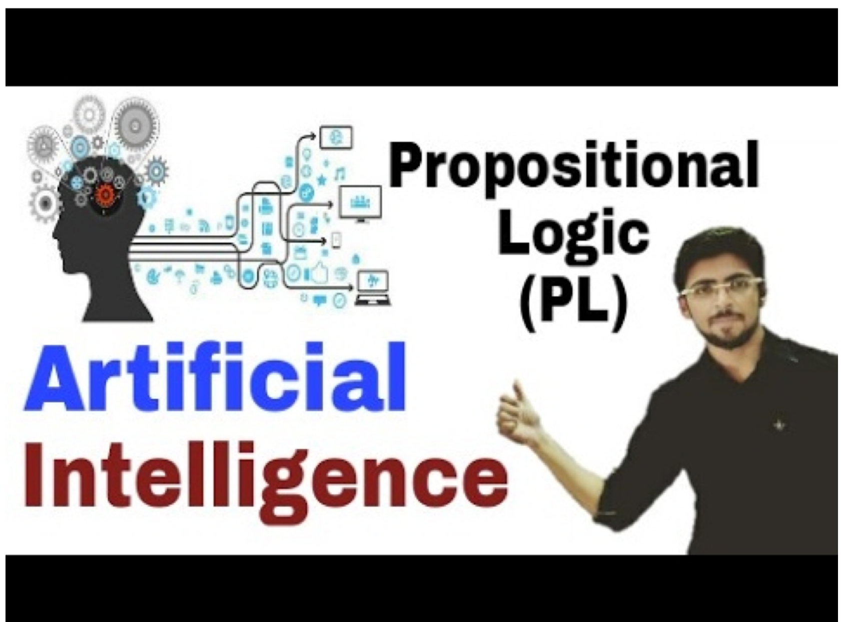
Lecture 2: Propositional Logic

Dr. Chengjiang Long
Computer Vision Researcher at Kitware Inc.
Adjunct Professor at SUNY at Albany.
Email: **clong2@albany.edu**

Introduction: Logic?

- Logic
  1) is the study of the logic relationships between objects
  2) forms the basis of all mathematical reasoning and all automated reasoning



Introduction: PL?

- In Propositional Logic (a.k.a Propositional Calculus or Sentential Logic), the objects are called propositions
- **Definition**: A proposition is a statement that is either true or false, but not both
- We usually denote a proposition by a letter: *p*, *q*, *r*, *s*, …

Outline

- Defining Propositional Logic
- Precedence of Logical Operators
- Usefulness of Logic
- Logical Equivalences

Outline

- **Defining Propositional Logic**
- Precedence of Logical Operators
- Usefulness of Logic
- Logical Equivalences

Introduction: Proposition

- **Definition**: The value of a proposition is called its truth value; denoted by
  - *T* or 1 if it is true or
  - *F* or 0 if it is false
- Opinions, interrogative, and imperative are not propositions
- **Truth table**

p
0
1

Propositions

- Propositional logic operates with statements. Statements could be true or false and are called **propositions**.
- *Is the sentence proposition?*

Richmond is the capital of Virginia.	Yes (True)
$2 + 3 = 7$.	Yes (False)
Open the door.	No
$5 + 7 < 10$.	Yes (False)
The moon is a satellite of the earth.	Yes (True)
$x + 5 = 7$.	No
$x + 5 > 9$ for every real number x.	Yes (False)

Propositions: Examples

- The following are propositions
  - Today is Monday
  - The grass is wet
  - It is raining
- The following are not propositions
  - C++ is the best language — *Opinion*
  - When is the pretest? — *Interrogative*
  - Do your homework — *Imperative*

Are these propositions?

