Propositional Logic: Syntax and Natural Deduction [1]

The Plot That Will Unfold

- I want to provide some key historical and intellectual context to the 'model theoretic' approach to natural language semantics, context that is usually missing from introductory semantics classes / texts.

- I want to begin by getting across the very motivation for having 'models' at all:

    - o Why did logicians develop this mathematical device?
    - o What is it good for? What did it originally do for us?

- Consequently, I'm going to start off by giving a *purely syntactic* introduction to the systems of Propositional Logic and First Order (Predicate) Logic.

    - o We'll then see that certain key questions about these systems will require us to develop a mathematically precise characterization of what it is to be an 'interpretation' of these formal languages…

1. Some Historical Background

(1) **Valid Argument**
An argument is *valid* if whenever the premises are true, the conclusion *must* be true.

a. Valid Argument: All men are mortal. Socrates is a man.
Therefore, Socrates is mortal.

b. Invalid Argument: All cats are mortal. All dogs are mortal.
Therefore, all cats are dogs.

(2) **Key Insight from Ancient Philosophers (Aristotle)**
Many valid arguments seem to share the same general *syntactic* form.

a. Valid Arguments:

All Ps are Q. S is a P.
Therefore, S is a Q.
(Barbara)

No Ps are Qs. All Ss are Ps.
Therefore, No Ss are Qs.
(Celarent)

b. Aristotelian Logic (Syllogistic Logic):
First to characterize valid arguments purely in terms of their *syntactic form*.
- o So successful that it *was* logic from 300BC to 1800s.

---

[1] These notes are based upon material in the following required reading: Gamut (1991), Chapter 1, Chapter 2 pp. 28-41, Chapter 4 pp. 128-141; Partee *et al*. (1993) Chapter 5, Chapter 6 pp. 97-98.

(3) **Major Weakness of Syllogistic Logic**

- Only applies to 'syllogisms', a very restricted form of argument.
- Consequently, cannot capture valid arguments based on *relations*:

  All horses are animals.
  Therefore, every horse's head is an animal's head.

(4) **Key Developments in the 19th Century**

a. Early-to-Mid 1800's

- Renewed activity in logic; first highly original work in centuries (George Boole, Augustus DeMorgan)
- Renewed interest in whether mathematics derives from 'pure logic'.

b. Frege's *Begriffsschrift* (1879)

- Birth of modern formal logic.
- First appearance of quantified variables
- Essentially all of First Order Logic (with a little Second Order Logic, too)
- Highly idiosyncratic 2-dimensional notation[2]
- Reconstructs an unprecedented amount of mathematics and natural language reasoning
  - First real indication that math could just derive from pure logic

c. Further Developments After *Begriffsshrift*

- Other logicians develop and modify the key insights of *Begriffsschrift*
- Peano introduces many of our modern logical notations: $\forall$, $\exists$, $\supset$
- **Principia Mathematica (1910-1913; but begun much earlier)**
  - Could be called the 'climax' of the 19th century period
  - Three-volume work reconstructing enormous amounts of mathematics from a logical proof system close to what we have today.

[2] For a nice overview of Frege's funky logical notation: http://en.wikipedia.org/wiki/Begriffsschrift

(5) **The Core Idea Behind a Formal Logic**

- A precisely defined formal notation for representing *certain aspects* (not all) of the 'logical structure' of an assertion.

- A set of *syntactically defined* rules for deriving formulas in the notation from other formulas in the notation.

Illustration: Natural Deduction (Logic 101):

1.	p → (~q ∨ r)	Premise
2.	q	Premise
3.	q → ~r	Premise
4.	~r	Modus Ponens 2,3
5.	q & ~r	Conjunction 2,4
6.	~~(q & ~r)	Double Negation 5
7.	~(~q ∨ ~~r)	DeMorgans 6
8.	~(~q ∨ r)	Double Negation 7
9.	~p	Contraposition 1,8

(6) **An Immediate Problem for This Program (Foreshadowing)**

- As with classical logic, the goal of the enterprise is to provide a purely *syntactic* characterization of what it is to be a *valid argument*.

