More on Logical Implication

- $p \rightarrow q$ is *defined* to be ((not(p)) or q)
- But how does this connect with intuitive meaning of implication?
- A truth value implies itself, so $t \rightarrow t$ and $f \rightarrow f$ both evaluate to t
- *Anything* implies the truth, so $f \rightarrow t$ evaluates to t
- However, something true cannot imply something false, so $t \rightarrow f$ evaluates to f

Clicker Question

- Is (not(p)) $\rightarrow$ p
  1. Tautology
  2. Contingency
  3. Contradiction ?

Sequent Calculus

In this lecture we move from truth tables to a more sophisticated (and sometimes more natural) form of proof.

We explain by example the relevance of this to database manipulation and checking English grammar.

Along the way we encounter the issue of completeness of a proof system.

Problem: Database Join

employee

name	ID
dave	2345
ken	9324
mary	6782
phil	7934

lecturer

name	sex	age
dave	male	42
mary	female	21
phil	male	64

lecturer name = employee name
age < 50

new

name	sex	age	ID
dave	male	42	2345
mary	female	21	6782

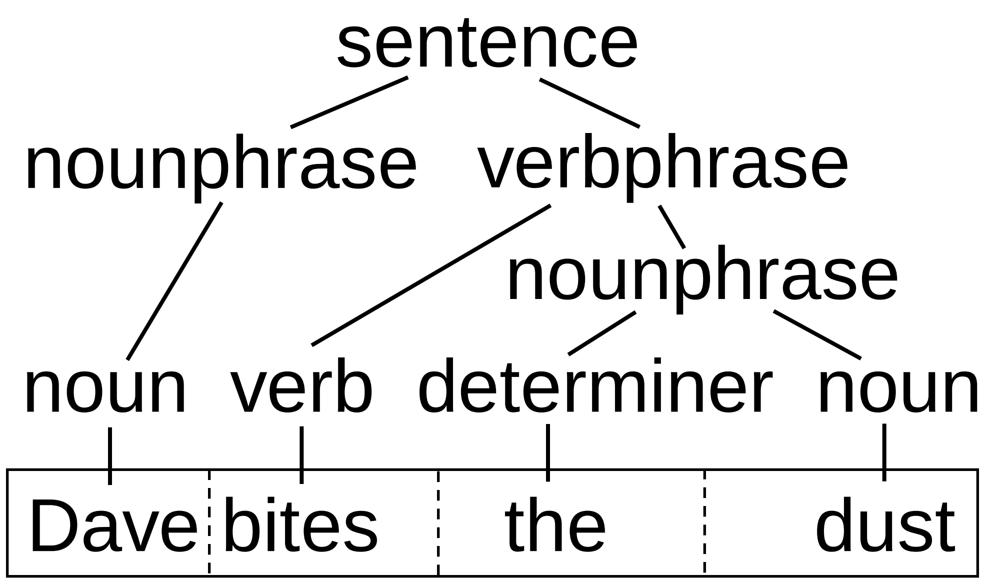
Problem: English Grammar

Grammar

sentence $\Rightarrow$ nounphrase, verbphrase
nounphrase $\Rightarrow$ noun | determiner, noun
verbphrase $\Rightarrow$ verb | verb, nounphrase

Parse

Sequence



A Little More Notation

$F \vdash A$ means that A can be proved from F

Where F is a sequence of expressions [E1,E2,…]

$[A|F]$ adds A to F to make a larger sequence

$A \in F$ is true if A appears in F

A Set of Proof Rules

Sequent	Already derived sequent(s)		
$F \vdash A$	None ($A \in F$)		
$F \vdash A$ and B	$F \vdash A$	$F \vdash B$	
$F \vdash A$ or B	$F \vdash A$		
$F \vdash A$ or B	$F \vdash B$		
$F \vdash C$	(A or $B \in F$)	$[A\|F] \vdash C$	$[B\|F] \vdash C$
$F \vdash B$	($A \rightarrow B \in F$)	$F \vdash A$	
$F \vdash A \rightarrow B$	$[A\|F] \vdash B$		

