Monotonicity and Polarity in Natural Logic

Larry Moss, Indiana University

Workshop on Semantics for Textual Inference, July 10, 2011

"*Natural Logic* is a cover term for a family of formal approaches to semantics and textual inferencing as currently practiced by computational linguists.

"They have in common a proof theoretical rather than a model-theoretic focus
and an overriding concern with feasibility."

Natural Logic sometimes refers just to work on monotonicity,
but in this talk I'll be broader.

Natural logic: my take on what it's all about

Program

Re-think semantics based on computational linguistics.

Re-work the relation of logic and language, starting with inference.

First step: show that significant parts of natural language inference can be carried out in decidable logical systems.

Whenever possible, to obtain complete axiomatizations, because the resulting logical systems are likely to be interesting.

To connect the work to a host of areas in logic and theoretical CS.
But these are all the first step, and they hardly touch upon the real goals.

Differences between my projects and those of others here

Work in the RTE community features

- sentences from life
- actual running systems
- sustained work on knowledge acquisition

Differences between my projects and those of others here

Work in the RTE community features

- sentences from life
- actual running systems
- sustained work on knowledge acquisition

In contrast, what I'm doing will look like a toy.

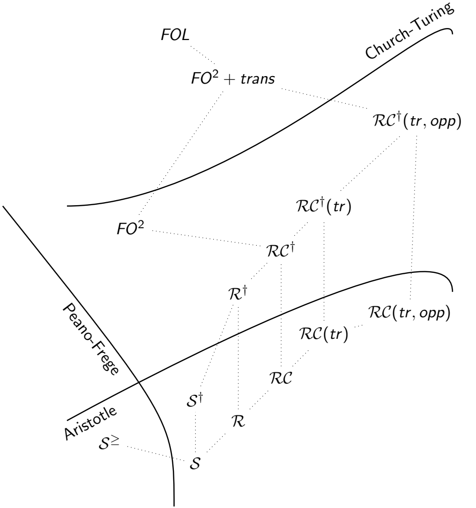
Differences between my projects and those of others here

	RTE	NL now	NL, hope
semantics	don't have/want	needed, mostly classical	needed, but flexibly so
grammar	don't have/want	needed	???
shallow vs. deep	shallow: H–T	deep	deep???
aim	$\geq$ 90% (say)	complete	complete
logic	irrelevant	centerpiece	??
algorithm	centerpiece	implicit	running

Differences between my projects and those of others here

	RTE	NL now	NL, hope
semantics	don't have/want	needed, mostly classical	needed, but flexibly so
grammar	don't have/want	needed	???
shallow vs. deep	shallow: H–T	deep	deep???
aim	$\geq$ 90% (say)	complete	complete
logic	irrelevant	centerpiece	??
algorithm	centerpiece	implicit	running
community	huge, funded	tiny and old	more than the union

Most of the fragments which have been treated



first-order logic

FO^2 + "R is trans"

2 variable FO logic

† adds full N-negation

$\mathcal{RC}(tr)$ + opposites

$\mathcal{RC}$ + (transitive) comparative adjs

$\mathcal{R}$ + relative clauses

$\mathcal{S}$ + full N-negation

$\mathcal{R}$ = relational syllogistic

$\mathcal{S}^{\geq}$ adds $|p| \geq |q|$

$\mathcal{S}$: all/some/no p are q

Syntax: Start with a collection of unary atoms (for nouns).
Sentences: *All p are q*, *Some p are q*

Semantics: A model $\mathcal{M}$ is a set M,
and for each noun p we have an interpretation $[\![p]\!] \subseteq M$.

$$\begin{array}{lll} \mathcal{M} \models \textit{All p are q} & \text{iff} & [\![p]\!] \subseteq [\![q]\!] \\ \mathcal{M} \models \textit{Some p are q} & \text{iff} & [\![p]\!] \cap [\![q]\!] \neq \emptyset \end{array}$$

Proof system is based on the following rules:

$$\frac{}{\textit{All p are p}} \qquad \frac{\textit{All p are n} \quad \textit{All n are q}}{\textit{All p are q}}$$

$$\frac{\textit{Some p are q}}{\textit{Some q are p}} \qquad \frac{\textit{Some p are q}}{\textit{Some p are p}} \qquad \frac{\textit{All q are n} \quad \textit{Some p are q}}{\textit{Some p are n}}$$

If Γ is a set of sentences, we write $\mathcal{M} \models \Gamma$ if for all $\varphi \in \Gamma$, $\mathcal{M} \models \varphi$.

$\Gamma \models \varphi$ means that every $\mathcal{M} \models \Gamma$ also has $\mathcal{M} \models \varphi$.

A proof tree over Γ is a finite tree $\mathcal{T}$
whose nodes are labeled with sentences,
and each node is either an element of Γ,
or comes from its parent(s) by an application of one of the rules.

$\Gamma \vdash \varphi$ means that there is a proof tree $\mathcal{T}$ for over Γ
whose root is labeled φ.

Example of a derivation

If there is an n, and if all n are p and also q, then some p are q.

Some n are n, *All n are p*, *All n are q* $\vdash$ *Some p are q*.

The proof tree is

$$\dfrac{\textit{All } n \textit{ are } q \qquad \dfrac{\dfrac{\textit{All } n \textit{ are } p \quad \textit{Some } n \textit{ are } n}{\textit{Some } n \textit{ are } p}}{\textit{Some } p \textit{ are } n}}{\textit{Some } p \textit{ are } q}$$

Beyond first-order logic: cardinality

Read $\exists^{\geq}(X, Y)$ as "there are at least as many Xs as Ys".

$$\frac{\textit{All } Y \textit{ are } X}{\exists^{\geq}(X, Y)} \qquad \frac{\exists^{\geq}(X, Y) \quad \exists^{\geq}(Y, Z)}{\exists^{\geq}(X, Z)}$$

$$\frac{\textit{All } Y \textit{ are } X \quad \exists^{\geq}(Y, X)}{\textit{All } X \textit{ are } Y}$$

$$\frac{\textit{Some } Y \textit{ are } Y \quad \exists^{\geq}(X, Y)}{\textit{Some } X \textit{ are } X} \qquad \frac{\textit{No } Y \textit{ are } Y}{\exists^{\geq}(X, Y)}$$

The point here is that by working with a weak basic system, we can go beyond the expressive power of first-order logic.

The languages $\mathcal{S}$ and $\mathcal{S}^\dagger$ add noun-level negation

Let us add complemented atoms $\overline{p}$ on top of
the language of All and Some,
with interpretation via set complement: $[\![\overline{p}]\!] = M \setminus [\![p]\!]$.

So we have

$$
\left.
\begin{array}{ll}
\mathcal{S} & \left\{
\begin{array}{l}
\textit{All } p \textit{ are } q \\
\textit{Some } p \textit{ are } q \\
\textit{All } p \textit{ are } \overline{q} \equiv \textit{No } p \textit{ are } q \\
\textit{Some } p \textit{ are } \overline{q} \equiv \textit{Some } p \textit{ aren't } q
\end{array}
\right. \\
& \\
& \quad \textit{Some non-}p\textit{ are non-}q
\end{array}
\right\} \quad \mathcal{S}^\dagger
$$

$$\frac{}{\textit{All } p \textit{ are } p} \qquad \frac{\textit{Some } p \textit{ are } q}{\textit{Some } p \textit{ are } p} \qquad \frac{\textit{Some } p \textit{ are } q}{\textit{Some } q \textit{ are } p}$$

$$\frac{\textit{All } p \textit{ are } n \quad \textit{All } n \textit{ are } q}{\textit{All } p \textit{ are } q} \qquad \frac{\textit{All } n \textit{ are } p \quad \textit{Some } n \textit{ are } q}{\textit{Some } p \textit{ are } q}$$

$$\frac{\textit{All } q \textit{ are } \overline{q}}{\textit{All } q \textit{ are } p}\ \textit{Zero} \qquad \frac{\textit{All } \overline{q} \textit{ are } q}{\textit{All } p \textit{ are } q}\ \textit{One}$$

$$\frac{\textit{All } p \textit{ are } \overline{q}}{\textit{All } q \textit{ are } \overline{p}}\ \textit{Antitone} \qquad \frac{\textit{Some } p \textit{ are } \overline{p}}{\varphi}\ \textit{Ex falso quodlibet}$$

The system uses

$$\frac{\textit{Some } p \textit{ are } \overline{p}}{\varphi}\ \textit{Ex falso quodlibet}$$

and this is prima facie weaker than reductio ad absurdum.

