Introduction to Mathematical Logic

For CS Students

CS104

Yida TAO (陶伊达)

2025 年 2 月 27 日

Table of Contents

A Logic Puzzle

1 Warm up

Alice, Alice's husband, their son, their daughter, and Alice's brother were involved in a murder. One of the five killed one of the other four. The following facts refer to the five people mentioned:

- A man and a woman were together in a bar at the time of the murder.
- The victim and the killer were together on a beach at the time of the murder.
- One of Alice's two children was alone at the time of the murder.
- Alice and her husband were not together at the time of the murder.
- The victim's twin was not the killer.
- The killer was younger than the victim.

Which one of the five was the victim? Write down your argument (论证).

What constitutes an argument?

- **Declarative sentences**: e.g., The killer was younger than the victim.
- **Common sense**: e.g., a father cannot be younger than his child; a parent and his or her child cannot be twins.
- **Inference rules**: e.g., if x, then ...... we have a contradiction. Therefore, x must not be true.

How to (mechanically) judge an argument is rigorous?

- To make arguments rigorous, we need to develop a language in which we can express sentences in such a way that brings out their logical structure.
- The language we begin with is the language of propositional logic.

Table of Contents

2 Propositions & Connectives

Definition

Propositional logic is based on **propositions**.

- **Proposition (命题)**: A proposition is a declarative sentence that can be judged as either true or false.
- **Atomic Proposition (原子命题)**: A proposition that does not contain any smaller part that is still a proposition is called an atomic proposition.

Which are atomic propositions?

- Snow is white.
- $2 + 2 = 5$
- What a big snow!
- Where are you going?
- $x + y < 0$
- This sentence is false.

Logical Connectives

Compound Proposition (复合命题): A proposition that involves the assembly of multiple propositions is called a compound proposition.

Words that connect multiple propositions to form a compound proposition are called **logical connectives**. For example:

- … and … (并且)
- not … (并非)
- … or … (或者)
- if … then … (如果… 那么…)
- … if and only if … (当且仅当)

Logical Connectives

Connectives	Read as	名称	Term
$\wedge$	and	合取	conjunction
$\vee$	or	析取	disjunction
$\neg$	not	否定	negation
$\rightarrow$	if...then...	蕴含	implication
$\leftrightarrow$	if and only if	等价	equivalence

Atom proposition	Compound proposition
p: Today is Friday.	Today is not Friday.
p: 2 is a prime number, q: 2 is an even number.	2 is both a prime and an even number.
p: Freshmen take Java classes, q: Freshmen take Python classes.	Freshmen take either Java or Python classes.
p: It will rain tomorrow, q: I will stay home and read.	If it rains tomorrow, then I will stay home and read.
p: A triangle is an isosceles triangle, q: A triangle has two equal angles.	A triangle is an isosceles triangle if and only if it has two equal angles.
p: 你是大一新生，q: 你能在寝室用电脑。	只有你不是大一新生，才能在寝室用电脑。

Table of Contents

3 From Symbols to Language

Languages in General

- All languages have a set of symbols, or alphabet(字母表)
  - For example, the letter “A” is part of English, but not part of Hindi.
- The syntax (语法) of a language is the grammar of the language.
  - For example, English is a Subject-Verb-Object language, while Arabic is Subject-Object-Verb.
- The semantics (语义) of a language is the meaning of the significant parts.
  - For example, “right” have different meanings in English,

We will introduce the formal language of propositional logic $\mathscr{L}^P$ in terms of:

- Syntax:
  - The alphabet
  - The rules governing the arrangement of symbols to form valid strings.
- Semantics: Truth values
- Inference rules and deduction systems

Table of Contents

Alphabet

<table>
<tr><th></th><th>symbols</th></tr>
<tr><td>binary</td><td>01 (or ab)</td></tr>
<tr><td>Roman</td><td>abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ</td></tr>
<tr><td>decimal</td><td>0123456789</td></tr>
<tr><td>special</td><td>~`!@#$%^&*()_-+={[}]|\:;"'<,>.?/</td></tr>
<tr><td>keyboard</td><td>Roman + decimal + special</td></tr>
<tr><td>genetic code</td><td>ATCG</td></tr>
<tr><td>protein code</td><td>ACDEFGHIKLMNPQRSTVWY</td></tr>
</table>

Alphabet of $\mathscr{L}^P$

$\mathscr{L}^P$, as the language of proposition logic, has 3 types of symbols.

