

Mathematical Logic and Computability

Lecture 2: Syntax of First-Order Logic

Yunpyo An

Ulsan National Institute of Science and Technology

Mathematical Logic and Computability

Aug 28, 2023

Outline

Syntax of
First-Order Logic

Alphabets

Alphabets of First-Order
Logic

Terms and Formulas

Induction in the Calculi of
Terms and Formulas

Syntax of First-Order Logic

Alphabets

Alphabets of First-Order Logic

Terms and Formulas

Induction in the Calculi of Terms and Formulas

Definition

An *alphabet* \mathbb{A} is a finite set of **symbols**.

The finite sequences of symbols from an alphabet \mathbb{A} are called *strings* or *words*.

The \mathbb{A}^* denotes the set of all strings over \mathbb{A} .

Example

Suppose that $\mathbb{A} = \{a, b, c\}$ is an alphabet.

Then, \mathbb{A}^* is the set of all strings over \mathbb{A} , i.e.,

$$\mathbb{A}^* = \{\square, a, b, c, aa, ab, ac, ba, bb, bc, ca, cb, cc, \\ aaa, aab, aac, aba, abb, abc, aca, acb, acc, \dots\}$$

where \square is the *empty string*.

Lemma

For a nonempty set M the followings are equivalent:

- 1. M is at most countable. (i.e., M is finite or countably infinite.)*
- 2. There is an injective map $\beta : M \rightarrow \mathbb{N}$.*
- 3. There is a surjective map $\alpha : \mathbb{N} \rightarrow M$.*

Alphabets of First-Order Logic

Definition

The alphabet of a first-order language consists of the following symbols:

1. Variables: v_0, v_1, v_2, \dots
2. $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ (logical connectives)
3. \forall, \exists (quantifiers)
4. \equiv (equality)
5. $(,)$ (parentheses)
6. For every $n \geq 1$, a (possible empty) set of n -ary function symbols
7. For every $n \geq 1$, a (possible empty) set of n -ary relation symbols
8. a (possible empty) set of constant symbols

Let \mathbb{A} be the set of symbols 7 to 5. Let S be the set of symbols 6, 7, and 8. The set S determines the first-order language, for convenience, we denote \mathbb{A}_S as the alphabet of the first-order language.

Alphabets of First-Order Logic

Question: Assume that S is a countable set. Is \mathbb{A}_S countable?

Terms and Formulas

Ok, we define the alphabets of first-order logic.

We already know that not all strings over an alphabet are interpretable.

But, we don't define the meaning (semantics) of first-order logic yet.

We define the *syntax* of first-order logic. It is the grammar of first-order logic.

Definition

S -terms (T^S) are precisely those strings in A_S^* that can be obtained by applying the following rules:

1. Every variable is an S -term.
2. Every constant symbol is an S -term.
3. If the strings t_1, \dots, t_n are S -terms and f is an n -ary function symbol, then $f(t_1, \dots, t_n)$ is an S -term.

Definition

S-formulas (L^S) are precisely those strings in \mathbb{A}_S^\star that can be obtained by applying the following rules:

1. If t_1 and t_2 are S-terms, then $t_1 \equiv t_2$ is an S-formula.
2. If t_1, \dots, t_n are S-terms and R is an n -ary relation symbol, then $R(t_1, \dots, t_n)$ is an S-formula.
3. If ϕ is an S-formula, then $\neg\phi$ is an S-formula.
4. If ϕ and ρ are S-formulas, then $(\phi \wedge \rho)$, $(\phi \vee \rho)$, $(\phi \rightarrow \rho)$, and $(\phi \leftrightarrow \rho)$ are S-formulas.
5. If ϕ is an S-formula and x is variable, then $\forall x\phi$ and $\exists x\phi$ are S-formulas.

We might wonder why we need to define the syntax of first-order logic.

The syntax of first-order logic is construction of strings in first-order logic.

The terms and formulas are the different types of strings in first-order logic.

We borrow the idea from next lecture 'semantics' of first-order logic, the terms are the 'objects' and the formulas are the 'properties' of the objects.

