Logic for Linguists: Lecture 1

Gregory Wilsenach

University of Cambridge

18th October 2019

What is a Logic anyway?

It is difficult to give a response that is both formal and informative.

Informally, a logic consists of a specified syntax, which logicians understand to be mere marks on a page, together with a specified semantics, usually denoted by some function that assigns to each sentence a set of ‘abstract structures’ which unambiguously define its meaning.

Examples includes positional logic, predicate logic, computer programs, etc.

A logic is often, but not always, paired with a proof system, which gives a formal method of deriving true statements from other true statements.

Logicians study a great many more things as well, including notions of sets and classes (and theories of the infinite), theories of probability, and more.

Logic vs. Natural Language

We have already run into some important differences between natural language and logic.

Logic	Natural Language
Syntax and semantics specified explicitly	Syntax and meaning determined by fieldwork
Syntax and semantics fixed	Syntax and semantics changes with time
Semantics defined recursively (compositionally)	Semantics has complex layers
Clear and unambiguous semantics	Inherent ambiguity

So why should logic be interesting to linguists?

Because logic is intrinsically interesting.

There are also more practical reasons

- The idea of associating a sentence with a family of structures, and the *model theory* developed by logicians, is often invoked when developing semantics for natural languages (see, for example, *From Discourse to Logic* by Kemp and Reyle).
- First-order logic (and its extensions) are used all over the place, for example to model discourse (see, for example, discourse representation theory).
- The Lambda calculus and other ideas from recursion theory are similarly invoked in semantics.

Some Readings

The intersection between logic and language runs deep. See

- *Mathematical Methods in Linguistics* by Partee
- *Mathemtical Linguistics* by Kornai
- *The Mathematics of Sentence Structure* by Lambek
- *Logical Semantics* by Carpenter
- *Logics of Conversation* by Asher and Lascarides
- *Logic and Language* by van Benthem and Ter Meulen

A Tentative Overview and Some Discussion

Part I

- Propositional Logic
- Predicate (First-Order) Logic
- Second-Order Logic, Generalised Quantifiers, and Extensions
- Model Logic

Part II

- Automata and Regular Languages
- Turing Machines, Recursion Theory, the Chomsky Hierarchy,
- Gödel's Incompleteness Theorems and Paradoxes

Propositional Logic

What is a Proposition?

A proposition is a statement that must be either true or false.

These are propositions

- "There is at least one person in this class."
- "The number four is not divisible by two."
- "Helen of Troy existed."
- "There is at least one person in this class and Helen of Troy existed."

These are not

- "Who is Steve?"
- "Stop that!"

But what about these?

- "Brown is a wonderful colour."
- "The present king of France is bald." (see Russell's theory of descriptions)

Propositional Logic

We will now define a logic that allows us to define sentences by starting with a set of atomic propositions and then combining them by taking conjunctions, disjunctions, implications, and negations.

As alluded to at the beginning of this talk we should begin by first defining the syntax of this logic and then its semantics.

But first some examples.

Examples

It is useful to have in mind a problem that propositional logic might help solve. Let's consider arguments formed by stringing together propositions and consider their validity.

Premises:

1. If it is raining, it is not cold
2. If it is not raining, John is not wearing a coat
3. It is cold

Conclusion: John is not wearing a coat.

Premises:

1. If it is snowing, it is cold or it is wet
2. If it is cold, John is wearing a coat
3. It is snowing

Conclusion: John is wearing a coat

The Syntax

The sentences of propositional logic are built from variables (meant to denote atomic propositions) and the connectives $\wedge$ (and), $\vee$ (or), $\neg$ (not), and $\rightarrow$ (implies).

We usually denote variables by lower case letters:

- $a, b, c, \ldots$ or $p, q, \ldots$

We define the sentences of propositional logic as follows:

- each variable is a sentence,
- if ϕ is a sentence then $\neg\phi$ is a sentence, and
- if ϕ and ψ are sentences then $\phi \vee \psi$, $\phi \wedge \psi$, and $\phi \rightarrow \psi$ are sentences.

Semantics (Valuations)

The semantics has already been alluded to, and should be quite obvious from the names we chose to give the logical connectives. We define the semantics formally by defining first the notion of a valuation.