Difference from Truth Tables

$$[p(a), p(X) \rightarrow p(f(X))] \vdash p(f(f(a)))$$

isn't readily proved via a truth table but a proof is:

$$[p(a), p(X) \rightarrow p(f(X))] \vdash p(f(f(a)))$$

$$p(f(a)) \rightarrow p(f(f(a))) \in [p(a), p(X) \rightarrow p(f(X))]$$

$$[p(a), p(X) \rightarrow p(f(X))] \vdash p(f(a))$$

$$p(a) \rightarrow p(f(a)) \in [p(a), p(X) \rightarrow p(f(X))]$$

$$[p(a), p(X) \rightarrow p(f(X))] \vdash p(a)$$

$$p(a) \in [p(a), p(X) \rightarrow p(f(X))]$$

Better Than Truth Tables?

We can deal with proofs that truth tables can't handle but the *particular* proof rules we gave are not able to prove all the things we could prove from the truth tables:

- We can't prove not(p and not(p))
- We can't prove (a and b) $\rightarrow$ a or b

We can deal with these limitations by adding more proof rules and axioms.

In General

For any system of logic we can ask the following questions:

- Is it sound (it can give no incorrect answers)?
- Is it complete (it can give all correct answers)?
- Is it decidable (it actually will find all answers)?

A Larger Proof Tree

- $[B \rightarrow C] \vdash (A \text{ or } B) \rightarrow (A \text{ or } C)$
  - $[(A \text{ or } B), B \rightarrow C] \vdash (A \text{ or } C)$
    - $(A \text{ or } B) \in [(A \text{ or } B), B \rightarrow C]$
    - $[A, (A \text{ or } B), B \rightarrow C] \vdash (A \text{ or } C)$
      - $[A, (A \text{ or } B), B \rightarrow C] \vdash A$
        - $A \in [A, (A \text{ or } B), B \rightarrow C]$
    - $[B, (A \text{ or } B), B \rightarrow C] \vdash (A \text{ or } C)$
      - $[B, (A \text{ or } B), B \rightarrow C] \vdash C$
        - $B \rightarrow C \in [B, (A \text{ or } B), B \rightarrow C]$
        - $[B, (A \text{ or } B), B \rightarrow C] \vdash B$
          - $B \in [B, (A \text{ or } B), B \rightarrow C]$

A Proof Strategy

Sequent	Pre-derived			
$F \vdash A$	$A \in F$			Try this first
$F \vdash A$ and B	$F \vdash A$	$F \vdash B$		Then this for an “and”
$F \vdash A$ or B	$F \vdash A$			Then these for an “or”
$F \vdash A$ or B	$F \vdash B$			
$F \vdash C$	A or $B \in F$	$[A \mid F] \vdash C$	$[B \mid F] \vdash C$	
$F \vdash B$	$A \to B \in F$	$F \vdash A$		Otherwise one of these
$F \vdash A \to B$	$[A \mid F] \vdash B$			Then this for an “$\to$”

Solution: Database Join (1)

employee	
name	ID

lecturer		
name	sex	age

employee(dave, 2345)
employee(ken, 9324)
employee(mary, 6782)
employee(phil, 7934)

lecturer(dave, male, 42)
lecturer(mary, female, 21)
lecturer(phil, male, 64)

lecturer(N,S,A) → (name(N) and sex(S) and age(A))

employee(N,I) → (name(N) and id(I))

Solution: Database Join (2)

lecturer(N1,S,A) and
employee(N2,I) and
N1 = N2 and
A < 50

$\rightarrow$ new(N1,S,A,I)

employee(dave, 2345)
employee(ken, 9324)
employee(mary, 6782)
employee(phil, 7934)

lecturer(dave, male, 42)
lecturer(mary, female, 21)
lecturer(phil, male, 64)

new(dave, male, 42, 2345)
new(mary, female, 21, 6782)