- But, the notion of 'validity' is a *semantic* one.
(if premises *true*, then conclusion must be *true*)

- **So, how can you be sure that your syntactically defined system does what it's supposed to do?**
    - How do you show that it *only* derives valid inferences?
    - How do you show that it derives *all* the valid inferences?

- **To do this, you need a mathematically precise characterization of what it means for a formula to be 'true' (relative to an 'interpretation').**

    - It wasn't until about the 1930's or so until this idea reached its modern form (Löwenheim, Gödel, Tarski)

Let's begin by putting ourselves in the position of a person in 1910 or so…
We'll start off with a <u>purely syntactic</u> introduction to Propositional Logic

2. A Review of Propositional Logic (PL): Syntax and Informal Semantics [3]

The system of Propositional Logic (PL) is intended to capture the inferences that depend upon the meaning of the so-called 'sentential connectives': *and*, *or*, *if…then*, and *not*

(7) **The Vocabulary of Symbols**

a. The Non-Logical Constants (a.k.a 'The Logical Variables')
An infinite set of *proposition letters*: {p, q, r, s, t, … p_1, p_2, p_3, p_4, …}

b. The Logical Constants:

(i)	~	Negation	'It is not the case that…'
(ii)	&	Conjunction	'and'
(iii)	v	Disjunction	'or' (inclusive)
(iv)	→	(Material) Implication	'if…then'

c. Syntactic Symbols: (,)

(8) **The Definition of a 'Well-Formed Formula' (WFF) of PL**
The set of 'well-formed formulae' of PL, WFF, is the smallest set such that:

a. If φ is a proposition letter, then $\varphi \in$ WFF
b. If $\varphi, \psi \in$ WFF, then
1. $\sim\varphi \in$ WFF
2. $(\varphi \,\&\, \psi) \in$ WFF
3. $(\varphi \vee \psi) \in$ WFF
4. $(\varphi \rightarrow \psi) \in$ WFF

(9) **Using PL To Encode Sentences of English**

- We can use the syntactic rules in (8) and the informal semantics in (7) to write PL formulae that 'encode' certain statements of English:

a. *Sentence:* "Dave is tall, and if Dave isn't tall, then Mary is dancing."
Encoding: (p & (~p → q))

b. *Sentence:* "If Bill or John is leaving, then Mary and Sue are not happy."
Encoding: ((b v j) → (~m & ~s))

- In setting up such encodings, it is critical to supply a 'key', indicating which English assertions the propositional letters 'stand for':

c. *Key:*

p: Dave is tall	b: Bill is leaving	m: Mary is happy
q: Mary is dancing	j: John is leaving	s: Sue is happy

[3] My discussion here will assume prior familiarity with the overall system of Propositional Logic. Students are referred to Partee *et al.* (1993), Chapter 6 for crucial background.

(10) **An Important Property of the PL Syntax**

- In (8), we provide a *recursive definition* of the well-formed formulae of PL.
- This definition allows us to use *mathematical induction* to prove things about *all* well-formed formulae of PL.

(11) **Illustration: An Inductive Proof about PL Formulae**

a. Claim: Every well-formed formulae of PL has an even number of parentheses. (*i.e.*, If $\varphi \in$ WFF, then there are an even number of parentheses in φ)

b. Proof By Strong Induction
We'll prove the claim by mathematical induction *on the number of logical constants in* φ

(i) Base Step: 0
If φ contains no logical constants, then φ is a proposition letter, and so φ contains 0 parentheses (and 0 is even).

(ii) Induction Step
Let n be such that for all m < n, if φ contains m logical constants, then φ has an even number of parentheses.

- Now suppose that φ contains n logical constants. There are four possible cases to consider:

1. φ is of the form $\sim\psi$, where ψ contains (n-1) logical constants. By assumption, then, ψ contains an even number of parentheses. Thus, so does φ.

2. φ is of the form $(\chi \ \&\ \psi)$, where χ contains m < n logical constants, and ψ contains j < n logical constants. By assumption, then, χ and ψ both contain an even number of parentheses. Thus, so does φ.

3. φ is of the form $(\chi \vee \psi)$, where χ contains m < n logical constants, and ψ contains j < n logical constants. By argument parallel to (2), so does φ.