One of the logical issues in this work is to determine exactly where various principles are needed.

The next language uses "see" or r as variables for transitive verbs.

All p are q	*All p aren't q* $\equiv$ *No p are q*
Some p are q	*Some p aren't q*
All p see all q	*All p don't see all q* $\equiv$ *No p sees any q*
All p see some q	*All p don't see some q* $\equiv$ *No p sees all q*
Some p see all q	*Some p don't see any q*
Some p see some q	*Some p don't see some q*

The interpretation is the natural one, using the subject wide scope readings in the ambiguous cases.

This is $\mathcal{R}$.
The first system of its kind was Nishihara, Morita, Iwata 1990.

The language $\mathcal{R}^{\dagger}$ allows complemented atoms $\overline{p}$ as head nouns.

All p are q	$\forall(p, q)$
Some p are q	$\exists(p, q)$
All p r all q	$\forall(p, \forall(q, r))$
All p r some q	$\forall(p, \exists(q, r))$
Some p r all q	$\exists(p, \forall(q, r))$
Some p r some q	$\exists(p, \exists(q, r))$
No p are q	$\forall(p, \overline{q})$
Some p aren't q	$\exists(p, \overline{q})$
No p r any q	$\forall(p, \forall(q, \overline{r}))$
No p r all q	$\forall(p, \exists(q, \overline{r}))$
Some p don't r any q	$\exists(p, \forall(q, \overline{r}))$
Some p don't r some q	$\exists(p, \exists(q, \overline{r}))$

All p are q	$\forall(p, q)$
Some p are q	$\exists(p, q)$

All p r all q	$\forall(p, \forall(q, r))$
All p r some q	$\forall(p, \exists(q, r))$
Some p r all q	$\exists(p, \forall(q, r))$
Some p r some q	$\exists(p, \exists(q, r))$
No p are q	$\forall(p, \overline{q})$
Some p aren't q	$\exists(p, \overline{q})$
No p r any q	$\forall(p, \forall(q, \overline{r}))$
No p r all q	$\forall(p, \exists(q, \overline{r}))$
Some p don't r any q	$\exists(p, \forall(q, \overline{r}))$
Some p don't r some q	$\exists(p, \exists(q, \overline{r}))$

set terms c

positive	p	$\forall(p, r)$	$\exists(p, r)$
negative	$\overline{p}$	$\exists(p, \overline{r})$	$\forall(p, \overline{r})$

$\forall(p, r)$	those who r all p
$\exists(p, r)$	those who r some p
$\forall(p, \overline{r})$	those who fail-to-r all $p \approx$ those who r no p
$\exists(p, \overline{r})$	those who fail-to-r some $p \approx$ those who don't r some p

All p are q	$\forall(p,q)$	
Some p are q	$\exists(p,q)$	
All p r all q	$\forall(p,\forall(q,r))$	
All p r some q	$\forall(p,\exists(q,r))$	
Some p r all q	$\exists(p,\forall(q,r))$	
Some p r some q	$\exists(p,\exists(q,r))$	simplifies to
No p are q	$\forall(p,\overline{q})$	
Some p aren't q	$\exists(p,\overline{q})$	$\forall(p,c)$ $\exists(p,c)$
No p sees any q	$\forall(p,\forall(q,\overline{r}))$	
No p sees all q	$\forall(p,\exists(q,\overline{r}))$	
Some p don't r any q	$\exists(p,\forall(q,\overline{r}))$	
Some p don't r some q	$\exists(p,\exists(q,\overline{r}))$	

set terms c

positive	p	$\forall(p,r)$	$\exists(p,r)$
negative	$\overline{p}$	$\exists(p,\overline{r})$	$\forall(p,\overline{r})$

We start with one collection of unary atoms (for nouns) and another of binary atoms (for transitive verbs).

expression	variables	syntax
unary atom	$p,\ q$	
binary atom	r	
positive set term	c^+	$p \mid \exists(p,r) \mid \forall(p,r)$
set term	c, d	$p \mid \exists(p,r) \mid \forall(p,r) \mid$
		$\overline{p} \mid \exists(p,\overline{r}) \mid \forall(p,\overline{r})$
$\mathcal{R}$ sentence	φ	$\forall(p,c) \mid \exists(p,c)$
$\mathcal{R}^\dagger$ sentence	φ	$\forall(p,c) \mid \exists(p,c) \mid \forall(\overline{p},c) \mid \exists(\overline{p},c)$

We need one last concept, syntactic negation:

expression	syntax	negation
positive set term c	p	$\overline{p}$
	$\overline{p}$	p
	$\exists(p, r)$	$\forall(p, \overline{r})$
	$\forall(p, r)$	$\exists(p, \overline{r})$
	$\exists(p, \overline{r})$	$\forall(p, r)$
	$\forall(p, \overline{r})$	$\exists(p, r)$
$\mathcal{R}$ sentence φ	$\forall(p, c)$	$\exists(p, \overline{c})$
	$\exists(p, c)$	$\forall(p, \overline{c})$

Note that $\overline{\overline{p}} = p$, $\overline{\overline{c}} = c$ and $\overline{\overline{\varphi}} = \varphi$.

Theorem

There are no finite syllogistic logical systems which are sound and complete for $\mathcal{R}$.

However, there is a logical system (presented below) which uses reductio ad absurdum

$$\dfrac{\begin{matrix}[\varphi] \\ \vdots \\ \exists(p, \overline{p})\end{matrix}}{\overline{\varphi}}\ \text{RAA}$$

and which is complete.

Theorem

There are no finite syllogistic logical systems which are sound and complete for $\mathcal{R}$.

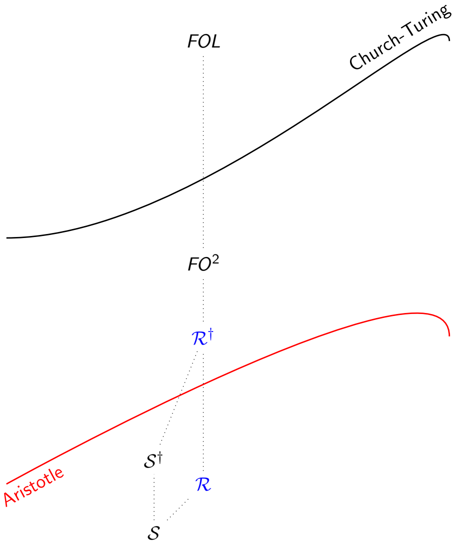
However, there is a logical system (presented below) which uses reductio ad absurdum

$$\frac{\begin{array}{c}[\varphi] \\ \vdots \\ \exists(p, \overline{p})\end{array}}{\overline{\varphi}}\ \text{RAA}$$

and which is complete.

Theorem

There are no finite, sound and complete syllogistic logical systems for $\mathcal{R}^\dagger$, even ones which allow *RAA*.



