lecture 03, the formal language PL

phil1012 introductory logic

overview

this lecture

- an introduction to the formal language PL
- the vocabulary and syntax of PL
- how formulas of PL are constructed
- translating propositions from English into PL with the use of a glossary

learning outcomes

- after doing the relevant reading for this lecture, listening to the lecture, and attending the relevant tutorial, you will be able to:
  - explain what the different symbols of PL mean
  - translate PL formulas into English
  - translate propositions from English into PL, providing a glossary for your translation
  - identify well-formed formulas of PL
  - construct construction tables for formulas of PL

required reading

- all of chapter 2

the language of propositional logic (PL)

the language of propositional logic (PL)

- the formal symbolic language of propositional logic (PL)
- the language of PL consists of a **vocabulary** and a **syntax**
- the vocabulary concerns the basic symbols
- the syntax concerns how basic symbols can be put together to form complex symbols

the vocabulary of PL

the vocabulary of PL

- basic propositions are represented in PL by simple capital letters with or without numerical subscripts
- they are called **sentence letters**
  - A,A2,A3,…B,B2,B3,…,C,C2,C3,…,Z,Z2,Z3A, A_2, A_3, \ldots B, B_2, B_3, \ldots, C, C_2, C_3, \ldots, Z, Z_2, Z_3

- suppose you want to use ‘A’ to represent the proposition expressed by the sentence ‘John is short’
- in order to do this you would write a **glossary** like this:
    - AA: John is short

- we use sentence letters only for basic propositions. so the following is not okay:
    - AA: John is short and Jane is tall
- remember that a negation is not a basic proposition. so the following is not okay:
    - AA: John is not short

- we have introduced symbols to represent propositions (sentence letters)
- now we introduce symbols to represent the **connectives**
    - ¬,∧,∨,→,↔\lnot, \land, \lor, \rightarrow, \leftrightarrow

conjunction

- the connective conjunction is represented in PL by the symbol:
    - ∧\land
- the name of the symbol is **caret**

- we can use caret to represent the conjunction of AA and BB as follows:
    - (A∧B)(A \land B)
- we can use this to represent the proposition that John is short and Jane is tall

disjunction

- the connective disjunction is represented in PL by the symbol:
    - ∨\lor
- the name of the symbol is **vel**

- we can use vel to represent the disjunction of AA and BB as follows:
    - (A∨B)(A \lor B)
- we can use this to represent the proposition that John is short or Jane is tall

conditional

- the connective conditional is represented in PL by the symbol:
    - →\rightarrow
- the name of the symbol is **arrow**

- we can use arrow to represent a conditional with AA as its antecedent and BB as its consequent as follows:
    - (A→B)(A \rightarrow B)
- we can use this to represent the proposition that if John is short, then Jane is tall

biconditional

- the connective biconditional is represented in PL by the symbol:
  - ↔\leftrightarrow
- the name of the symbol is **double arrow**

- we can use double arrow to represent a biconditional with AA as its left-hand side and BB as its right-hand side as follows:
  - (A↔B)(A \leftrightarrow B)
- we can use this to represent the proposition that John is short if and only if Jane is tall

negation

- the connective negation is represented in PL by the symbol:
  - ¬\lnot
- the name of the symbol is **neg**

- we can use neg to represent a negation with AA as its negand as follows:
  - ¬A\lnot A
- we can use this to represent the proposition that John is not tall

the connectives of PL

- here are the symbols for the connectives in PL:

connective	name	symbol
negation	neg	¬\lnot
conjunction	caret	∧\land
disjunction	vel	∨\lor
conditional	arrow	→\rightarrow
biconditional	double arrow	↔\leftrightarrow

- here are some alternative symbols for the connectives

connective	symbol
negation	~
conjunction	&
disjunction	v
conditional	->
biconditional	<->

- note! although the choice of symbols for the connectives is entirely arbitrary, the choice we have made is final, and henceforth we will only use these symbols

---

- we now have the most important parts of the vocabulary of PL: sentence letters, connective symbols, and punctuation symbols '(' and ')'
- now we need to say how these can be put together

the syntax of PL

the syntax of PL

- we are now ready to precisely state the syntax of PL
- see the handout on the syntax of PL