- 2+2=5
- Every integer is divisible by 12
- Microsoft is an excellent company

Logical connectives

- Connectives are used to create a compound proposition from two or more propositions
  - Negation (denote ~ or $\neg$ or !) $\neg
  - And or logical conjunction (denoted $\wedge$) $\wedge
  - Or or logical disjunction (denoted $\vee$) $\vee
  - XOR or exclusive or (denoted $\oplus$) $\xor
  - Implication (denoted $\Rightarrow$ or $\rightarrow$)

    $\Rightarrow, $\rightarrow
  - Biconditional (denoted $\Leftrightarrow$ or $\leftrightarrow$)

    $\LeftRightarrow, $\leftrightarrow
- We define the meaning (semantics) of the logical connectives using <u>truth tables</u>

Logical Connective: Negation

- $\neg p$, the negation of a proposition p, is also a proposition
- Examples:
  - Today is not Monday
  - It is not the case that today is Monday, etc.

p	$\neg p$
0	1
1	0

Logical Connective: Logical And

- The logical connective And is true only when both of the propositions are true. It is also called a <u>conjunction</u>
- Examples
  - It is raining and it is warm
  - (2+3=5) and (1<2)
  - Schroedinger's cat is dead and Schroedinger's is not dead.

p	***q***	$p \wedge q$
0	0	
0	1	
1	0	
1	1	

Logical Connective: Logical Or

- The logical disjunction, or logical Or, is true if one or both of the propositions are true.
- Examples
  - It is raining or it is the second lecture
  - $(2+2=5) \vee (1<2)$
  - You may have cake or ice cream

p	q	$p \wedge q$	$p \vee q$
0	0	0	
0	1	0	
1	0	0	
1	1	1	

Logical Connective: Exclusive Or

- The exclusive Or, or XOR, of two propositions is true when exactly one of the propositions is true and the other one is false
- Example
  - The circuit is either ON or OFF but not both
  - Let $ab<0$, then either $a<0$ or $b<0$ but not both
  - You may have cake or ice cream, but not both

p	q	$p \wedge q$	$p \vee q$	$p \oplus q$
0	0	0	0	
0	1	0	1	
1	0	0	1	
1	1	1	1	

Logical Connective: Implication (1)

- **Definition:** Let *p* and *q* be two propositions. The implication $p \rightarrow q$ is the proposition that is false when *p* is true and *q* is false and true otherwise
  - *p* is called the hypothesis, antecedent, premise
  - *q* is called the conclusion, consequence

p	q	$p \wedge q$	$p \vee q$	$p \oplus q$	$p \Rightarrow q$
0	0	0	0	0	
0	1	0	1	1	
1	0	0	1	1	
1	1	1	1	0	

Logical Connective: Implication (2)

- The implication of $p \rightarrow q$ can be also read as
  - If *p* then *q*
  - *p* implies *q*
  - If *p*, *q*
  - *p* only if *q*
  - *q* if *p*
  - *q* when *p*
  - *q* whenever *p*
  - *q* follows from *p*
  - *p* is a sufficient condition for *q* (*p* is sufficient for *q*)
  - *q* is a necessary condition for *p* (*q* is necessary for *p*)

Logical Connective: Implication (3)

- Examples
  - If you buy you air ticket in advance, it is cheaper.
  - If x is an integer, then $x^2 \geq 0$.
  - If it rains, the grass gets wet.
  - If the sprinklers operate, the grass gets wet.
  - If 2+2=5, then all unicorns are pink.

Exercise: Which of the following implications is true?

- If -1 is a positive number, then 2+2=5

True. The premise is obviously false, thus no matter what the conclusion is, the implication holds.

- If -1 is a positive number, then 2+2=4

True. Same as above.

- If $\sin x = 0$, then $x = 0$

False. x can be a multiple of π. If we let $x=2\pi$, then $\sin x=0$ but $x\neq0$. The implication "if $\sin x = 0$, then $x = k\pi$, for some k" is true.

Logical Connective: Biconditional (1)

- **Definition:** The biconditional $p \leftrightarrow q$ is the proposition that is true when p and q have the same truth values. It is false otherwise.
- Note that it is equivalent to $(p \rightarrow q) \wedge (q \rightarrow p)$

p	q	$p \wedge q$	$p \vee q$	$p \oplus q$	$p \Rightarrow q$	$p \Leftrightarrow q$
0	0	0	0	0	1	
0	1	0	1	1	1	
1	0	0	1	1	0	
1	1	1	1	0	1	

Logical Connective: Biconditional (2)

- The biconditional $p \leftrightarrow q$ can be equivalently read as
  - p if and only if q
  - p is a necessary and sufficient condition for q
  - if p then q, and conversely
  - p iff q (Note typo in textbook, page 9, line 3)
- Examples
  - $x>0$ if and only if x^2 is positive
  - The alarm goes off iff a burglar breaks in
  - You may have pudding iff you eat your meat

Exercise: Which of the following biconditionals is true?

