)\), which will be equivalent to \((\sim P \rightarrow \sim Q)\), which is exactly what we want to say in expressing a necessary condition.

9. **Using Negation and Disjunction to Define Conditional**

It can be proven that \((A \rightarrow B)\) is equivalent to \((\sim A \lor B)\). Hence the following should also obtain:

1. If you are tired of London, you are tired of life. \(P \rightarrow Q\)
2. If you are not tired of life, you are not tired of London. \(\sim Q \rightarrow \sim P\) (contraposition, from 1)
3. Either you are not tired of London, or you are tired of life. \(\sim P \lor Q\) (from 1)
4. Either you are not not tired of life, or you are not tired of London. (from 2)
5. Either you are tired of life, or you are not tired of London. (double negation, from 4)

---

**Semantics: Sample Test with answer key**

Time: 90 minutes

Answer all the questions.

1. **What is the difference between **denotation** and **reference**? Use at least one example to illustrate your explanations. (20%)**

   Ans: Denotation is the potential possibility of an expression in being related to some entity in the world. Reference is the actual use of an expression in an utterance to point to some entity in the world. Example: “the book”: its denotation can be described as “a unique, paged publication for reading”, while its use in the following sentence may refer to an actual book copy titled *Les Miserables*.

   Speaker A: I have always wanted to read the book but find it too long to finish.

2. **What is sense? Give one example to show how we can perceive the existence of sense. The example should not be any one already mentioned in lecture notes. (20%)**

   “Sense” is the meaning network of a language. The sense of a word is the portion of meaning segmented from the overall meaning network that is encapsulated in the linguistic unit (i.e. a slice of meaning taken from the meaning network that is encoded in a linguistic unit). Sense can be perceived through the study of special symmetric and asymmetric relations in the lexicon, sometimes through inter language comparisons. Examples: verbs related to cooking in Chinese sometimes have no equivalents in English, showing different semantic fields in the two languages.

3. **Why can the meaning of statement sentences be equated to truth-conditions? (20%)**

   Because the meaning of statement sentences can be equated to propositions. Propositions carry truth values. A proposition is true if it gives a true description of the event or state in the world. Otherwise, it is false. Therefore, the meaning of a proposition can be related to truth values. But it is not possible nor necessary for us to know
the exact truth value of a proposition in order to study its meaning. All we need to do is to exhaust its meaning possibilities by specifying under what conditions it is true. Once we spell out the truth conditions of a proposition, its literal meaning can be taken as adequately described. Hence the meaning of statement sentences can be equated to truth-conditions.

4. Compute the truth values of the following by using truth-tables:

a. $\neg P \rightarrow (P \rightarrow Q)$ (10%)

- $\neg P \rightarrow (P \rightarrow Q)$
  - 0 1 1 1 1 1
  - 0 1 1 1 0 0
  - 1 0 1 0 1 1
  - 1 0 1 0 1 0

b. $(P \land Q) \land \neg (P \leftrightarrow Q)$ (10%)

- $(P \land Q) \land \neg (P \leftrightarrow Q)$
  - 1 1 1 0 0 1 1
  - 0 0 0 0 0 1 0
  - 1 0 0 0 1 1 0
  - 0 0 1 0 1 0 0

5. Translate the following into propositional logic (10%)

Either you are mad or I am, but not both.

$(P \lor Q) \land \neg (P \land Q)$

6. Translate the following into an English sentence (10%)

$\forall x (\text{BOY}(x) \rightarrow \neg \text{LIKE}(x, \text{jim}))$.

Ans: No boy loves Jim.