new

name	sex	age	ID
dave	male	42	2345
mary	female	21	6782

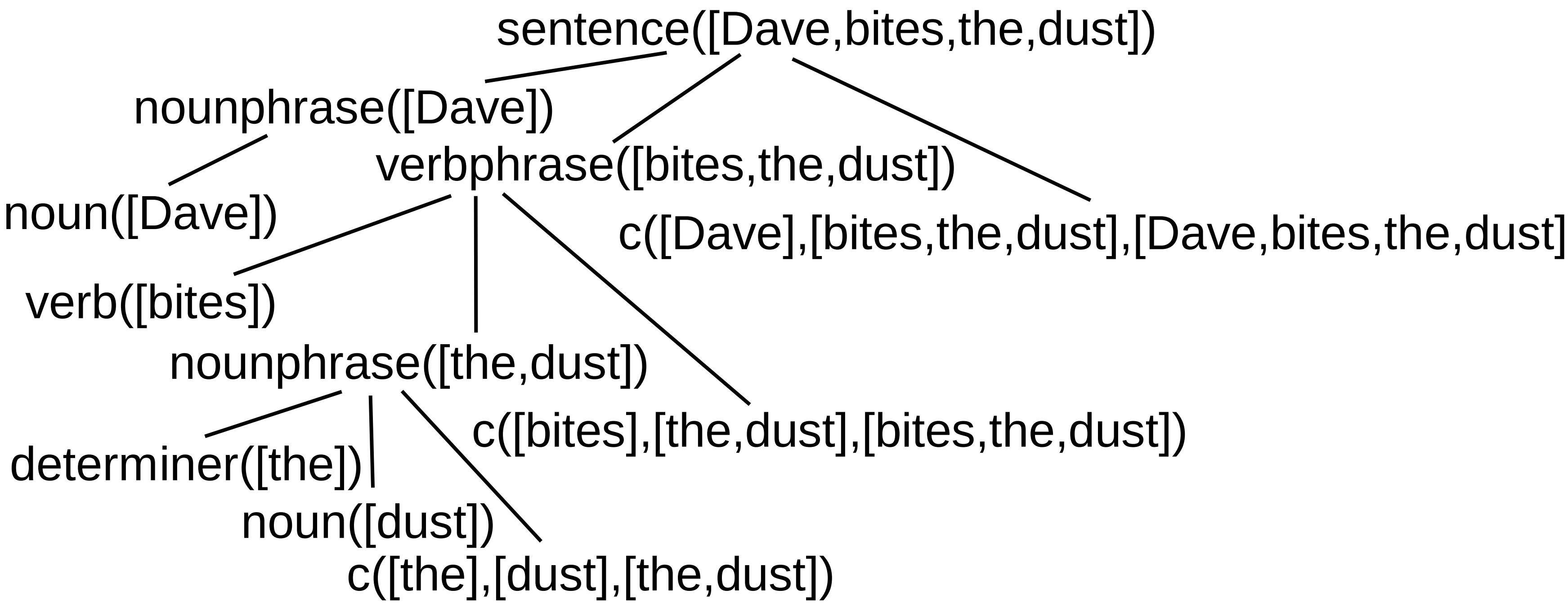
Solution: English Grammar (1)

sentence = concatenation of nounphrase with verbphrase
nounphrase = noun or concatenation of determiner with noun
verbphrase = verb or concatenation of verb with nounphrase
noun = [Dave] or [dust]
verb = [bites]
determiner = [the]

To logic

nounphrase(S1) and verbphrase(S2) and c(S1,S2,S) → sentence(S)
noun(S) or (determiner(S1) and noun(S2) and c(S1,S2,S)) → nounphrase(S)
verb(S) or (verb(S1) and nounphrase(S2) and c(S1,S2,S)) → verbphrase(S)
noun([Dave]) noun([dust])
verb([bites])
determiner([the])

Solution: English Grammar (2)



nounphrase(S1) and verbphrase(S2) and c(S1,S2,S) → sentence(S)
noun(S) or (determiner(S1) and noun(S2) and c(S1,S2,S)) → nounphrase(S)
verb(S) or (verb(S1) and nounphrase(S2) and c(S1,S2,S)) → verbphrase(S)

Things to Practice

- Try some sequent proofs of simple propositional expressions. You will get some examples in your tutorial - try more.
- Try some proofs where the expressions you use as axioms contain variables (like those in the database and grammar examples) so you get used to matching variables and constants during a proof.