Computational Logic — Lecture 8

Relational Proofs

Michael Genesereth

Autumn 2007

Logical Entailment

A set of premises logically entails a conclusion if and only if every interpretation that satisfies the premises also satisfies the conclusion.

2

Propositional Interpretations

	p	*q*	*r*
	0	0	0
	0	0	1
→	0	1	0
	0	0	1
	1	0	0
	1	0	1
	1	1	0
	1	1	1

For a language with n constants, there are 2^n interpretations.

3

Relational Interpretations

$\|i\|$	*a*	*b*	*r*
{○,●}	○	○	{}
{○,●}	○	○	{○}
{○,●}	○	○	{●}
{○,●}	○	○	{○, ●}
{○,●}	○	●	{}
{○,●}	○	●	{○}
{○,●}	○	●	{●}
{○,●}	○	●	{○, ●}
{○,●}	●	○	{}
{○,●}	●	○	{○}
{○,●}	●	○	{●}
{○,●}	●	○	{○, ●}

. . .

Infinitely many interpretations.

4

Logical Entailment and Provability

Good News: If Δ logically entails φ, then there is a finite proof of φ from Δ. And vice versa.

More Good News: If Δ logically entails φ, it is possible to find such a proof in finite time.

Sad News: If Δ does not logically entail φ, the process of finding a proof may run forever.

Not So Bad News: In many cases, the process can be stopped after finitely many steps.

5

Formal Proofs

A *formal proof* of φ from Δ is a sequence of sentences terminating in φ in which each item is either:

1. a premise (a member of Δ)
2. an instance of an axiom schema
3. the result of applying a rule of inference to earlier items in the sequence.

6

Old Rules of Inference

Modus Ponens (MP)

$$\frac{\varphi \Rightarrow \psi \quad \varphi}{\psi}$$

Modus Tolens (MT)

$$\frac{\varphi \Rightarrow \psi \quad \neg \psi}{\neg \varphi}$$

And Introduction (AI)

$$\frac{\varphi \quad \psi}{\varphi \wedge \psi}$$

And Elimination (AE)

$$\frac{\varphi \wedge \psi}{\varphi \quad \psi}$$

7

Universal Generalization

Rule of Inference

$$\frac{\varphi}{\forall \nu . \varphi}$$

Examples:

$$\frac{p(x)}{\forall x . p(x)}$$

$$\frac{p(x) \Rightarrow q(x)}{\forall x . (p(x) \Rightarrow q(x))}$$

8

Existential Generalization

Rule of Inference

$$\frac{\varphi[\tau]}{\exists \nu.\varphi[\nu]}$$

Examples:

$$\frac{p(a)}{\exists x.p(x)} \qquad \frac{p(a) \vee q(a)}{\exists x.(p(x) \vee q(x))}$$

9

Idea for Universal Instantiation

$$\frac{\forall \nu.\varphi}{\varphi[\nu \leftarrow \tau]}$$

Warning: This is not quite right.

10

Examples

$\forall y.hates(jane,y)$

$hates(jane,jill)$	$y \leftarrow jill$
$hates(jane,mother(jane))$	$y \leftarrow mother(jane)$
$hates(jane,y)$	$y \leftarrow y$
$hates(jane,z)$	$y \leftarrow z$

$\forall x.\exists y.hates(x,y)$

$\exists y.hates(jane,y)$	$x \leftarrow jane$	
$\exists y.hates(y,y)$	$x \leftarrow y$	*Wrong!!*

11

Bounding

A term τ is *bound* by ν in φ if and only if τ contains a variable μ *and* there is some free occurrence of ν in φ *and* that occurrence lies in the scope of a quantifier of μ.

mother*(*x*)** is bound by ***y in **∃*x.hates*(*x,y*)**.

Why?

The term *mother(x)* contains a variable *x*.
There is a free occurrence of *y* in $\exists x.hates(x,y)$.
That occurrence of *y* lies in scope of quantifier of *x*.

12

Substitutability

A term τ is *substitutable* for ν in φ if and only if it is *not* bound by ν in φ.