† adds full *N*-negation

relational syllogistic

p and q range over unary atoms,
c over set terms, and t over binary atoms or their negations.

$$\frac{\exists(p,q) \qquad \forall(q,c)}{\exists(p,c)} \qquad \frac{\forall(p,q) \qquad \forall(q,c)}{\forall(p,c)}$$

$$\frac{\forall(p,q) \qquad \exists(p,c)}{\exists(q,c)} \qquad \frac{}{\forall(p,p)} \qquad \frac{\exists(p,c)}{\exists(p,p)}$$

$$\frac{\forall(q,\bar{c}) \qquad \exists(p,c)}{\exists(p,\bar{q})} \qquad \frac{\forall(p,\bar{p})}{\forall(p,c)} \qquad \frac{\exists(p,\exists(q,t))}{\exists(q,q)}$$

$$\frac{\forall(p,\forall(n,t)) \qquad \exists(q,n)}{\forall(p,\exists(q,t))} \qquad \frac{\exists(p,\exists(q,t)) \qquad \forall(q,n)}{\exists(p,\exists(n,t))}$$

$$\frac{\forall(p,\exists(q,t)) \qquad \forall(q,n)}{\forall(p,\exists(n,t))} \qquad \begin{array}{c}[\varphi] \\ \vdots \\ \dfrac{\exists(p,\overline{p})}{\overline{\varphi}}\ \text{RAA}\end{array}$$

EXAMPLE OF A PROOF IN THE SYSTEM FOR $\mathcal{R}^{\dagger}$

What do you think? Sound or unsound?

$$\begin{array}{ll} & \textit{All } X \textit{ see all } Y, \textit{All } X \textit{ see some } Z, \textit{All } Z \textit{ see some } Y \\ \models & \textit{All } X \textit{ see some } Y \end{array}$$

Example of a proof in the system for $\mathcal{R}^\dagger$

What do you think? Sound or unsound?

$$\begin{array}{ll} & \textit{All } X \textit{ see all } Y, \textit{All } X \textit{ see some } Z, \textit{All } Z \textit{ see some } Y \\ \models & \textit{All } X \textit{ see some } Y \end{array}$$

The conclusion does indeed follow:
take cases as to whether or not there are X.

We should have a formal proof.

Example of a proof in this system

All X see all Y, *All X see some Z*, *All Z see some Y*

⊨ *All X see some Y*

Some X see no Y

Some X are X   *All X see some Z*

Some X see some Z

Some Z are Z   *All Z see some Y*

Some Z see some Y

Some Y are Y   *All X see all Y*

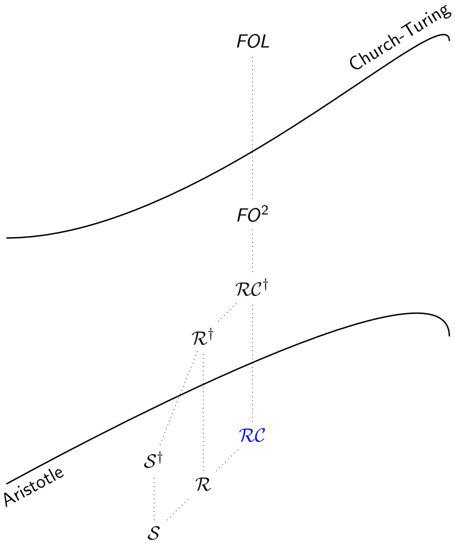
All X see some Y   *Some X see no Y*

Some X aren't X

$$
\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{[\textit{Some X see no Y}]}{\textit{Some X are X}} \quad \textit{All X see some Z}}{\textit{Some X see some Z}}}{\textit{Some Z are Z}} \quad \textit{All Z see some Y}}{\textit{Some Z see some Y}}}{\textit{Some Y are Y}} \quad \textit{All X see all Y}}{\textit{All X see some Y}} \quad [\textit{Some X see no Y}]}{\textit{Some X aren't X}}}{\textit{All X see some Y}}\ \textit{RAA}
$$

This shows that

$$\textit{All X see all Y}, \textit{All X see some Z}, \textit{All Z see some Y} \vdash \textit{All X see some Y}$$



† adds full *N*-negation

add relative clauses
= relativized quantifiers

Inference with relative clauses

What do you think about these?

All skunks are mammals
____
All who fear all who respect all skunks fear all who respect all mammals

All skunks are mammals
____
All who fear all who respect some skunks fear all who respect some mammals

All skunks are mammals
____
Some who fear all who respect some skunks fear some who respect some mammals

$\mathcal{RC}$ allows sentential subjects to be noun phrases containing subject relative clauses.

who r all p   *who r some p*

who don't r all p   *who don't r any p*

---

expression	syntax	
$\mathcal{RC}$ sentence	$\forall(d^+, c)$	$\exists(d^+, c)$
$\mathcal{RC}^\dagger$ sentence	$\forall(d, c)$	$\exists(d, c)$

d^+ is a positive set term, and c is an arbitrary set term.

The main rules are

$$\frac{\forall(p,q)}{\forall(\forall(q,r),\forall(p,r))} \quad \frac{\forall(p,q)}{\forall(\exists(p,r),\exists(q,r))} \quad \frac{\exists(p,q)}{\forall(\forall(p,r),\exists(q,r))}$$

These rules are based on McAllester and Givan (1992).

In a variant of this language which
admits iterated relative clauses, we would just have

$$\forall(s, m) \vdash \forall(\forall(\forall(s, r), f), \forall(\forall(m, r), f),$$

$$\dfrac{\dfrac{\forall(s, m)}{\forall(\forall(m, r), \forall(s, r))}}{\forall(\forall(\forall(s, r), f), \forall(\forall(m, r), f))}$$

Incorporating inexpressible background constraints

kissing involves touching

$$\frac{\textit{All skunks are mammals}}{\textit{All who fear all who touch all skunks fear all who kiss all skunks}}$$

The point is that we incorporate the constraint
into the proof theory, not as a meaning postulate.

Incorporating inexpressible background constraints

Suppose that $r \Rightarrow s$

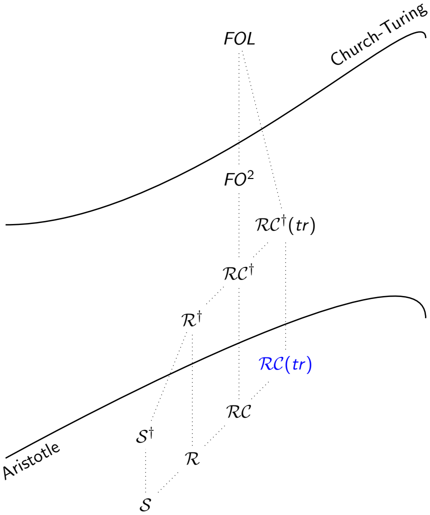
$$\frac{\forall(d, \forall(c, r))}{\forall(d, \forall(c, s))} \qquad \frac{\forall(d, \exists(c, r))}{\forall(d, \exists(c, s))} \qquad \frac{\exists(d, \forall(c, r))}{\exists(d, \forall(c, s))} \qquad \frac{\exists(d, \exists(c, r))}{\exists(d, \exists(c, s))}$$

$$\frac{}{\forall(\exists(c, r), \exists(c, s))} \qquad \frac{}{\forall(\forall(c, r), \forall(c, s))}$$

We again have completeness in the relevant sense.

Next: comparative adjectives

used for inferences involving phrases like bigger than some kitten



† adds full *N*-negation
* adds relative clauses

tr adds comparatives, requiring transitivity

Every giraffe is taller than every gnu
Some gnu is taller than every lion
<u>Some lion is taller than some zebra</u>
Every giraffe is taller than some zebra

We extend $\mathcal{RC}$ to a language $\mathcal{RC}(tr)$ by taking a set **A** of comparative adjective phrases in the base.

In the semantics, we would require of a model that for $a \in \mathbf{A}$, $[\![a]\!]$ must be a transitive relation. (We could also require irreflexivity.)