- Atomic proposition (Atom): $p, q, r, \ldots p_1, r_2, \ldots$
- Logical connectives: $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$
- Punctuation: (and)

Expressions of $\mathscr{L}^P$

- Expressions (表达式): finite strings of symbols, e.g.,:
  - p
  - $pqqr$
  - $(\neg r)$
  - $p \wedge q \rightarrow r$
- The length of an expression is the number of occurrences of symbols in it.
- Two expressions u and v are equal if they are of the same length and have the same symbols in the same order.

Formulas of $\mathscr{L}^P$

The well-formed formulas (**wff**) of propositional logic are expressions which we obtain by using the construction rules below:

- Every atom (e.g., p) is a well-formed formula.
- If α is a well-formed formula, then so is $(\neg\alpha)$.
- If α and β are well-formed formulas, then so is $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \rightarrow \beta)$, and $(\alpha \leftrightarrow \beta)$. (braces matter!)
- Nothing else is a well-formed formula.

Definition 3.1 Atom($\mathscr{L}^P$)

Atom($\mathscr{L}^P$) is the set of expressions of $\mathscr{L}^P$ consisting of an atom proposition symbol only.

Definition 3.2 Form($\mathscr{L}^P$)

$Form(\mathscr{L}^P)$ is the smallest set of expressions that satisfies (i)∼(iii):

(i) Atom($\mathscr{L}^P$) $\subseteq$ Form($\mathscr{L}^P$)

(ii) If $\alpha \in$ Form($\mathscr{L}^P$), then $(\neg\alpha) \in$ Form($\mathscr{L}^P$).

(iii) If $\alpha, \beta \in$ Form($\mathscr{L}^P$), then $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \rightarrow \beta)$, $(\alpha \leftrightarrow \beta) \in$ Form($\mathscr{L}^P$).

Which are well-formed formulas in $\mathscr{L}^P$?

- $(\neg p)$
- $((q\vee$
- $((p\neg\wedge)q(r\neg$
- $(((\neg p) \leftrightarrow (q \vee r)) \rightarrow (r \wedge p))$

How do we prove that an expression is/isn't a wff?

Common properties of well-formed formula

Let's consider common properties of wff (if an expression doesn't have the property, then it is not a wff).

Lemma 3.1

Well-formed formulas of $\mathscr{L}^P$ are non-empty expressions.

Lemma 3.2

Every well-formed formula of $\mathscr{L}^P$ has an equal number of opening and closing brackets.

Prove Lemma 3.2 by inductive proof.

Common properties of well-formed formula

- Empty expression: an expression of length 0, denoted by λ.
- uv denotes the result of concatenating two expressions u, v in this order. Note that $\lambda u = u\lambda = u$.
- v is a segment of u if $u = w_1 v w_2$ where u, v, w_1, w_2 are expressions.
- v is a proper segment of u is v is non-empty and $v \neq u$.
- If $u = vw$, where u, v, w are expressions, then v is an initial segment (prefix) of u, w is a terminal segment (suffix) of u.
- If $u = vw$, where u, v, w are expressions, and v, w are non-empty, then v is an proper prefix of u, w is a proper suffix of u.

Common properties of well-formed formula

Lemma 3.3

- Every proper prefix of a well-formed formula in $\mathscr{L}^P$ has more opening brackets than closing brackets.
- Similarly, every proper suffix of a well-formed formula in $\mathscr{L}^P$ has more closing brackets than opening brackets.
- Hence, proper prefix and proper suffix are not wff in $\mathscr{L}^P$ (Lemma 3.2).

Proof of Lemma 3.3 by (structural) induction:
Let $P(\varphi)$ be the property that every proper prefix of the well-formed formula φ has more opening brackets than closing brackets. We prove $P(\varphi)$ is true for all well-formed formulas φ by structural induction.

Base Case: If φ is an atom, then there are no proper prefixes and the claim is vacuously true.

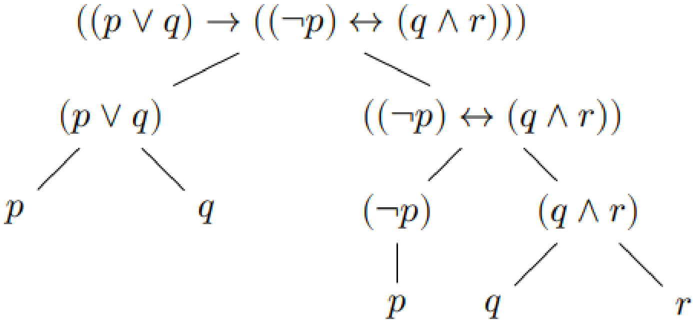
Inductive Hypothesis: Assume that $P(\alpha)$ and $P(\beta)$ are true for some well-formed formulas α and β.

Complete the induction step by yourself.

