University of Malta
BSc(Hons.) IT Year I

Introduction to Logic and Logic Programming

Lecture 3

Propositional Calculus

- Negation, Conjunction and Disjunction
- Logical Equivalence
- Negative Normal Form
- Translation
- Satisfiability, Logical Truth

Department of Computer Science and AI
1988/9

Logical Connectives: Negation

- **Syntax**

  If P is any formula then -P is its negation.

- **Semantics**

  If P makes a true claim, then -P makes a false one. If P makes a false claim, then -P makes a true one.

  We can summarise the semantics using a truth table.

P	-P
t	f
f	t

Logical Connectives Conjunction/Disjunction

- **Syntax**

  If P and Q are formulas, then $(P \wedge Q)$ is their conjunction and $(P \vee Q)$ is their disjunction.

- **Semantics**

P	Q	$(P \wedge Q)$
t	t	t
t	f	f
f	t	f
f	f	f

P	Q	$(P \vee Q)$
t	t	t
t	f	t
f	t	t
f	f	f

N.B Note that disjunction is inclusive not exclusive.

Exclusive Disjunction

Client: Coke please.
Server: Regular, large or giant?

How can we translate the available options:

$$R \vee L \vee G$$

No good because consistent with all more than one option, unlike

$$(R \wedge -L \wedge -G) \vee$$
$$(-R \wedge L \wedge -G) \vee$$
$$(-R \wedge -L \wedge G)$$

Conjunction/Disjunction Commutativity and Associativity

P	Q	$(P \wedge Q)$
t	t	t
t	f	f
f	t	f
f	f	f

P	Q	$(P \vee Q)$
t	t	t
t	f	t
f	t	t
f	f	f

- By looking at the truth tables for conjunction and disjunction we can easily establish that they are *commutative*, i.e. that

  $(P \wedge Q)$ has the same truth value as $(Q \wedge P)$
  $(P \vee Q)$ has the same truth value as $(Q \vee P)$

- For any of the eight possible truth assignments to propositional variables P, Q, R, conjunction and disjunction are *associative*, i.e.

  - (P & Q) & R has the same truth value as P & (Q & R)
  - (P v Q) v R has the same truth value as P v (Q v R)

Commutativity and Associativity

These properties imply that expressions like

$$(P \mathbin{\&} Q) \mathbin{\&} (R \mathbin{\&} S)$$

$$(P \vee Q \vee R) \vee S$$

can unambiguously be written without parentheses and with the individual propositions (conjuncts or disjuncts) in any order.

N.B. This is not the case when the expression contains a mixture of connectives. The following expressions

$$(P \mathbin{\&} Q) \vee R$$

$$P \mathbin{\&} (Q \vee R)$$

$$(P \mathbin{\&} R) \vee Q$$

yield different results for given values of P, Q, and R.

Logical Equivalence

Certain pairs of formulas have identical truth tables, e.g.

P	Q	-(P v Q)	-P & -Q
t	t	f	f
t	f	f	f
f	t	f	f
f	f	t	t

To signify that two formulas have the same truth table we will use the operator $\equiv$. Thus

$$-(P \vee Q) \equiv -P \mathbin{\&} -Q$$

N.B: The operator $\equiv$ is *not* part of the logic. It is part of the *meta-language* that we use that we use *to talk about* sentences in the logic. We say that the logic sentences are part of the *object language*.

Meta Language and Object Language

The distinction between meta language and object language also crops up in English. Consider the sentence:

The equation "$x^2 + y^2 = 0$" describes a circle.

Here again, two languages are in operation: English (the whole sentence) and Mathematics (between quotes). The English sentence (the meta-language) makes a claim about the Mathematics sentence (the object language).

Exercise: identify the object and meta languages in the following sentence:

For any formulas P, Q of the propositional calculus, the metastatement P $\equiv$ Q means simply that they have the same truth table.

Negation Normal Form

For any sentences P and Q

- **Double Negation:** --P $\equiv$ P
- **DeMorgan:** -(P $\wedge$ Q) $\equiv$ (-P $\vee$ -Q)
- **DeMorgan:** -(P $\vee$ Q) $\equiv$ (-P $\wedge$ -Q)

Using these three laws, any sentence can be transformed into a logically equivalent sentence where the - applies only to atomic sentences.

To obtain such a sentence, one simply drives negation inwards, using DeMorgan's laus as appropriate, and eliminating double negations as they occur. Such a sentence is said to be in *negation normal form*

Example of a Derivation

$$\begin{aligned} \text{-((A} \vee \text{B)} \wedge \text{-C)} &\equiv \text{-(A} \vee \text{B)} \vee \text{--C} \\ &\equiv \text{(-A} \wedge \text{-B)} \vee \text{--C} \\ &\equiv \text{(-A} \wedge \text{-B)} \vee \text{C} \end{aligned}$$

Translation between English and FOL

What makes a translation between E, and English sentence, and L, a FOL sentence, correct?

L and E must have the same truth conditions. In other words, L must be true *in exactly the same circumstances* as E.

This is not to be confused with being true in some particular circumstance. Otherwise any true sentence of English could be translated by any other true sentence of FOL. If this were the case, then if Claire and Max are both hungry right now, we might translate "Claire is hungry" as hungry(Max).

style

Good translations are also in part a question of style. To translate “it is not true that Claire and Max are hungry”, (a) is better than (b), even though the two are logically equivalent.

(a) -(Hungry(Claire) $\wedge$ Hungry(Max))
(b) -Hungry(Claire) $\vee$ -Hungry(Max)

An Ontology

English	FOL
Names:	
Max	Max
Claire	Claire
Folly	Folly
2pm	2:00
2.01pm	2:01
Predicates:	
x is a disk	Disk(x)
x is a person	Person(x)
x is a student	Student(x)
t is earlier than t'	$t < t'$
x was blank at time t	Blank(x,t)
x was angry at time t	Angry(x,t)
x owned y at time t	Owned(x,t)
x gave y to z at time t	Gave(x,y,t)
x erased y at time t	Erased(x,y,t)

Examples

1. Max is a student, not a disk
2. Claire erased Folly at 2pm and then gave it to Max.
3. 2:00pm is between 1:55pm and 2:05pm
4. When Max gave Folly to Claire at 2pm, it wasn't blank, but it was five minutes later.

Satisfiability

A sentence is satisfiable if there is some logically possible (cf. physically possible) circumstance under which it could be true. For example

> John runs faster than light

is logically satisfiable since it is logically possible to conceive of circumstances under which John runs faster than light. That this is physically impossible in our universe is irrelevant.

Satisfiability of a Set of Sentences

The idea can be extended to a *set* of sentences. The set of sentences is satisfiable is there are circumstances under which they are simultaneously true. For instance, there are circumstances that make any two of the sentences below true, but none that make all of them true.

1. Happy(Claire) v Happy(Max)
2. -Happy(Claire)
3. -Happy(Max)

So the set is not satisfiable.

Logical Truth

A related notion is that of *logically true sentence*. A logically true sentence is one that is true regardless of the circumstances, e.g.

Hungry(Mike) v -Hungry(Mike)
-(Hungry(Mike) $\wedge$ -Hungry(Mike))

A sentence P is logically true if and only if -P is unsatisfiable

A sentence P is satisfiable if and only if -P is not logically true