- we use lowercase Greek letters like α\alpha and β\beta to express general claims about the formulas of PL
- these are called **well-formed formula variables**
- well-formed formula variables can stand in for single sentence letters like AA or for complex formulas like (A∧B)(A \land B)
- they are not part of the language of PL

- the symbols of PL are:
  - sentence letters (basic propositions):
    - A,A2,A3,…B,B2,B3,…,C,C2,C3,…,Z,Z2,Z3A, A_2, A_3, \ldots B, B_2, B_3, \ldots, C, C_2, C_3, \ldots, Z, Z_2, Z_3
  - connectives:
    - ¬∧∨→↔\lnot \hspace{6pt} \land \hspace{6pt} \lor \hspace{6pt} \rightarrow \hspace{6pt} \leftrightarrow
  - punctuation symbols (parentheses):
    - ()(\hspace{12pt})

- wffs of PL are defined as follows:
  - any basic proposition is a wff
  - if α\alpha and β\beta are wffs, then so are:
    - ¬α\lnot \alpha
    - (α∧β)(\alpha \land \beta)
    - (α∨β)(\alpha \lor \beta)
    - (α→β)(\alpha \rightarrow \beta)
    - (α↔β)(\alpha \leftrightarrow \beta)
  - nothing else is a wff

- the syntax of PL is specified by a **recursive definition**
- the **base clause** of the definition states certain things are wffs
- the **recursive clause** states that certain other things are wffs in terms of things we already know to be wffs

---

- all of the following can be generated using this definition:
  - ((R→P)↔R)((R\rightarrow P)\leftrightarrow R)
  - (B↔((C∧A)∨C))(B\leftrightarrow ((C\land A)\lor C))
  - ¬(B∨(B↔A))\lnot (B\lor (B\leftrightarrow A))
  - ((H→G)→(H↔(G∨H)))((H\rightarrow G)\rightarrow (H\leftrightarrow (G\lor H)))
  - ((Z∨(((Y∨Y)∧(Z→Z))∨Y))∨Y)((Z\lor (((Y\lor Y)\land (Z\rightarrow Z))\lor Y))\lor Y)
  - (((C∨((G↔(D∧A))→A))∨I)∧((D↔J)↔J))(((C\lor ((G\leftrightarrow (D\land A))\rightarrow A))\lor I)\land ((D\leftrightarrow J)\leftrightarrow J))

- the symbols of PL can be divided into the following categories:
  - **logical symbols** (the connective symbols)
  - **nonlogical symbols** (sentence letters)
  - **auxiliary symbols** (parentheses)
- the nonlogical symbols do not have a fixed meaning and need to be specified in a glossary

- this is why you must always provide a glossary for your translations

---

- given this syntax for PL we can show how any well-formed formula of PL is constructed
- suppose we want to construct (¬P∧(Q∨R))(\lnot P \land (Q \lor R))
- we might construct it as follows using a **construction table**

---

step	wff constructed at this step	from steps/by clause
1	PP	/(2i)
2	QQ	/(2i)
3	RR	/(2i)

---

step	wff constructed at this step	from steps/by clause
1	PP	/(2i)
2	QQ	/(2i)
3	RR	/(2i)
4	¬P\lnot P	1 /(2ii ¬\lnot)
5	(Q∨R)(Q \lor R)	2,3 /(2ii ∨\lor)

---

step	wff constructed at this step	from steps/by clause
1	PP	/(2i)
2	QQ	/(2i)
3	RR	/(2i)
4	¬P\lnot P	1 /(2ii ¬\lnot)
5	(Q∨R)(Q \lor R)	2,3 /(2ii ∨\lor)
6	(¬P∧(Q∨R))(\lnot P \land (Q \lor R))	4, 5 /(2ii ∧\land)

- the other wffs in the construction are **subformulas** of the formula
- the connective added last in the construction is the **main connective**
- the main connectives gives its name to the formula it constructs

wrapping up

this lecture

- an introduction to the formal language PL
- the vocabulary and syntax of PL
- how formulas of PL are constructed
- translating propositions from English into PL with the use of a glossary

next lectures

- lecture 4, issues in translation: assertability and implicature
- lecture 5, the semantics of PL