Every giraffe is taller than every gnu
Some gnu is taller than every lion
<u>Some lion is taller than some zebra</u>
Every giraffe is taller than some zebra

$$\frac{\forall(p, \exists(q, r))}{\forall(\exists(p, r), \exists(q, r))} \qquad \frac{\forall(p, \forall(q, r))}{\forall(\exists(p, r), \forall(q, r))}$$

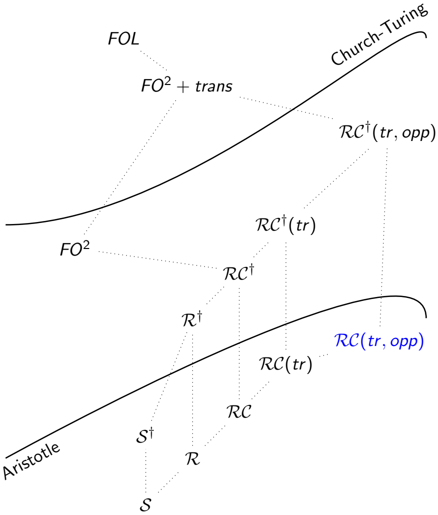
$$\frac{\exists(p, \forall(q, r))}{\forall(\forall(p, r), \forall(q, r))} \qquad \frac{\exists(p, \exists(q, r))}{\forall(\forall(p, r), \exists(q, r))}$$

Every giraffe is taller than every gnu
Some gnu is taller than every lion
Some lion is taller than some zebra
Every giraffe is taller than some zebra

$$\dfrac{\dfrac{\forall(\mathsf{gir}, \forall(\mathsf{gnu}, \mathsf{taller})) \quad \exists(\mathsf{gnu}, \forall(\mathsf{lion}, \mathsf{taller}))}{\forall(\mathsf{gir}, \forall(\mathsf{lion}, \mathsf{taller}))} \quad \exists(\mathsf{lion}, \exists(\mathsf{zebra}, \mathsf{taller}))}{\forall(\mathsf{giraffe}, \exists(\mathsf{zebra}, \mathsf{taller}))}$$

Next: relational converses

used for inferences relating bigger and smaller



† adds full *N*-negation

* adds relative clauses

opp adds opposites of comparative adjectives

$$\frac{\forall(p, \forall(q, t))}{\forall(q, \forall(p, t^{-1}))} \qquad \frac{\exists(p, \forall(q, t))}{\forall(q, \exists(p, t^{-1}))} \text{ (scope)} \qquad \frac{\forall(p, \exists(q, r^{-1}))}{\forall(\forall(q, r), \forall(p, r))}$$

$$\frac{\exists(\exists(p, r^{-1}), \exists(q, r))}{\exists(p, \exists(q, r))} \qquad \frac{\exists(\forall(p, r), \forall(q, r^{-1}))}{\forall(p, \forall(q, r^{-1}))} \qquad \frac{\exists(\forall(p, r), \exists(q, r^{-1}))}{\exists(q, \forall(p, r^{-1}))}$$

$$\frac{\forall(p, \exists(q, r)) \quad \forall(\exists(p, r^{-1}), \exists(n, r))}{\forall(p, \exists(n, r))} \ (\star) \qquad \frac{\forall(p, \exists(q, r)) \quad \forall(\exists(p, r^{-1}), \forall(n, r))}{\forall(p, \forall(n, r))}$$

(scope): if some p is bigger than all q,
then all q are smaller than some p or other.

($\star$): if every dog is bigger than some hedgehog,
and everything smaller than some dog is bigger than some cat,
then every dog is bigger than some cat.

Logic beyond the Aristotle boundary

So far in this talk, all of the systems have been syllogistic
to one degree or another.

$\mathcal{R}^\dagger$ and $\mathcal{RC}^\dagger$ lie beyond the Aristotle boundary,
due to full negation on nouns.

It is possible to formulate a logical system with
a restricted notion of variables,
prove completeness,
and yet stay inside the Church-Turing boundary.

Example of a proof in the system

From all keys are old items, infer everyone who owns a key owns an old item

$$
\dfrac{\dfrac{[\exists(key, own)(x)]^2 \qquad \dfrac{[own(x, y)]^1 \qquad \dfrac{[key(y)]^1 \quad \forall(key, old\text{–}item)}{old\text{–}item(y)}\ \forall E}{\exists(old\text{–}item, own)(x)}\ \exists I}{\exists(old\text{–}item, own)(x)}\ \exists E^1}{\forall(\exists(key, own), \exists(old\text{–}item, own))}\ \forall I^2
$$

Example of a proof in the system

From all keys are old items, infer everyone who owns a key owns an old item

1	$\forall(key, old\text{--}item)$	hyp
2	$\quad\exists(key, own)(x)$	hyp
3	$\quad key(y)$	$\exists E$, 2
4	$\quad own(x, y)$	$\exists E$, 2
5	$\quad old\text{--}item(y)$	$\forall E$, 1, 3
6	$\quad\exists(old\text{--}item, own)(x)$	$\exists I$, 4, 5
7	$\forall(\exists(key, own), \exists(old\text{--}item, own))$	$\forall I$, 1–6

Adding Transitivity to $\mathcal{RC}^{\dagger}$

We begin with the logical system for $\mathcal{RC}^{\dagger}$,
and then we add a rule:

$$\frac{a(x,y) \quad a(y,z)}{a(x,z)}\ \text{trans}$$

This rule is added for all $a \in \mathbf{A}$, and all x, y, z.

This gives a language $\mathcal{RC}^{\dagger}(tr)$.

Every sweet fruit is bigger than every kumquat

Every fruit bigger than some sweet fruit is bigger than every kumquat

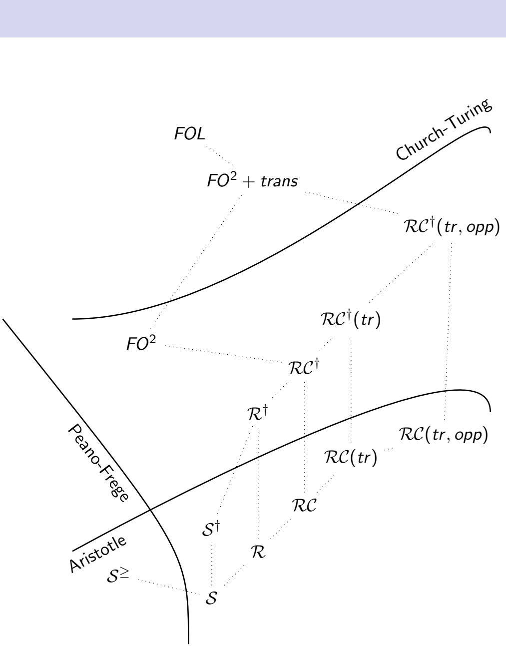
$$
\dfrac{[\exists(\mathsf{sw},\mathsf{bigger})(x)]^3 \qquad \dfrac{\dfrac{[\mathsf{bigger}(x,y)]^2 \qquad \dfrac{[\mathsf{kq}(z)]^1 \qquad \dfrac{[\mathsf{sw}(y)]^2 \quad \forall(\mathsf{sw},\forall(\mathsf{kq},\mathsf{bigger}))}{\forall(\mathsf{kq},\mathsf{bigger})(y)}\ \forall E}{\mathsf{bigger}(y,z)}\ \forall E}{\mathsf{bigger}(x,z)}\ \textsf{trans}}{\forall(\mathsf{kq},\mathsf{bigger})(x)}\ \forall I^1}{\dfrac{\forall(\mathsf{kq},\mathsf{bigger})(x)}{\forall(\exists(\mathsf{sw},\mathsf{bigger}),\forall(\mathsf{kq},\mathsf{bigger}))}\ \forall I^3}\ \exists E^2
$$

Transitivity should not be treated as a meaning postulate,
since even stating it would seem to render the logic undecidable.