Are common properties sufficient to check wff?

Constructing a wff

If an expression can be **constructed** by the **construction rules** of wff, then it is a wff.



Parse Tree

Definition 3.3

A parse tree of a formula in $\mathscr{L}^P$ is a tree such that

- The root is the formula.
- Leaves are atoms, and
- Each internal node is formed by applying some formation rule on its children.

Theorem 3.1

An expression of $\mathscr{L}^P$ is a well-formed formula if and only if there is a parse tree of it.

Once we have the parse tree, we could construct the wff from bottom to top.

Parse Tree

Definition 3.4

A formula G is a subformula of formula F if G occurs within F. G is a proper subformula of F if $G \neq F$.
The nodes of the parse tree of F form the set of subformulas of F.

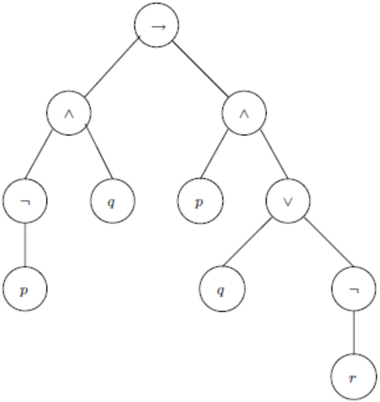
Definition 3.5

Immediate subformulas are the children of a formula in its parse tree, and *leading connective* is the connective that is used to join the children.

Q: The subformulas and leading connective for $((\neg p) \leftrightarrow (q \wedge r))$?

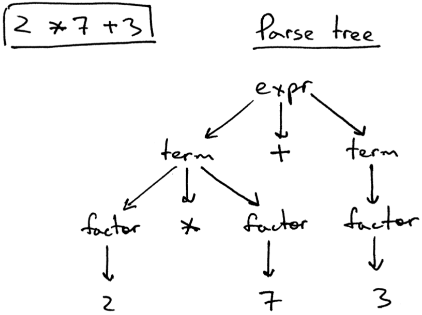
Parse Tree

We can simplify a parse tree by highlighting only the leading connectives and the atom propositions (leaves). Example: $(((\neg p) \land q) \to (p \land (q \lor (\neg r))))$

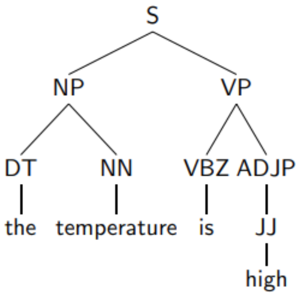


Applications

A parse tree is created by a parser, which is a component of a compiler that processes the source code and checks it for syntactic correctness.



In NLP, a parse tree can be used to computationally reason about a natural-language sentence.



Unique Readability Theorem

There is a unique way to construct every well-formed formula.

Proof by induction

Let $P(\varphi)$ be the property that there is a unique way to construct the well-formed formula φ. We prove this property for all well-formed formulas φ by structural induction.

Base case: There is only one way to construct an atom.

Inductive Hypothesis: Assume that $P(\alpha)$ and $P(\beta)$ are true for some well-formed formulas α and β.

Constructing a wff

Unique Readability Theorem

There is a unique way to construct every well-formed formula.

Proof by induction (continued)

Inductive Step: Consider two possibilities:

- $\varphi = (\neg\alpha)$
- $\varphi = (\alpha \star \beta)$

(Proved in class)

Conventions

For readability, we use the following convention, which states that we may drop the outermost parentheses.

If F or (F) is a formula, then we view F and (F) as the same formula.

But, how do we interpret formulas like: $p \wedge q \rightarrow \neg r \vee q$?

Precedence

Parentheses are used to resolve ambiguity. But they are hard to read.
If no parentheses are present, we could use **precedence** and **associativity** to disambiguate formulas.

- Each connective on the left has priority over those on the right. $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
- Parentheses take the highest precedence.
- Connectives are assumed to associate to the right (right associative), i.e., first group the rightmost occurrence. For example, $p \rightarrow q \rightarrow r$ means $p \rightarrow (q \rightarrow r)$

Add back the brackets based on the precedence rules.

- $\neg p \to q$
- $p \wedge q \to r$
- $p \wedge q \to \neg r \vee q$
- $\neg p \to p \wedge \neg q \vee r \leftrightarrow q$

Readings

Optional

- TextA: Chapter 2
- TextB: Chapter 1.3
- Text1: 第二章 2.1, 2.2, 2.3
- Text3: 第二章 2.1, 2.2, 2.4

Assignments

Coursework

- Assignment 2

Introduction to Mathematical Logic

Thank you for listening!
Any questions?