Some texts say "x is *free* for y in φ" instead of "x is substitutable for y in φ".

mother(*jane*) is free for y in *hates*(*jane*,y).
mother(x) is free for y in *hates*(*jane*,y).
mother(x) is free for y in $\exists z.hates(z,y)$.
mother(x) is *not* free for y in $\exists x.hates(x,y)$.
mother(x) is free for y in $(\forall x.\forall y.l(x,y) \wedge \exists z.h(z,y))$.

13

Inappropriateness

An occurrence of a term τ is *inappropriate* for a variable ν in φ if and only if τ contains a variable μ *and* there is some free occurrence of ν in φ that lies in the scope of a quantifier of μ.

mother(x) is inappropriate for y in $\exists x.hates(x,y)$.

14

Substitutability

A term τ is *substitutable* for ν in φ if and only if it is not inappropriate with ν in φ.

Some texts say "x is *free* for y in φ" instead of "x is substitutable for y in φ".

mother(*jane*) is free for y in *hates*(*jane*,y).
mother(x) is free for y in *hates*(*jane*,y).
mother(x) is free for y in $\exists z$.*hates*(z,y).
mother(x) is *not* free for y in $\exists x$.*hates*(x,y).
mother(x) is free for y in $(\forall x.\forall y.l(x,y) \wedge \exists z.h(z,y))$.

15

Universal Instantiation

$$\frac{\forall \nu.\varphi}{\varphi[\nu \leftarrow \tau]}$$

where τ is free for ν in φ

16

Existential Instantiation I

$$\frac{\exists \nu.\varphi}{\varphi[\nu \leftarrow \sigma]}$$

where $\exists \nu.\varphi$ contains no free variables

where σ is a new object constant

17

Examples

$\exists y.p(y)$

$p(c)$

$\exists y.y*y{=}0$

1*1=0 *Wrong!*

$\exists y.y*y{=}x$

$c*c{=}x$ *Wrong!*

$c*c{=}4$

$c*c{=}6$

18

Existential Instantiation II

$$\frac{\exists \nu.\varphi}{\varphi[\nu \leftarrow \pi(\tau_1,...,\tau_n)]}$$

where $\tau_1,...,\tau_n$ are free in $\exists \nu.\varphi$

where π is a new function constant

19

Examples

$\exists y.y*y=x$

$f(x)*f(x)=x$

$f(4)*f(4)=4$

$f(6)*f(6)=6$

$\exists y.y*y=x$

$sqrt(x)*sqrt(x)=x$

$log(x)*log(x)=x$ *Wrong!*

20

Formal Proofs

A *formal proof* of φ from Δ is a sequence of sentences terminating in φ in which each item is either:

1. a premise (a member of Δ)
2. an instance of an axiom schema
3. the result of applying a rule of inference to earlier items in the sequence.

21

Example

Everybody loves somebody. Everybody loves a lover. Show that Jack loves Jill.

1.	$\forall x.\exists y.loves(x, y)$	Premise
2.	$\forall u.\forall v.\forall w.(loves(v, w) \Rightarrow loves(u, v))$	Premise
3.	$\exists y.loves(jill, y)$	UI : 1
4.	$loves(jill, mike)$	EI : 3
5.	$\forall v.\forall w.(loves(v, w) \Rightarrow loves(jack, v))$	UI : 2
6.	$\forall w.(loves(jill, w) \Rightarrow loves(jack, jill))$	UI : 5
7.	$loves(jill, mike) \Rightarrow loves(jack, jill)$	UI : 6
8.	$loves(jack, jill)$	MP : 7, 4

22

Harry and Ralph

Every horse can outrun every dog. Some greyhounds can outrun every rabbit. Harry is a horse. Ralph is a rabbit. Can Harry outrun Ralph?