4. φ is of the form $(\chi \rightarrow \psi)$, where χ contains m < n logical constants, and ψ contains j < n logical constants. By argument parallel to (2), so does φ.

(12) **Proof by Induction on the Complexity of Formula**

- The general structure of the proof in (11) is *extremely* common in (meta)logic. It's often called *proof by induction on the complexity of formula.*
- As the example in (11) makes clear, such 'proofs by induction on complexity of formula' are really just a special case of 'proof by (mathematical) induction'

<u>Induction on the Complexity of Formula:</u>
Suppose that you've shown both (i) and (ii) below. It follows that every formula φ in the language has property P.

(i) Every primitive formula of the language has property P.

(ii) For every complex formula φ, if the immediate subformulae of φ have property P, then so does φ

---

3. A Review of Propositional Logic (PL): Natural Deduction

(13) **Major Goal of This Section**
Let's provide a *purely syntactic* characterization of 'valid inference' in the PL notation.

- This syntactic characterization will be embodied in a *proof system* (natural deduction)
- We're going to lay out some rules – stated entirely in *syntactic terms* – for deriving formulae in PL from other formulae.
  - o As we'll see, these syntactic rules intuitively capture certain key aspects of the everyday meaning of the English logical words 'not', 'and', 'or', 'if..then'

(14) **Definition of a Derivation**
A *derivation* is a finite, numbered list of formulae, where each formula is accompanied by a coded statement indicating how it entered the derivation (see (5))

<u>Such 'coded statements' will be of the following two forms:</u>

(i) 'Assumption' indicates that the formula is an assumption (premise)

(ii) 'RULE n, …, m' Where 'RULE' is the name of one of the derivation rules below, and 'n, …, m' are the numbers of the formulae in the derivation that are 'input' to the rule

(15) **Some Terminology and Notation**

a. Assumptions of the Derivation:
The formulae in the derivation that are accompanied by 'Assumption'

b. Conclusion of the Derivation: The final line in the derivation.

c. Key notation: $\{ \varphi_1 , \varphi_2 , \varphi_3 \ldots \} \vdash \psi$
'There is a derivation where *a subset of* $\{ \varphi_1 , \varphi_2 , \varphi_3 \ldots \}$ are the assumptions, and ψ is the conclusion'

Note: Given the way '$\vdash$' is defined in (15c), the set $\{ \varphi_1 , \varphi_2 , \varphi_3 \ldots \}$ could be *infinite.*
This will make certain proofs about the system easier later on…

Now let's get to stating and illustrating some of the deduction rules!...

(16) **The Rule of '&-Introduction' (I&)**
The following is an acceptable derivation:

1.	…	
…	…	
n_1	φ	
…	…	
n_2	ψ	
…	…	
m	$(\varphi \,\&\, \psi)$	I& n_1, n_2

Illustration: { p, q, r } ⊢ ((p & q) & (p & r))

1.	p	Assumption
2.	q	Assumption
3.	r	Assumption
4.	(p & q)	I& 1,2
5.	(p & r)	I& 1,3
6.	((p & q) & (p & r))	I& 4,5

Intuitive Motivation:
If we can (in English) assert some sentence 'S_1', and some sentence 'S_2', then we can also assert the sentence 'S_1 and S_2'

Note:
From now on, I'll leave 'the following is an acceptable derivation' implicit in the statement of the deduction rules…

(17) **The Rule of '&-Elimination' (E&)**

1.	…		1.	…	
…	…		…	…	
n	(φ & ψ)		n	(φ & ψ)	
…	…		…	…	
m	φ	E& n	m	ψ	E& n

Intuitive Motivation:
If we can (in English) assert some sentence 'S_1 and S_2', then we can also assert 'S_1'.
If we can (in English) assert some sentence 'S_1 and S_2', then we can also assert 'S_2'.

Illustration: Proving the Associativity of '&' (Part 1)

1.	(p & (q & r))	Assumption
2.	p	E& 1
3.	(q & r)	E& 1
4.	q	E& 3
5.	(p & q)	I& 2,4
6.	r	E& 3
7.	((p & q) & r)	I& 5,6

Note: By a similar derivation, we can show ((p & q) & r) ⊢ (p & (q & r))

(18) **The Rule of 'Repetition'**
The following rule doesn't add anything interesting, but it makes proofs easier.