Instead, it is a proof rule:

$$\frac{a(x, y) \quad a(y, z)}{a(x, z)} \text{ trans}$$

(I have not proved that one can't formulate a decidable logic which can directly express transitivity using variables and also cover the sentences we've seen.
But there are results that suggest it.)



first-order logic

FO^2 + "R is trans"

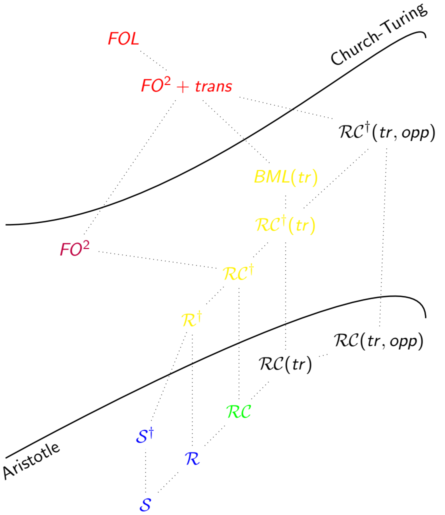
2 variable FO logic

† adds full N-negation

$\mathcal{RC}(tr)$ + opposites
$\mathcal{RC}$ + (transitive) comparative adjs
$\mathcal{R}$ + relative clauses
$\mathcal{S}$ + full N-negation
$\mathcal{R}$ = relational syllogistic
$\mathcal{S}^{\geq}$ adds $|p| \geq |q|$
$\mathcal{S}$: all/some/no p are q

Complexity

(mostly) best possible results on the validity problem



undecidable
Church 1936
Grädel, Otto, Rosen 1999

in co-NEXPTIME
EXPTIME
Lutz & Sattler 2001

Co-NEXPTIME
Grädel, Kolaitis, Vardi '97
EXPTIME
Pratt-Hartmann 2004

lower bounds also open

Co-NP
McAllester & Givan 1992

NLOGSPACE

What are the simplest forms of reasoning?

- Monotonicity in both mathematics and language
- Equational reasoning
- Syllogistic reasoning

Example of mathematical reasoning with monotone and antitone functions

Which is bigger?

$$\left(7+\frac{1}{4}\right)^{-3} \quad \text{or} \quad \left(7+\frac{1}{\pi^2}\right)^{-3}$$

Example of mathematical reasoning with monotone and antitone functions

Which is bigger?

$$\left(7 + \frac{1}{4}\right)^{-3} \quad \text{or} \quad \left(7 + \frac{1}{\pi^2}\right)^{-3}$$

$$\cfrac{\cfrac{\cfrac{\cfrac{2 \leq \pi}{\frac{1}{\pi} \leq \frac{1}{2}}\ 1/x \text{ is antitone}}{\frac{1}{\pi^2} \leq \frac{1}{4}}\ x^2 \text{ is monotone}}{7 + \frac{1}{\pi^2} \leq 7 + \frac{1}{4}}\ 7 + x \text{ is monotone}}{(7 + \frac{1}{4})^{-3} \leq (7 + \frac{1}{\pi^2})^{-3}}\ x^{-3} \text{ is antitone}$$

A first monotonicity judgment for language

every dog barks

Assume: barks loudly $\leq$ barks $\leq$ vociferates
Notice that if we replace barks by a "bigger" word,
we have an inference.
For example:

$$\frac{\textit{every dog barks}}{\textit{every dog vociferates}}$$

A first monotonicity judgment for language

every dog barks

Assume: barks loudly $\leq$ barks $\leq$ vociferates
Notice that if we replace barks by a "bigger" word,
we have an inference.
For example:

$$\frac{\textit{every dog barks}}{\textit{every dog vociferates}}$$

Notation

We'll indicate this by

every dog barks$^{\uparrow}$

What goes up, what goes down?

Assume: barks loudly $\leq$ barks $\leq$ vociferates
Assume: old dog $\leq$ dog $\leq$ animal

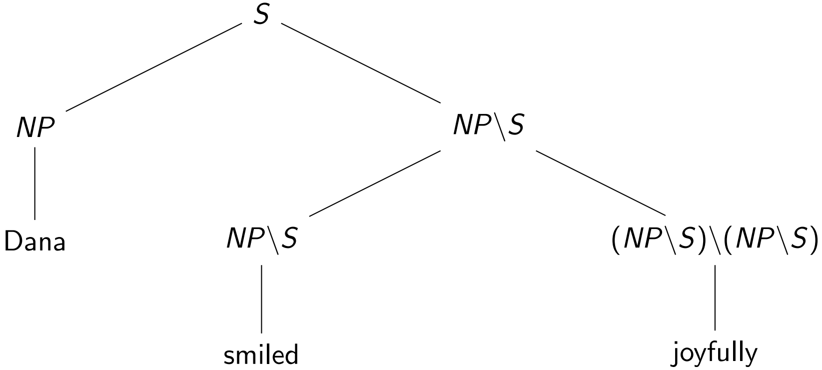
We want

every dog$^{\downarrow}$ *barks*$^{\uparrow}$
no dog$^{\downarrow}$ *barks*$^{\downarrow}$
not every dog$^{\uparrow}$ *barks*$^{\downarrow}$
some dog$^{\uparrow}$ *barks*$^{\uparrow}$
most dogs$^{\times}$ *bark*$^{\uparrow}$ no monotonicity in first argument

A categorial lexicon

(Dana, NP)
(Kim, NP)
(smiled, $NP\backslash S$)
(laughed, $NP\backslash S$)
(cried, $NP\backslash S$)
(praised, $(NP\backslash S)/NP$)
(teased, $(NP\backslash S)/NP$)
(interviewed, $(NP\backslash S)/NP$)
(joyfully, $(NP\backslash S)\backslash(NP\backslash S)$)
(carefully, $(NP\backslash S)\backslash(NP\backslash S)$)
(excitedly, $(NP\backslash S)\backslash(NP\backslash S)$)

A parse tree showing that Dana smiled joyfully is an S



It works by

- Assigning sets to the base types, here NP, S.
- Using function sets for the slash types
- Giving fixed meanings to the lexical items
- Working up the tree using function application

The previous stuff gives a model.

Overall semantic facts are defined in terms of models,
as we have already seen.

For this talk, simpler base types will do

pr for "property", *t* for "truth value".

Also, I'll ignore the directionality of the slash arrows
to make things much simpler,
and to highlight what is new here.

$$\begin{array}{lcl} \text{every} & : & (pr,(pr,t)) \\ \text{some} & : & (pr,(pr,t)) \\ \text{no} & : & (pr,(pr,t)) \\ \text{any} & : & (pr,(pr,t)) \end{array}$$

(Note that we already have a problem in giving the semantics of "any".)

A PREORDER IS A PAIR $\mathbb{P} = (P, \leq)$, WHERE $\leq$ IS REFLEXIVE AND TRANSITIVE

PREORDERS ARE NEEDED TO REALLY DISCUSS UPWARD/DOWNWARD MONOTONICITY

The proposal is to enrich the basic
semantic architecture of CG by moving from sets to preorders.

A preorder is a pair $\mathbb{P} = (P, \leq)$, where $\leq$ is reflexive and transitive

Preorders are needed to really discuss upward/downward monotonicity

The proposal is to enrich the basic
semantic architecture of CG by moving from sets to preorders.

A function $f : \mathbb{P} \to \mathbb{Q}$ is

monotone if $p \leq q$ in $\mathbb{P}$ implies $f(p) \leq f(q)$ in $\mathbb{Q}$.
antitone if $p \leq q$ in $\mathbb{P}$ implies $f(q) \leq f(p)$ in $\mathbb{Q}$.

A preorder is a pair $\mathbb{P} = (P, \leq)$, where $\leq$ is reflexive and transitive

Preorders are needed to really discuss upward/downward monotonicity

The proposal is to enrich the basic
semantic architecture of CG by moving from sets to preorders.

A function $f : \mathbb{P} \to \mathbb{Q}$ is

monotone if $p \leq q$ in $\mathbb{P}$ implies $f(p) \leq f(q)$ in $\mathbb{Q}$.
antitone if $p \leq q$ in $\mathbb{P}$ implies $f(q) \leq f(p)$ in $\mathbb{Q}$.

From now on, all functions are monotone

$-\mathbb{Q}$ is $(Q, \geq)$: it's $\mathbb{Q}$ upside-down.

$-(-\mathbb{Q}) = \mathbb{Q}$.

An antitone $f : \mathbb{P} \to \mathbb{Q}$ is exactly a montone $f : \mathbb{P} \to -\mathbb{Q}$.

LET'S THINK ABOUT MONOTONICITY IN CONNECTION WITH TRUTH TABLES

T means "true" and F means "false".

