

COMP210: Artificial Intelligence

Lecture 19, Propositional logic for knowledge representation. Inference systems.

Trevor Bench-Capon

http://www.csg.c.uiv.ac.uk/~tbc/COMO_210/

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 179

Recap & Overview

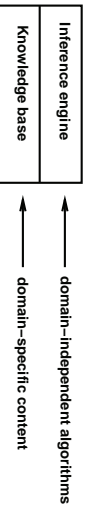
- Last time: the syntax and semantics of propositional logic.
- Today: Propositional logic for KR.

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 229

Knowledge Representation

- We are interested in a computer-suitable language to represent explicit knowledge
- reason*



Knowledge base = set of sentences in a formal language

- Clear syntax and semantics
- Adequate (in many aspects)
- Natural

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 229

Example (1)

$\text{alarm_beeps} \wedge \text{hot}$
 $\wedge (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire})$
 $\wedge (\text{alarm_beeps} \Rightarrow \text{smoky})$
 $\wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers})$

$\left. \vphantom{\begin{matrix} \text{alarm_beeps} \wedge \text{hot} \\ \wedge (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire}) \\ \wedge (\text{alarm_beeps} \Rightarrow \text{smoky}) \\ \wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers}) \end{matrix}} \right\} \text{switch_on_sprinklers}$

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 429

Example (1)

$\text{alarm_beeps} \wedge \text{hot}$
 $\wedge (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire})$
 $\wedge (\text{alarm_beeps} \Rightarrow \text{smoky})$
 $\wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers})$

$\left. \vphantom{\begin{matrix} \text{alarm_beeps} \wedge \text{hot} \\ \wedge (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire}) \\ \wedge (\text{alarm_beeps} \Rightarrow \text{smoky}) \\ \wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers}) \end{matrix}} \right\} \text{switch_on_sprinklers}$

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 429

Example (2)

$(\text{hot} \wedge \text{smoky} \Rightarrow \text{fire})$
 $\wedge (\text{alarm_beeps} \Rightarrow \text{smoky})$
 $\wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers})$

$\left. \vphantom{\begin{matrix} (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire}) \\ \wedge (\text{alarm_beeps} \Rightarrow \text{smoky}) \\ \wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers}) \end{matrix}} \right\} \text{switch_on_sprinklers}$

$\text{alarm_beeps} \wedge \text{hot} \Rightarrow \text{switch_on_sprinklers}$

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 529

Example (3)

$(\text{hot} \wedge \text{smoky} \Rightarrow \text{fire})$
 $\wedge (\text{alarm_beeps} \Rightarrow \text{smoky})$
 $\wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers})$

$\left. \vphantom{\begin{matrix} (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire}) \\ \wedge (\text{alarm_beeps} \Rightarrow \text{smoky}) \\ \wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers}) \end{matrix}} \right\} \text{switch_on_sprinklers} \Rightarrow \neg \text{fire}$

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 629

Example (4)

$(\text{hot} \wedge \text{smoky} \Rightarrow \text{fire})$
 $\wedge (\text{alarm_beeps} \Rightarrow \text{smoky})$
 $\wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers})$

$\left. \vphantom{\begin{matrix} (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire}) \\ \wedge (\text{alarm_beeps} \Rightarrow \text{smoky}) \\ \wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers}) \end{matrix}} \right\} \text{switch_on_sprinklers} \Rightarrow \neg \text{smoky}$

Trevor Bench-Capon

COMP210: Artificial Intelligence, Lecture 19: Propositional Logic for Knowledge Representation, Inference Systems, -- p. 729

Propositional Logic for KR

- Describe what we know about a particular domain by a propositional formula, KB .
- Formulate a hypothesis, α .
- We want to know whether KB implies α .

Thor Borch-Olesen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Example

$$\left. \begin{array}{l} (\text{hot} \wedge \text{smoky} \Rightarrow \text{fire}) \\ \wedge (\text{alarm_beeps} \Rightarrow \text{smoky}) \\ \wedge (\text{fire} \Rightarrow \text{switch_on_sprinklers}) \end{array} \right\} \models \neg \text{switch_on_sprinklers} \Rightarrow \neg \text{fire}$$

Thor Borch-Olesen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Entailment

- Entailment means that one thing follows from another:

$$KB \models \alpha$$

- Knowledge base KB entails sentence α if and only if α is true in *all worlds* where KB is true
 - E.g., the KB containing "the Giants won" and "the Reds won" entails "Either the Giants won or the Reds won"
 - E.g., $x + y = 4$ entails $4 = x + y$
- Entailment is a relationship between sentences (i.e., syntax) that is based on semantics

Thor Borch-Olesen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Thor Borch-Olesen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Entailment Test

How do we know that $KB \models \alpha$?