- $x^2 + y^2 = 0$ if and only if $x=0$ and $y=0$

  True. Both implications hold

- $2 + 2 = 4$ if and only if $\sqrt{2}<2$

  True. Both implications hold.

- $x^2 \geq 0$ if and only if $x \geq 0$

  False. The implication "if $x \geq 0$ then $x^2 \geq 0$" holds.
  However, the implication "if $x^2 \geq 0$ then $x \geq 0$" is false.
  Consider $x=-1$.
  The hypothesis $(-1)^2=1 \geq 0$ but the conclusion fails.

Converse, Inverse, Contrapositive

- For the proposition $P \longrightarrow Q$,
  - the proposition $\neg P \longrightarrow \neg Q$ is called its **inverse**,
  - the proposition is $Q \longrightarrow P$ called its **converse**,
  - the proposition $\neg Q \longrightarrow \neg P$ is called its **contrapositive**.

- The inverse and converse of a proposition are not necessarily logically equivalent to the proposition.

- The contrapositive of a proposition is always logically equivalent to the proposition.

Converse, Inverse, Contrapositive

- **Example:** for the proposition "If it rains, then I get wet",
  - Inverse: If does not rain, then I don’t get wet.
  - Converse: If I get wet, then it rains.
  - Contrapositive: If I don't get wet, then it does not rain.

- Therefore, "If it rains, then I get wet." and "If I don't get wet, then it does not rain." are logically equivalent. If one is true then the other is also true, and vice versa.

Truth Tables

- Truth tables are used to show/define the relationships between the truth values of
  - the individual propositions and
  - the compound propositions based on them

p	q	$p \wedge q$	$p \vee q$	$p \oplus q$	$p \Rightarrow q$	$p \Leftrightarrow q$
0	0	0	0	0	1	1
0	1	0	1	1	1	0
1	0	0	1	1	0	0
1	1	1	1	0	1	1

Constructing Truth Tables

- Construct the truth table for the following compound proposition

$$((p \wedge q) \vee \neg q)$$

p	q	$p \wedge q$	$\neg q$	$((p \wedge q) \vee \neg q)$
0	0	0	1	1
0	1	0	0	0
1	0	0	1	1
1	1	1	0	1

Outline

- Defining Propositional Logic
- **Precedence of Logical Operators**
- Usefulness of Logic
- Logical Equivalences

Precedence of Logical Operators

- As in arithmetic, an ordering is imposed on the use of logical operators in compound propositions
- However, it is preferable to use parentheses to disambiguate operators and facilitate readability

$$\neg p \vee q \wedge \neg r \equiv (\neg p) \vee (q \wedge (\neg r))$$

- To avoid unnecessary parenthesis, the following precedences hold:
  1. Negation ($\neg$)
  2. Conjunction ($\wedge$)
  3. Disjunction ($\vee$)
  4. Implication ($\rightarrow$)
  5. Biconditional ($\leftrightarrow$)

Outline

- Defining Propositional Logic
- Precedence of Logical Operators
- **Usefulness of Logic**
- Logical Equivalences

Usefulness of Logic

- Logic is more precise than natural language
  - You may have cake or ice cream.
  - Can I have both?
  - If you buy your air ticket in advance, it is cheaper.
  - Are there or not cheap last-minute tickets?

- For this reason, logic is used for hardware and software specification
  - Given a set of logic statements, one can decide whether or not they are satisfiable (i.e., consistent), although this is a costly process…

Bitwise Operations

- Computers represent information as bits (binary digits)
- A bit string is a sequence of bits
- The length of the string is the number of bits in the string
- Logical connectives can be applied to bit strings of equal length
- Example

	0110 1010 1101
	0101 0010 1111
	_______________
Bitwise OR	0111 1010 1111
Bitwise AND	...
Bitwise XOR	…