$\neg P$: *not P*
$P \wedge Q$: *P and Q*.
$P \vee Q$: *P or Q*.
$P \rightarrow Q$: *P implies Q*; or *If P, then Q*.

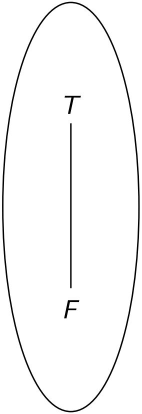
P	$\neg P$
T	F
F	T

P	Q	$P \wedge Q$
T	T	T
T	F	F
F	T	F
F	F	F

P	Q	$P \vee Q$
T	T	T
T	F	T
F	T	T
F	F	F

P	Q	$P \rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

The main preorder here is the tiny preorder I'll call $\mathbb{2}$.



$\mathbb{2}$

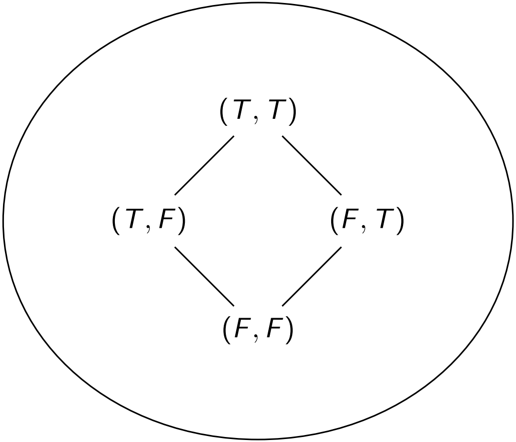
Notice that $F < T$.

But what are the preorders?

But for $\wedge$, $\vee$, and $\rightarrow$, we need to think about pairs of truth values, so we need a preorder with four elements.

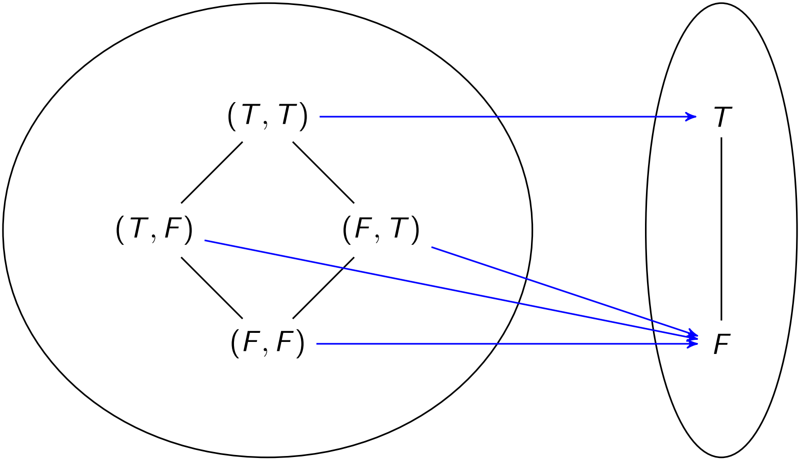
Which should we use?

But what are the preorders?



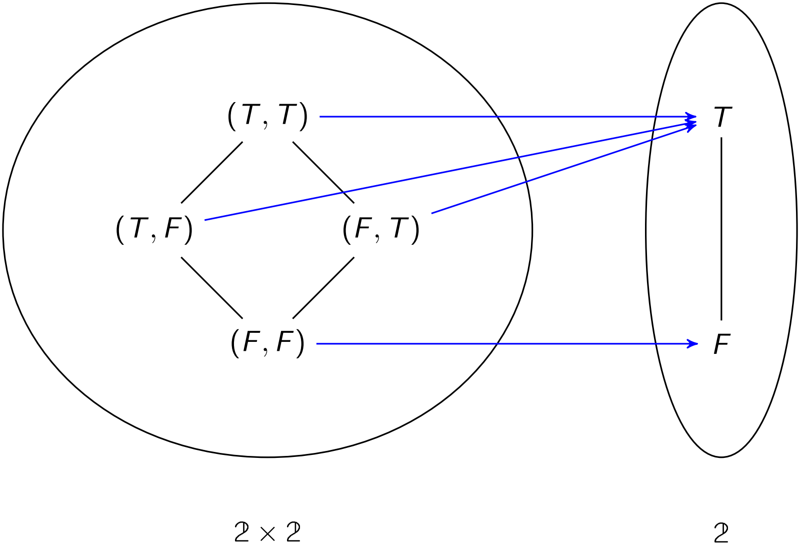
$\mathbb{2} \times \mathbb{2}$

Conjunction ∧ as a monotone function



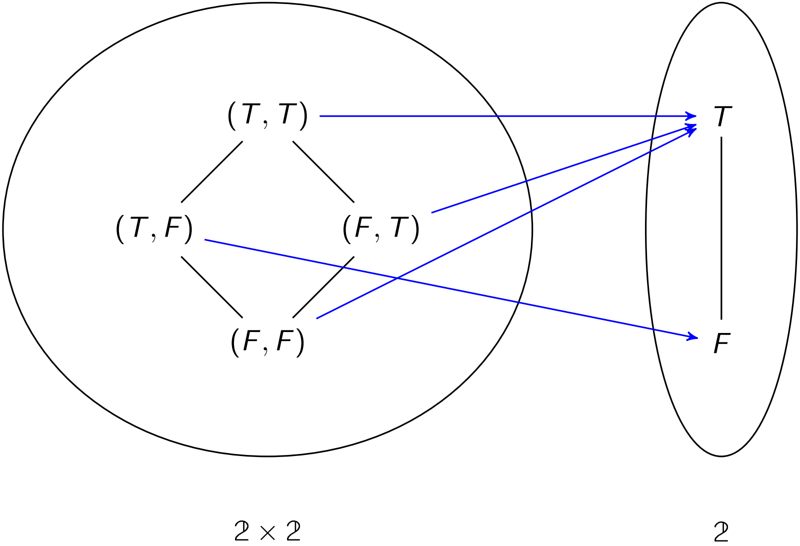
$\mathbb{2} \times \mathbb{2}$ $\mathbb{2}$

Disjunction ∨ as a monotone function

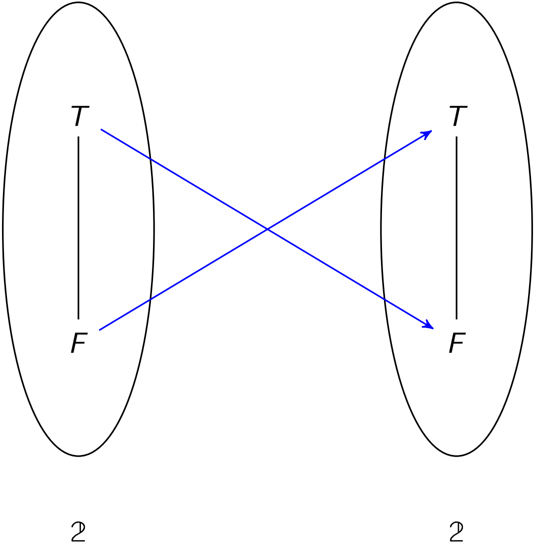


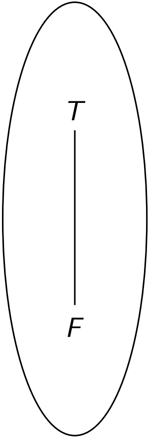
What about implication →?

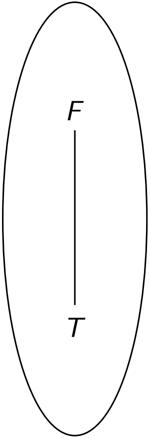
Is it a monotone function from 2×2 to 2?



Is negation monotone?