A *valuation* or is a function v that maps each variable either to true (written $\top$) or false (written $\bot$).

We shall see that, by taking the obvious definitions of the logical connectives, a valuation uniquely defines an evaluation of a propositional sentence.

Semantics (Models)

We say that v evaluates ϕ to $\top$ (or that v is a model of ϕ or that v satisfies ϕ) and write $v \models \phi$ if

- ϕ is just a variable a and $v(a) = \top$;
- ϕ is equal to $\neg\psi$ and $v \not\models \psi$;
- ϕ is equal to $\psi \wedge \theta$ and both $v \models \psi$ and $v \models \theta$;
- ϕ is equal to $\psi \vee \theta$ and either $v \models \psi$ or $v \models \theta$ (or both); or
- ϕ is equal to $\psi \rightarrow \theta$ and $v \models \theta$ if $v \models \psi$.

An Example

Let's consider an example. Let's suppose we have a valuation defined for the variables $\{a, b, c, d\}$ such that

$$v(a) := v(b) := \top \text{ and } v(c) := v(d) := \bot$$

and we want to evaluate the sentence

$$\phi := (\neg(a \land \neg(c))) \lor (b \land \neg(d)).$$

We evaluate this sentence recursively. We have $v \models \phi$ if, and only if, $v \models \neg(a \land \neg(c))$ or $v \models b \land \neg(d)$. Let's check the second sentence first. We have $v \models b \land \neg(d)$ if, and only if, $v \models b$ and $v \models \neg d$. But $v(b) = \top$ and $v \models \neg d$ as $v \not\models d$ since $v(d) = \bot$. It follows that $v \models \phi$.

Semantics

We can then associate a sentence ϕ with the set $\mathrm{mod}(\phi)$ consisting of all those valuations v such that $v \models \phi$. This is the semantics of ϕ.

We say that a sentence ϕ is a *tautology* if every valuation is a model of ϕ. We say that ϕ is *consistent* if there exists v such that $v \models \phi$, and otherwise we say ϕ is *inconsistent*.

This puts us in a computational difficult situation. In order to determine the semantics of a formula we need to evaluate the formula for every possible valuation, and there are 2^n of those (where n is the number of variables that appear in ϕ).

There is a neat (but laborious) means for representing the semantics of a propositional sentence. These representations are called *truth tables*.

Truth Tables

The idea is very simple, we simply write out all of the evaluations.

a	c	$\neg(a \wedge \neg(c))$
$\bot$	$\bot$	$\top$
$\bot$	$\top$	$\top$
$\top$	$\bot$	$\bot$
$\top$	$\top$	$\top$

This gives us a natural (but laborious) means for checking if a given sentence is a tautology.

Exercise: Is the formula ϕ defined two slides ago a tautology?

Inference and Tautology

Perhaps the most useful application of propositional logic is that it allows us to recognise certain valid forms of reasoning.

The most famous inference rule *modus ponens* can be written as

$$(a \wedge (a \rightarrow b)) \rightarrow b.$$

I leave it as an exercise to verify that this is a tautology. We can similarly show that *modus tollens*

$$(\neg b \wedge (a \rightarrow b)) \rightarrow \neg a$$

is a tautology.We can also establish the validity of *argument by contraposition*

$$(a \rightarrow b) \leftrightarrow (\neg b \rightarrow \neg a).$$

Let's return to our earlier example

Premises:

1. If it is raining, it is not cold.
2. If it is not raining, John is not wearing a coat.
3. It is cold.

Conclusion: John is not wearing a coat. Is this a valid argument?

Let a denote "it is raining", b denote "it is cold", and c denote "John is wearing a coat". Then this argument is of the form

$$\phi := ((a \to \neg b) \wedge (\neg a \to \neg c) \wedge b) \to \neg c.$$

We can use the tautologies on the previous slide to establish that ϕ is a tautology.

Concluding Remarks

We introduced the notion of a logic and very briefly discussed the relationship between logic and linguistics.

We gave very brief introductions to numerous topics that will be covered in this course.

We discussed propositional logic, introduced a lot of notation, and explained how this logic can be used to recognise valid logical forms.