Bad Arguments: Translation from English and Other Issues

Specifying systems

Specifying the behaviour of complex systems is a fundamental skill for computer science professionals.

- Specifying access to computer systems
- Specify desired behaviour for a software module
- Specifying a protocol for two programs to communicate

Logic is the main technique for capturing and analysing these specifications.

If we can get these specifications written in a logical form, we can properly analyse and test them.

Translating from English to conditional statements

We may translate sentences of the following type using $p \rightarrow q$:

- if p then q
- p implies q
- p only if q (?)
- p is sufficient for q
- q follows from p
- q whenever p
- q when p
- q is necessary for p

Problems with translating from English

The language is not precise, it contains ambiguities.

We must ensure when translating that we capture the intent of the original sentence.

Three main problems:

- unclear precedence and association
- inclusive vs. exclusive OR
- conditional sentences

The contrapositive

¬q → ¬p is the contrapositive of p→q.

They have the same truth table, and so are logically equivalent.

A conditional statement is true if and only if its contrapositive is true.

The converse

q→p is the converse of p→q.

These two are not logically equivalent, as their truth tables differ.

The inverse

¬p → ¬q is the inverse of p → q.

These are not logically equivalent, as their truth tables differ.

Conditional statements and equivalents

p→q is equivalent to:

- the contrapositive
- ¬p ∨ q

but is not equivalent to:

- the inverse
- the converse

Example: If I am in Cork then I am in Ireland.

p = I am in Cork

q = I am in Ireland

F "If I am in Ireland, then I am in Cork"

T "Either I am not in Cork or I am in Ireland" (weird)

F "If I am not in Cork, then I am not in Ireland"

T "If I am not in Ireland, then I am not in Cork"

Logical fallacies

A logical fallacy is a bad argument—a sequence of statements that do not guarantee that the conclusion is true whenever the premises are true.

It is important to spot these in order to be a better computer scientist.