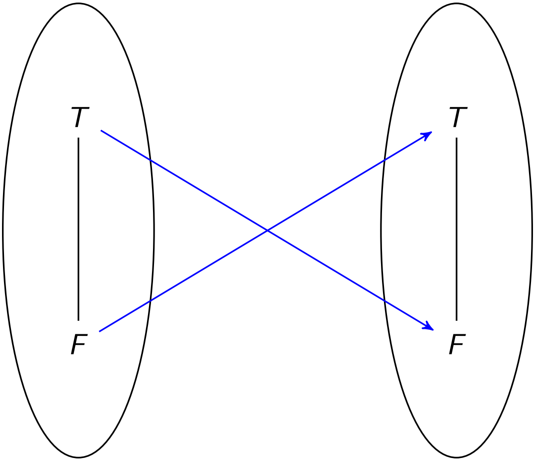
$\mathbb{2}$

$-\mathbb{2}$

$= \mathbb{2}$ upside down

Negation is antitone

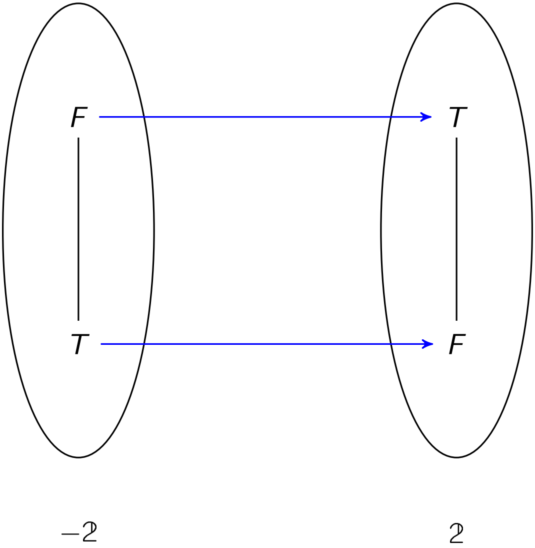
This is the same as a monotone function from $-\mathbb{2}$ to $\mathbb{2}$



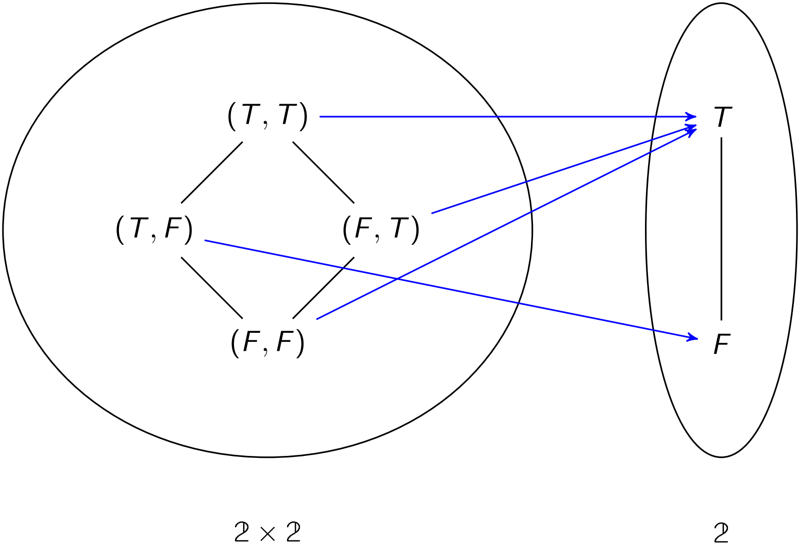
$\mathbb{2}$ $\mathbb{2}$

Negation is antitone

This is the same as a monotone function from -2 to 2

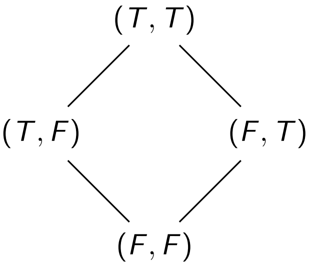


Let's go back to implication →

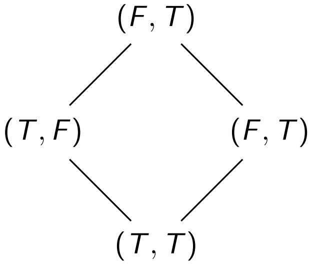


Find a preorder $\mathbb{P}$ so that
$\rightarrow$ is a monotone function from $\mathbb{P}$ to $\mathbb{2}$.

Hint: it's not $-(\mathbb{2} \times \mathbb{2})$, but this is on the right track.



$\mathbb{2} \times \mathbb{2}$

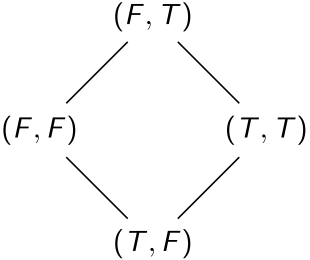


$-(\mathbb{2} \times \mathbb{2})$

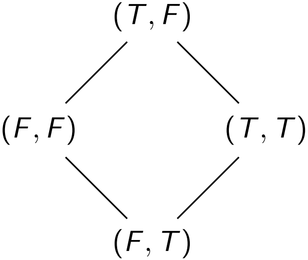
Question

Find a preorder $\mathbb{P}$ so that
$\rightarrow$ is a monotone function from $\mathbb{P}$ to $\mathbb{2}$.

Hint: try the orders below:



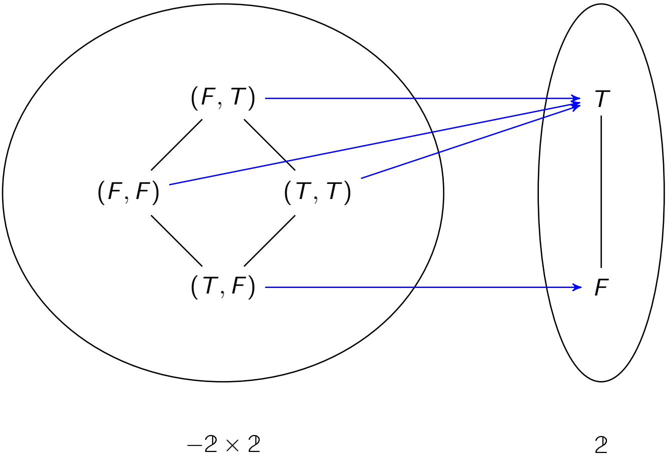
$-\mathbb{2} \times \mathbb{2}$



$\mathbb{2} \times -\mathbb{2}$

NOW WE CAN SETTLE THE MATTER ABOUT IMPLICATION $\longrightarrow$

IT IS A MONOTONE FUNCTION FROM -2×2 TO 2



The main fact that we need later

Definition

Let $\mathbb{P}$ and $\mathbb{Q}$ be preorders. Then

$$[\mathbb{P}, \mathbb{Q}]$$

is the set of all monotone functions $f : \mathbb{P} \to \mathbb{Q}$,
made into a preorder by declaring

$$f \leq g \quad \text{iff} \quad \text{for all } p \in P,\ f(p) \leq g(p) \text{ in } \mathbb{Q}$$

Fact

$$[\mathbb{P}, -\mathbb{Q}] = -[-\mathbb{P}, \mathbb{Q}]$$

This means that any lexical items typed as $\mathbb{P} \to -\mathbb{Q}$
could just as well be typed as $-\mathbb{P} \to \mathbb{Q}$.

However, the orders $[\mathbb{P}, -\mathbb{Q}]$ and $[-\mathbb{P}, \mathbb{Q}]$ are opposites.

Proposal, briefly

intended as a formalization of Dowty 1994

Take categorial grammar a la

Ajdukiewicz-Bar Hillel-Lambek-van Benthem

and interpret the syntactic types not in sets but in preorders, adding the ability to use opposite of a preorder as well.

van Benthem had the idea of using categorial grammar in order to formalize the $^{\uparrow}$, $^{\downarrow}$ notation which we saw earlier. His proposal was then worked out by Sanchez-Valencia.

One generates sentences in CG using ordinary words, and after a sentence is parsed, the proof tree is decorated with $^{\uparrow}$, $^{\downarrow}$ notations.