- Models
- Inference

Thor Borch-Olesen

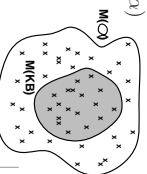
COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Thor Borch-Olesen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Models

- Logicians typically think in terms of *models*, which are formally structured worlds with respect to which truth can be evaluated
- We say m is a model of a sentence α if α is true in m
- Each line on a truth table which evaluates to true is a model for the formula
- $M(\alpha)$ is the set of all models of α
- Then $KB \models \alpha$ if and only if $M(KB) \subseteq M(\alpha)$
- E.g. $KB = \text{Giants won and Reds won}$ $\alpha = \text{Giants won}$



Thor Borch-Olesen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Example

$$\left. \begin{array}{l} (\text{Hot} \wedge \text{Smoky} \Rightarrow \text{Fire}) \\ \wedge (\text{Alarm_beeps} \Rightarrow \text{Smoky}) \\ \wedge (\text{Fire} \Rightarrow \text{sWitch_on_sprinklers}) \end{array} \right\} \models \neg \text{sWitch_on_sprinklers} \Rightarrow \neg \text{Fire}$$

Example

Example

$$\left. \begin{array}{l} (\text{Hot} \wedge \text{Smoky} \Rightarrow \text{Fire}) \\ \wedge (\text{Alarm_beeps} \Rightarrow \text{Smoky}) \\ \wedge (\text{Fire} \Rightarrow \text{sWitch_on_sprinklers}) \end{array} \right\} \models \neg \text{sWitch_on_sprinklers} \Rightarrow \neg \text{Fire}$$

Abbreviations:

$$((H \wedge S \Rightarrow F) \wedge (A \Rightarrow S) \wedge (F \Rightarrow W)) \models (\neg W \Rightarrow \neg F)$$

... gives a truth value for all possible interpretations.

H	S	F	A	W	$((H \wedge S \Rightarrow F) \wedge (A \Rightarrow S) \wedge (F \Rightarrow W))$	$\neg W \Rightarrow \neg F$
T	T	T	T	T	T	T
T	T	T	T	F	F	F
T	T	T	F	T	T	T
T	T	T	F	F	F	F
T	T	F	T	T	T	T
T	T	F	T	F	F	F
T	T	F	F	T	T	T
T	T	F	F	F	F	F
T	F	T	T	T	T	T
T	F	T	T	F	F	F
T	F	T	F	T	T	T
T	F	T	F	F	F	F
T	F	F	T	T	T	T
T	F	F	T	F	F	F
T	F	F	F	T	T	T
T	F	F	F	F	F	F
F	T	T	T	T	T	T
F	T	T	T	F	F	F
F	T	T	F	T	T	T
F	T	T	F	F	F	F
F	T	F	T	T	T	T
F	T	F	T	F	F	F
F	T	F	F	T	T	T
F	T	F	F	F	F	F
F	F	T	T	T	T	T
F	F	T	T	F	F	F
F	F	T	F	T	T	T
F	F	T	F	F	F	F
F	F	F	T	T	T	T
F	F	F	T	F	F	F
F	F	F	F	T	T	T
F	F	F	F	F	F	F

Thor Borch-Olesen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 0_1029

Truth Table

... gives a truth value for all possible interpretations.

H	S	F	A	W	$((H \wedge S \Rightarrow F) \wedge (A \Rightarrow S) \wedge (F \Rightarrow W))$	$\neg W \Rightarrow \neg F$
T	T	T	T	T	T	T
T	T	T	T	F	F	F
T	T	T	F	T	T	T
T	T	F	F	T	F	F
T	T	F	T	T	F	F
T	F	T	T	T	F	F
T	F	F	T	T	F	F
T	F	F	F	T	F	F
...

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 1029

Modus Ponens

This well known proof rule is called *modus ponens*, i.e. in general

$$\frac{A \Rightarrow B, A}{B}$$

where A and B are any WFF.

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 1029

Inference

• $KB \vdash_i \alpha$ = sentence α can be derived from KB by procedure i

• *Soundness*: i is sound if

whenever $KB \vdash_i \alpha$, it is also true that $KB \models \alpha$

• *Completeness*: i is complete if

whenever $KB \models \alpha$, it is also true that $KB \vdash_i \alpha$

That is, the procedure will answer any question whose answer follows from what is known by the KB .

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 1029

• Another common proof rule, known as \wedge -elimination is

$$\frac{A \wedge B}{A} \quad \text{or} \quad \frac{A \wedge B}{B}$$

The first of these can be read *if A and B hold (or are provable or true) then A must also hold*.