Note that, we don't define the meaning of the terms and formulas yet. Keep in mind it.

Induction in the Calculi of Terms and Formulas

Let S be the set of symbols and $Z \subseteq A_S^*$ be the set of strings over the alphabet A_S .

We want to construct that from symbol set S to the terms T^S and formulas L^S .

We have the rules to construct the terms and formulas!

We can construct the terms and formulas by inductions. Let's the induction begin!

Assume that ζ_1, \dots, ζ_n all belong to Z . Then also ζ belongs to Z writing as following

$$\frac{\zeta_1, \dots, \zeta_n}{\zeta} \quad (1)$$

We call this rule as *inference rule*.

It allows $n = 0$, the first sort of rules in 5 and 6 is "premise-free" rules.

Induction in the Calculi of Terms and Formulas

The calculus **C** (rule) of terms T^S as follows:

$$\frac{\Box}{x} \quad (2)$$

$$\frac{\Box}{c}, c \in S \quad (3)$$

$$\frac{t_1, \dots, t_n}{f(t_1, \dots, t_n)}, f \in S, f \text{ is } n\text{-ary function symbol} \quad (4)$$

We can prove that some strings are terms by using the inference rules it is called *induction over calculus C*.

Induction in the Calculi of Terms and Formulas

The calculus **C** (rule) of formulas L^S as follows:

$$\frac{t_1, t_2}{t_1 \equiv t_2} \quad (5)$$

$$\frac{t_1, \dots, t_n}{R(t_1, \dots, t_n)}, R \in S, \quad R \text{ is } n\text{-ary relation symbol} \quad (6)$$

$$\frac{\phi}{\neg \phi}, \quad (7)$$

$$\frac{\phi, \rho}{(\phi \star \rho)}, \quad \text{where } \star = \vee, \wedge, \rightarrow, \leftrightarrow \quad (8)$$

$$\frac{\phi}{\forall x \phi}, \quad (9)$$

$$\frac{\phi}{\exists x \phi}, \quad (10)$$

Definition

The function var , which associates with each S -terms the set of variables occurring in it:

$$\text{var}(x) := \{x\}$$

$$\text{var}(c) := \emptyset$$

$$\text{var}(f(t_1, \dots, t_n)) := \text{var}t_1 \cup \dots \cup \text{var}t_n$$

Definition

The function SF , which assigns to each formula the set of its subformulas as following:

$$SF(t_1 \equiv t_2) := \{t_1 \equiv t_2\}$$

$$SF(R(t_1, \dots, t_n)) := \{R(t_1, \dots, t_n)\}$$

$$SF(\neg\phi) := \{\neg\phi\} \cup SF(\phi)$$

$$SF((\phi \star \rho)) := \{(\phi \star \rho)\} \cup SF(\phi) \cup SF(\rho) \quad \text{where } \star = \vee, \wedge, \rightarrow, \leftrightarrow$$

$$SF(\forall x\phi) := \{\forall x\phi\} \cup SF(\phi)$$

$$SF(\exists x\phi) := \{\exists x\phi\} \cup SF(\phi)$$

Definition

The function free , which assigns to each formula the set of its free variables as following:

$$\text{free}(t_1 \equiv t_2) := \text{var}(t_1) \cup \text{var}(t_2)$$

$$\text{free}(R(t_1, \dots, t_n)) := \text{var}(t_1) \cup \dots \cup \text{var}(t_n)$$

$$\text{free}(\neg\phi) := \text{free}(\phi)$$

$$\text{free}((\phi \star \rho)) := \text{free}(\phi) \cup \text{free}(\rho)$$

$$\text{free}(\forall x\phi) := \text{free}(\phi) \setminus \{x\}$$

$$\text{free}(\exists x\phi) := \text{free}(\phi) \setminus \{x\}$$

where $\star = \vee, \wedge, \rightarrow, \leftrightarrow$

For summary of this lecture, we have the following definitions:

- ▶ **Terms** T^S 5
- ▶ **Formulas** L^S 6
- ▶ **Induction over calculus C** 2 and 5
- ▶ **Subformulas** 8
- ▶ **Free variables** 9