1.	…	
…	…	
n.	φ	
…	…	
m.	φ	Repetition n

Illustration:

1.	(p & (q & r))	Assumption
2.	p	E& 1
3.	(q & r)	E& 1
4.	q	E& 3
5.	r	E& 3
6.	(p & q)	I& 2,4
7.	r	Repetition 5
8.	((p & q) & r)	I& 6,7

(19) **The Rule of '$\rightarrow$-Elimination' (E$\rightarrow$)**
The following derivation rule is often referred to as 'modus ponens'

1.	…	
…	…	
n_1	$(\varphi \rightarrow \psi)$	
…	…	
n_2	φ	
…	…	
m	ψ	E$\rightarrow$ n_1, n_2

<u>Intuitive Motivation:</u>
If we can (in English) assert the sentence 'If S_1 then S_2', and we can assert the sentence 'S_1', then we can also assert 'S_2'

<u>Illustration:</u> $\{ (p \rightarrow (q \rightarrow r)), p, q \} \vdash r$

1.	$(p \rightarrow (q \rightarrow r))$	Assumption
2.	p	Assumption
3.	q	Assumption
4.	$(q \rightarrow r)$	E$\rightarrow$ 1,2
5.	q	Repetition 3
6.	r	E $\rightarrow$ 4,5

(20) **The Rule of '$\rightarrow$-Introduction' (I$\rightarrow$) and 'Conditional Proof'**

<u>Intuitive Motivation:</u>
If whenever we assume 'S_1' it follows that 'S_2', then we can assert 'if S_1 then S_2'.

<u>The Rule:</u>
The following is an acceptable derivation, as long as no line j > m makes reference to any lines from n_1 to n_2.

1.	…	
…	…	
n_1	| φ	Assumption
…	|	
n_2	| ψ	
m	$(\varphi \rightarrow \psi)$	I$\rightarrow$

- In such a derivation, we say that the formulae occurring on lines n_1 to n_2 have been *dropped* or *withdrawn*

- Note the key restriction that once a formula has been *dropped*, no subsequent lines can make reference to it.

Illustration: $((p \& q) \rightarrow r) \vdash (p \rightarrow (q \rightarrow r))$

1.	$((p \& q) \rightarrow r)$		Assumption
2.	│ p		Assumption
3.	│	│ q	Assumption
4.	│	│ $(p \& q)$	I& 2,3
5.	│	│ r	E→ 1, 4
6.	│ $(q \rightarrow r)$		I→
7.	$(p \rightarrow (q \rightarrow r))$		I→

(21) **The Rule of '∨-Introduction' (I∨)**

1.	…		1.	…	
…	…		…	…	
n.	φ		n.	ψ	
…	…		…	…	
m.	$(\varphi \vee \psi)$	I∨ n	m.	$(\varphi \vee \psi)$	I∨ n

Intuitive Motivation:
If we can (in English) assert a sentence 'S_1', then for any sentence 'S_2', we can assert 'S_1 or S_2' and 'S_2 or S_1' (inclusive 'or')

(22) **The Rule of '∨-Elimination' (E∨)**

1.	…	
…	…	
n_1	$(\varphi \vee \psi)$	
…	…	
n_2	$(\varphi \rightarrow \chi)$	
…	…	
n_3	$(\psi \rightarrow \chi)$	
…	…	
m	χ	E∨ n_1, n_2, n_3

Intuitive Motivation:
If we can (in English) assert 'S_1 or S_2', 'if S_1 then S_3' and also 'if S_2 then S_3', then we can also assert 'S_3'

(23) **Illustration of the Rules for Disjunction: Associativity of '∨' (Part 1)**