1.	$\forall x.\forall y.(h(x) \wedge d(y) \Rightarrow f(x,y))$	Premise
2.	$\exists y.(g(y) \wedge \forall z.(r(z) \Rightarrow f(y,z)))$	Premise
3.	$\forall y.(g(y) \Rightarrow d(y))$	Premise
4.	$\forall x.\forall y.\forall z.(f(x,y) \wedge f(y,z) \Rightarrow f(x,z))$	Premise
5.	$h(harry)$	Premise
6.	$r(ralph)$	Premise

Harry and Ralph (continued)

7.	$g(greg) \wedge \forall z.(r(z) \Rightarrow f(greg,z))$	$\mathrm{EI}:2$
8.	$g(greg)$	$\mathrm{AE}:7$
9.	$\forall z.(r(z) \Rightarrow f(greg,z))$	$\mathrm{AE}:7$
10.	$r(ralph) \Rightarrow f(greg,ralph)$	$\mathrm{UI}:9$
11.	$f(greg,ralph)$	$\mathrm{MP}:6,10$
12.	$g(greg) \Rightarrow d(greg)$	$\mathrm{UI}:3$
13.	$d(greg)$	$\mathrm{MP}:12,8$
14.	$\forall y.(h(harry) \wedge d(y) \Rightarrow f(harry,y))$	$\mathrm{UI}:1$
15.	$h(harry) \wedge d(greg) \Rightarrow f(harry,greg)$	$\mathrm{UI}:14$
16.	$h(harry) \wedge d(greg)$	$\mathrm{AI}:5,13$
17.	$f(harry,greg)$	$\mathrm{MP}:15,16$

Harry and Ralph (continued)

18.	$\forall y.\forall z.(f(harry,y) \wedge f(y,z) \Rightarrow f(harry,z))$	UI : 4
19.	$\forall z.(f(harry,greg) \wedge f(greg,z) \Rightarrow f(harry,z))$	UI : 18
20.	$f(harry,greg) \wedge f(greg,ralph) \Rightarrow f(harry,ralph))$	UI : 19
21.	$f(harry,greg) \wedge f(greg,ralph)$	AI : 17,11
22.	$f(harry,ralph)$	MP : 20, 21

25

Mendelson Logic

Mendelson Logic is that subset of Relational Logic in which there are only two operators, viz. $\neg$ and $\Rightarrow$, and one quantifier, viz. $\forall$. Fortunately, all sentences in Relational Logic can be reduced to logically equivalent sentences with these operators by applying the following rules.

$$(\psi \Leftrightarrow \varphi) \rightarrow ((\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi))$$
$$(\varphi \Leftarrow \psi) \rightarrow (\psi \Rightarrow \varphi)$$
$$(\psi \wedge \varphi) \rightarrow \neg(\neg\varphi \Rightarrow \psi)$$
$$(\psi \vee \varphi) \rightarrow (\neg\varphi \Rightarrow \psi)$$
$$\exists \nu.\varphi \rightarrow \neg\forall \nu.\neg\varphi$$

26

Mendelson Rules of Inference

Modus Ponens (MP)

$$\frac{\varphi \Rightarrow \psi \qquad \varphi}{\psi}$$

Universal Generalization (UG)

$$\frac{\varphi}{\forall \nu . \varphi}$$

27

Mendelson Axiom Schemata

II: $\varphi \Rightarrow (\psi \Rightarrow \varphi)$

ID: $(\varphi \Rightarrow (\psi \Rightarrow \chi)) \Rightarrow ((\varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow \chi))$

CR: $(\neg\psi \Rightarrow \varphi) \Rightarrow ((\neg\psi \Rightarrow \neg\varphi) \Rightarrow \psi)$
$(\psi \Rightarrow \varphi) \Rightarrow ((\psi \Rightarrow \neg\varphi) \Rightarrow \neg\psi)$

UD: $\forall\nu.(\varphi \Rightarrow \psi) \Rightarrow (\forall\nu.\varphi \Rightarrow \forall\nu.\psi)$

UI: $\forall\nu.\varphi \Rightarrow \varphi[\nu \leftarrow \tau]$
where τ is free for ν in φ

28

Provability

A sentence φ is *provable* from a set of sentences Δ if and only if there is a finite formal proof of φ from Δ using only Modus Ponens, Universal Generalization, and the Mendelson axiom schemata.

Soundness Theorem: If φ is provable from Δ, then Δ logically entails φ.

Completeness Theorem (Godel): If Δ logically entails φ, then φ is provable from Δ.

29

Decidability

A *class* of questions is *decidable* if and only if there is a procedure such that, when given as input any question in the class, the procedure halts and says *yes* if the answer is positive and *no* if the answer is negative.

Example: For any natural number *n*, determining whether *n* is prime.