Logic in Programming: Example 1

- Say you need to define a conditional statement as follows:
  - Increment x if all of the following conditions hold: $x > 0$, $x < 10$, $x=10$
- You may try: `If (0<x<10 OR x==10) x++;`
- But this is not valid in C++ or Java. How can you modify this statement by using logical equivalence
- Answer: `If (x>0 AND x<=10) x++;`

Logic in Programming: Example 2

- Say we have the following loop

```
While
   ((i<size AND A[i]>10) OR
    (i<size AND A[i]<0) OR
    (i<size AND (NOT (A[i]!=0 AND NOT (A[i]>=10)))))
```

- Is this a good code? Keep in mind:
  - Readability
  - Extraneous code is inefficient and poor style
  - Complicated code is more prone to errors and difficult to debug
  - Solution? Comes later…

Outline

- Defining Propositional Logic
- Precedence of Logical Operators
- Usefulness of Logic
- **Logical Equivalences**

Propositional Equivalences: Introduction

- To manipulate a set of statements (here, logical propositions) for the sake of mathematical argumentation, an important step is to replace one statement with another equivalent statement (i.e., with the same truth value)
- Below, we discuss:
  - Terminology
  - Establishing logical equivalences using truth tables
  - Establishing logical equivalences using known laws (of logical equivalences)

Terminology

- Definitions
  - A compound proposition that is always true, no matter what the truth values of the propositions that occur in it is called a tautology
  - A compound proposition that is always false is called a contradiction
  - A proposition that is neither a tautology nor a contradiction is a contingency
- Examples
  - A simple tautology is $p \vee \neg p$
  - A simple contradiction is $p \wedge \neg p$

Logical Equivalences: Definition

- **Definition**: Propositions p and q are logically equivalent if $p \leftrightarrow q$ is a tautology.
- Informally, p and q are equivalent if whenever p is true, q is true, and vice versa
- Notation: $p \equiv q$ (p is equivalent to q), $p \leftrightarrow q$, and $p \Leftrightarrow q$
- Alert: $\equiv$ is not a logical connective $\equiv$

Logical Equivalences: Example 1

- Are the propositions $(p \rightarrow q)$ and $(\neg p \vee q)$ logically equivalent?
- To find out, we construct the truth tables for each:

p	q	$p \rightarrow q$	$\neg p$	$\neg p \vee q$
0	0			
0	1			
1	0			
1	1			

The two columns in the truth table are identical, thus we conclude that

$$(p \rightarrow q) \equiv (\neg p \vee q)$$

Logical Equivalences: Example 1

- Show that $(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$

p	q	r	$p \to r$	$q \to r$	$(p \to r) \vee (q \to r)$	$p \wedge q$	$(p \wedge q) \to r$
0	0	0					
0	0	1					
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

Logical Equivalences: Cheat Sheet

Identities (Equivalences)	Name
$p \wedge T \equiv p$ $p \vee F \equiv p$	Identity laws
$p \vee T \equiv T$ $p \wedge F \equiv F$	Domination laws
$p \vee p \equiv p$ $p \wedge p \equiv p$	Idempotent laws
$\neg(\neg p) \equiv p$	Double negation law
$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$	Commutative laws
$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	Associative laws
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	Distributive laws
$\neg(p \wedge q) \equiv \neg p \vee \neg q$ $\neg(p \vee q) \equiv \neg p \wedge \neg q$	De Morgan's laws
$p \vee (p \wedge q) \equiv p$ $p \wedge (p \vee q) \equiv p$	Absorption laws
$p \vee \neg p \equiv T$ $p \wedge \neg p \equiv F$	Negation laws

Logical Equivalences Involving Conditional Statements.
$p \to q \equiv \neg p \vee q$ $p \to q \equiv \neg q \to \neg p$ $p \vee q \equiv \neg p \to q$ $p \wedge q \equiv \neg(p \to \neg q)$ $\neg(p \to q) \equiv p \wedge \neg q$ $(p \to q) \wedge (p \to r) \equiv p \to (q \wedge r)$ $(p \to r) \wedge (q \to r) \equiv (p \vee q) \to r$ $(p \to q) \vee (p \to r) \equiv p \to (q \vee r)$ $(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$

Next class

- Topic: Predicate Logic and Quantifies
- Pre-class reading: Chap 1.3-1.4