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 1029

Inference Example

fire \Rightarrow switch_on_sprinklers
 switch_on_sprinklers

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 1029

Natural Deduction Example

From $r \wedge s$ and $s \Rightarrow p$ can we prove p , i.e. show $r \wedge s, s \Rightarrow p \vdash p$?

- $r \wedge s$ [Given]
 - $s \Rightarrow p$ [Given]
 - s [1 \wedge elimination]
 - p [2,3 modus ponens]
- $$\frac{r \wedge s}{s} \quad \frac{s \Rightarrow p, s}{p}$$

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 1029

Proof Rules

• Stating that B follows (or is provable) from A_1, \dots, A_n can be written

$$\frac{A_1, \dots, A_n}{B}$$

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 1029

\vee -introduction

Another proof rule, known as \vee -introduction is

$$\frac{A}{A \vee B} \quad \text{or} \quad \frac{A}{B \vee A}$$

The first of these can be read *if A holds (or are provable or true) then $A \vee B$ must also hold*.

Thore Bach/Colson

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10, 2029

Proof Theory

- Reasoning about statements of the logic without considering interpretations is known as *proof theory*.
- Proof rules** (or inference rules) show us, given true statements how to generate further true statements.
- Axioms describe 'universal truths' of the logic.
 - Example $\vdash p \vee \neg p$ is an axiom of propositional logic.
- We use the symbol \vdash denoting *is provable* or *is true*.
- We write $A_1, \dots, A_n \vdash B$ to show that B is provable from A_1, \dots, A_n (given some set of inference rules).

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

Soundness and Completeness

- Let A_1, \dots, A_n, B be well formed formulae and let $A_1, \dots, A_n \vdash B$
 - denote that B is derivable from A_1, \dots, A_n .
 - Informally, soundness involves ensuring our proof system gives the *correct* answers.
 - Theorem(Soundness)** If $A_1, \dots, A_n \vdash B$ then $A_1 \wedge \dots \wedge A_n \models B$
 - Informally, completeness involves ensuring that *all* formulae that should be able to be proved can be.
 - Theorem(Completeness)** If $A_1 \wedge \dots \wedge A_n \models B$ then $A_1, \dots, A_n \vdash B$.

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

Proofs

- Let A_1, \dots, A_m, B be well-formed formulae
- There is a proof of B from A_1, \dots, A_m if there exists some sequence of formulae

$$C_1, \dots, C_n$$

such that $C_n = B$, and each formula C_k , for $1 \leq k < n$ is either an axiom or one of the formulae A_1, \dots, A_m , or else is the conclusion of a rule whose premises appeared earlier in the sequence.

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

More about Soundness & Completeness

- Example: An unsound (bad) inference rule is
$$\frac{A, B}{C}$$
- Using this rule from any p and q we could derive r yet $p \wedge q \models r$ does not hold.

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

Example

From $p \Rightarrow q, (\neg r \vee q) \Rightarrow (s \vee p)$, q can we prove $s \vee q$?

- $p \Rightarrow q$ [Given]
- $(\neg r \vee q) \Rightarrow (s \vee p)$ [Given]
- q [Given]
- $s \vee q$ [3, \vee introduction]

- Think how much work we would have had to do to construct a truth table to show $((p \Rightarrow q) \wedge ((\neg r \vee q) \Rightarrow (s \vee p))) \wedge q \models (s \vee q)$!

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

Is Natural Deduction Complete?

- The set of rules modus ponens and \wedge elimination is incomplete: Without \vee -introduction we cannot do the proof on page 23 yet

$$((p \Rightarrow q) \wedge ((\neg r \vee q) \Rightarrow (s \vee p))) \wedge q \models (s \vee q).$$

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

Exercise

Show r from $p \Rightarrow (q \Rightarrow r)$ and $p \wedge q$ using the rules we have been given so far. That is prove

$$p \Rightarrow (q \Rightarrow r), p \wedge q \vdash r.$$

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

Comments

- We haven't shown a full set of proof rules but just some examples.
- For a full set of proof rules look for *Natural Deduction* in a logic or AI book.
- More than 10 rules
- Intricate proofs (indirect proofs, reductio ad absurdum, etc)
- Note, at any step in the proof there may be many rules which could be applied. May need to apply search techniques, heuristics or strategies to find a proof.
- Getting computers to perform proof is an area of AI itself known as *automated reasoning*.

Theor. Borch/Orgen

COMP210: Artificial Intelligence, Lecture 10: Propositional Logic for Knowledge Representation, Reference system -- 10_2109

Summary

- We've discussed proof or inference rules and axioms.
- We've given examples of the proof rules \wedge -introduction, modus ponens and \vee -elimination.
- We've given some example proofs.
- After the Easter break we will look at how we can automate deduction