30

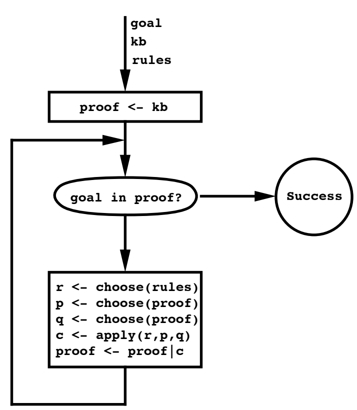
Semidecidability

A *class* of questions is semidecidable if and only if there is a procedure that halts and says *yes* if the answer is positive.

Obvious Fact: If a class of questions is decidable, it is semidecidable.

31

Semidecidability of Logical Entailment



32

Decidability Not Proved

Note that we have *not* shown that logical entailment for Relational Logic is decidable.

The procedure may not halt.

$p(x) \Rightarrow p(f(x))$
$p(f(f(a)))$
$p(f(b))$?

We cannot just run procedure on negated sentence because that may not be logically entailed either!

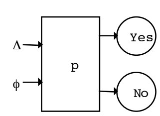
$p(x) \Rightarrow p(f(x))$
$p(f(f(a)))$
$\neg p(f(b))$?

33

Undecidability of Logical Entailment

Metatheorem: Logical Entailment for Relational Logic is *not* decidable.

Proof: Suppose there is a machine *p* that decides the question of logical entailment. Its inputs are Δ and φ.



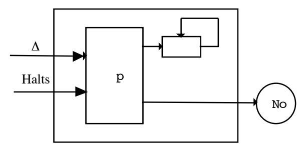
We can encode the behavior of this machine and its inputs as sentences and ask whether the machine halts as a conclusion.

What happens if we give this description and question to *p*? It says *yes*.

34

Undecidability (continued)

It is possible to construct a larger machine *p'* that enters an infinite loop if *p* says *yes* and halts if *p* says *no*.



We can also encode a description of this machine as a set of sentences and ask whether the machine halts as a conclusion.

What happens if we give this description and question to *p'*? If *p* says *yes*, then *p'* runs forever, contradicting the hypothesis that *p* computes correctly. If *p* says *no*, then *p'* halts, once again leading to contradiction. QED

35

Closure

The *closure* S^* of a set S of sentences is the set of all sentences logically entailed by S.

$$S^* = \{\varphi \mid S \models \varphi\}$$

Set of Sentences:

$p(a)$
$p(x) \Rightarrow p(f(x))$

Closure:

$p(a)$
$p(f(a))$
$p(f(f(a)))$
$p(a) \wedge p(f(a))$
$p(x) \Rightarrow p(f(x))$
…

36

Theories

A *theory* is a set of sentences closed under logical entailment, i.e. T is a theory if and only if $T^*=T$.

A theory T is *finitely axiomatizable* if and only if there is a *finite* set Δ of sentences such that $T=\Delta^*$.

A theory T is *complete* if and only, for all φ, either $\varphi \in T$ or $\neg\varphi \in T$.

Note: Not every theory is complete. Consider the theory consisting of all consequences of $p(a,b)$. Does this include $p(b,a)$? Does it include $\neg p(b,a)$?

37

Relationships on Theories

Decidable

Semidecidable   Finitely Axiomatizable

38

Arithmetization of Logical Entailment

The theory of arithmetic is the set of all sentences true of the natural numbers, 0, 1, +, *, and <.

Fact: It is possible to assign numbers to sentences such that
(1) Every sentence φ is assigned a unique number n_φ.
(2) The question of logical entailment $\Delta|=\varphi$ can be expressed as a numerical condition $r(n_\Delta,n_\varphi)$.

Conclusion: The theory of arithmetic is not decidable.

39

Incompleteness Theorem

Metatheorem (Godel): If Δ is a finite subset of the theory of arithmetic, then Δ^* is not complete.

Variant: Arithmetic is not finitely axiomatizable.

Proof: If there were a finite axiomatization, then the theory would be decidable. However, arithmetic is not decidable. Therefore, there is no finite axiomatization.

40

Summary

Logical Entailment for Relational Logic is semidecidable.

Logical Entailment for Relational Logic is *not* decidable.

Arithmetic is not finitely axiomatizable in Relational Logic.

41