$(p \vee (q \vee r)) \vdash ((p \vee q) \vee r)$

1.	$(p \vee (q \vee r))$	Assumption
2.	│ p	Assumption
3.	│ $(p \vee q)$	I∨ 2
4.	│ $((p \vee q) \vee r)$	I∨ 3
5.	$(p \rightarrow ((p \vee q) \vee r))$	I→
6.	│ $(q \vee r)$	Assumption
7.	│ │ q	Assumption
8.	│ │ $(p \vee q)$	I∨ 7
9.	│ │ $((p \vee q) \vee r)$	I∨ 8
10.	│ $(q \rightarrow ((p \vee q) \vee r))$	I →
11.	│ │ r	Assumption
12.	│ │ $((p \vee q) \vee r)$	I∨ 11
13.	│ $(r \rightarrow ((p \vee q) \vee r))$	I →
14.	│ $((p \vee q) \vee r)$	E∨ 6, 10, 13
15.	$((q \vee r) \rightarrow ((p \vee q) \vee r))$	I →
16.	$((p \vee q) \vee r)$	E∨ 1, 5, 15

(24) **The Rule of '~ -Elimination' (E~)**

For various reasons, it will help to have a special symbol indicating that a contradiction has been reached: ⊥ 'falsum'

1.	…	
…	…	
n_1	φ	
…	…	
n_2	$\sim\varphi$	
…	…	
m	$\bot$	E~ n_1 n_2

(25) **The Rule of '~ -Introduction' (I~)**

With this special symbol, we can now state the following rule of I~

1.	…	
…	…	
n_1	│ φ	Assumption
…	│	
n_2	│ $\bot$	
m	$\sim\varphi$	I~

<u>Intuitive Motivation:</u>

If whenever we assume 'S' a contrdiction follows, then we can assert 'not S'.

(26) **Illustration of the Rules for Negation: Double Negation (Part 1)**

p ⊢ ~~p

1.	p		Assumption
2.		~p	Assumption
3.		⊥	E~ 1,2
4.	~~p		I~

(27) **The Rule of 'Ex Falso Sequitur Quodlibet' (EFSQ)**

<u>Intuitive Motivation:</u> Anything follows from a contradiction.

1.	…	
…	…	
n	⊥	
m	φ	EFSQ

<u>Illustration:</u> { (p ∨ q) , ~p } ⊢ q

1.	(p ∨ q)		Assumption
2.	~p		Assumption
3.		p	Assumption
4.		⊥	E~ 2,3
5.		q	EFSQ
6.	(p → q)		I→
7.		q	Assumption
8.		q	Repetition 7
9.	(q → q)		I →
10.	q		E∨ 1, 6, 9

(28) **The Rule of 'Double Negation' (~~)**

1.	…	
…	…	
n	~~ φ	
…	…	
m	φ	~~ n

<u>Note:</u>
Although our system without (28) can prove one half of the equivalence 'φ *iff* ~~φ' (26), we need the rule in (28) for our system to derive the complete equivalence.

4. The Power of Our Natural Deduction System for PL

Although we our system has just 11 (really, 10) relatively simple rules, it can capture a great many intuitively valid inferences!

(29) **Derivation of DeMorgans, Part 1** ~(p & q) ⊢ (~p ∨ ~q)

1.	~(p & q)	Assumption
2.	~(~p ∨ ~q)	Assumption
3.	p	Assumption
4.	q	Assumption
5.	(p & q)	I& 3,4
6.	⊥	E~ 1, 5
7.	~q	I~
8.	(~p ∨ ~q)	I∨ 7
9.	⊥	E~ 2, 8
10	~p	I~
11.	(~p ∨ ~q)	I∨ 10
12.	⊥	E~ 2, 11
13.	~~(~p ∨ ~q)	I~
14.	(~p ∨ ~q)	~~ 13

(30) **Derivation of DeMorgans, Part 2** (~p ∨ ~q) ⊢ ~(p & q)

1.	(~p ∨ ~q)	Assumption
2.	~p	Assumption
3.	(p & q)	Assumption
4.	p	E& 3
5.	⊥	E~ 2, 4
6.	~(p & q)	I~
7.	(~p → ~(p & q))	I→
8.	~q	Assumption
9.	(p&q)	Assumption
10.	q	E& 9
11.	⊥	E~ 8,10
12.	~(p & q)	I~
13.	(~q → ~(p & q))	I→
14.	~(p & q)	E∨ 1, 7, 13

(31) **The Big Question**
Does our system offer a *perfect* syntactic characterization of 'validity' for PL?

- Does every derivation correspond to a valid inference?
- Does every valid inference in PL correspond to a derivation?

How would we even show this?...