But Dowty noted that it would be useful to have grammars which directly generate words-plus-polarities. I'm going to formalize Dowty's alternative idea.

We begin with a set $\mathcal{T}_0$ of basic types: for simplicity *pr* and *t*.
We then form a set $\mathcal{T}_1$ of types as follows:

- $\mathcal{T}_0 \subseteq \mathcal{T}_1$.
- If $\sigma, \tau \in \mathcal{T}_1$, then also $(\sigma, \tau) \in \mathcal{T}_1$.
- If $\sigma \in \mathcal{T}_1$, then also $-\sigma \in \mathcal{T}_1$.

Let $\equiv$ be the smallest equivalence relation on $\mathcal{T}_1$ such that the following hold:

- $-(-\sigma) \equiv \sigma$.
- $-(\sigma, \tau) \equiv (-\sigma, -\tau)$.
- If $\sigma \equiv \sigma'$, then also $-\sigma \equiv -\sigma'$.
- If $\sigma \equiv \sigma'$ and $\tau \equiv \tau'$, then $(\sigma, \tau) \equiv (\sigma', \tau')$.

The set of types

$\mathcal{T} = \mathcal{T}_1/\equiv$.

Determiners give constants, two each:

every^+	$(-pr,(pr,t))$	every^-	$(pr,(-pr,-t))$
some^+	$(pr,(pr,t))$	some^-	$(-pr,(-pr,-t))$
no^+	$(-pr,(-pr,t))$	no^-	$(pr,(pr,-t))$
any^+	$(-pr,(pr,t))$	any^-	$(-pr,(-pr,-t))$

Every intransitive verb such as 'runs' (and every plural noun) also gives two constants:

$\text{runs}^+ : pr$ $\qquad$ $\text{runs}^- : -pr$

Every transitive verb such as 'see' gives four constants:

see_1^+	$((pr,t),pr)$	see_2^+	$((-pr,t),pr)$
see_1^-	$((-pr,-t),-pr)$	see_2^-	$((pr,-t),-pr)$

'If' also gives two constants:

$\text{if}^+ : (-t,(t,t))$ $\qquad$ $\text{if}^- : (t,(-t,-t))$

For the semantics we use models $\mathcal{M}$.

$\mathcal{M}$ consists of an assignment of preorders $\sigma \mapsto \mathbb{P}_\sigma$ on $\mathcal{T}_0$,

$$pr \mapsto [\mathbb{X}, \mathbb{2}] \qquad\qquad t \mapsto \mathbb{2}$$

extended to $\mathcal{T}_1$ by

$$\begin{array}{lcll} \mathbb{P}_{(\sigma,\tau)} & = & [\mathbb{P}_\sigma, \mathbb{P}_\tau] & \text{monotone function preorder} \\ \mathbb{P}_{-\sigma} & = & -\mathbb{P}_\sigma & \text{opposite preorder} \end{array}$$

If $\sigma \equiv \tau$, then $\mathbb{P}_\sigma = \mathbb{P}_\tau$.

We use P_σ to denote the set underlying the preorder $\mathbb{P}_\sigma$.

The rest of the structure of $\mathcal{M}$ consists of an assignment $[\![c]\!] \in P_\sigma$ for each constant $c : \sigma$.

Some semantic interpretations

$\mathbb{X}$ is the flat preorder on an arbitrary set X
$[\mathbb{X}, \mathbb{2}]$ is in one-to-one correspondence with the set of subsets of X.

Define

$$\begin{array}{lcll} \text{every} & \in & [-[\mathbb{X},\mathbb{2}],[[\mathbb{X},\mathbb{2}],\mathbb{2}]] & = \mathbb{P}_{(-pr,(pr,t))} \\ \text{some} & \in & [[\mathbb{X},\mathbb{2}],[[\mathbb{X},\mathbb{2}],\mathbb{2}]] & \\ \text{no} & \in & [-[\mathbb{X},\mathbb{2}],[-[\mathbb{X},\mathbb{2}],\mathbb{2}]] & \end{array}$$

in the standard way:

$$\text{every}(p)(q) = \begin{cases} \text{true} & \text{if } p \leq q \\ \text{false} & \text{otherwise} \end{cases}$$

$$\text{some}(p)(q) = \neg\text{every}(p)(\neg \circ q)$$

$$\text{no}(p)(q) = \neg\text{some}(p)(q)$$

It follows from the Main Fact above that

$$\begin{array}{lcll} \text{every} & \in & [[\mathbb{X},\mathbb{2}],[-[\mathbb{X},\mathbb{2}],-\mathbb{2}]] & = \mathbb{P}_{(pr,(-pr,-t))} \\ \text{some} & \in & [-[\mathbb{X},\mathbb{2}],[-[\mathbb{X},\mathbb{2}],-\mathbb{2}]] & \\ \text{no} & \in & [[\mathbb{X},\mathbb{2}],[[\mathbb{X},\mathbb{2}],-\mathbb{2}]] & \end{array}$$

$$\dfrac{\mathsf{chase}_1^- : ((-pr, -t), -pr) \qquad \dfrac{\mathsf{every}^- : (pr, (-pr, -t)) \quad \mathsf{cat}^+ : pr}{\mathsf{every}^-(\mathsf{cat}^+) : (-pr, -t)}}{\mathsf{chase}_1^-(\mathsf{every}^-(\mathsf{cat}^+)) : -pr}$$

$$\mathsf{some}^+(\mathsf{dog}^+)(\mathsf{chase}_1^+(\mathsf{every}^+(\mathsf{cat}^-))) : t$$

$$\mathsf{some}^+(\mathsf{dog}^+)(\mathsf{chase}_2^+(\mathsf{no}^+(\mathsf{cat}^-))) : t$$

$$\mathsf{no}^+(\mathsf{dog}^-)(\mathsf{chase}_2^-(\mathsf{no}^+(\mathsf{cat}^+))) : t$$

THEOREM

The $+$, $-$ signs automatically indicate the monotonicity $\uparrow$ and $\downarrow$.

Another

Everything which sees any cat runs

$$\dfrac{\text{any}^- : (-pr,(-pr,-t)) \quad \text{cat}^- : -pr}{\text{any}^-(\text{cat}^-) : (-pr,-t)}$$

$$\dfrac{\text{see}_2^- : ((-pr,-t),-pr) \quad \text{any}^-(\text{cat}^-) : (-pr,-t)}{\text{see}_2^-(\text{any}^-(\text{cat}^-)) : -pr}$$

$$\dfrac{\text{every}^+ : (-pr,(pr,t)) \quad \text{see}_2^-(\text{any}^-(\text{cat}^-)) : -pr}{\text{every}^+(\text{see}_2^-(\text{any}^-(\text{cat}^-))) : (pr,t)}$$

$$\dfrac{\text{every}^+(\text{see}_2^-(\text{any}^-(\text{cat}^-))) : (pr,t) \quad \text{runs}^+ :}{\text{every}^+(\text{see}_2^-(\text{any}^-(\text{cat}^-))(\text{runs}^+) : t}$$

Note that any^+ and any^- should not have the same interpretation!!

$$\text{any}^- = \text{some}^- \qquad\qquad \text{any}^+ = \text{every}^+$$

Compare

$$\text{any}^+(\text{cat}^-)(\text{see}_1^-(\text{any}^+(\text{dog}^-))) : t.$$

$$\frac{}{t : \sigma \leq t : \sigma} \qquad \frac{t : \sigma \leq u : \sigma \quad u : \sigma \leq v : \sigma}{t : \sigma \leq v : \sigma}$$

$$\frac{u : \sigma \leq v : \sigma \quad t : (\sigma, \tau)}{t(u) : \tau \leq t(v) : \tau} \qquad \frac{u : (\sigma, \tau) \leq v : (\sigma, \tau) \quad t : \sigma}{u(t) : \tau \leq v(t) : \tau}$$

But it's open to get completeness for this logic,
and in fact there are interesting questions:

$$\mathsf{every}^{+}(\mathsf{see}_1^{-}(\mathsf{every}^{-}(\mathsf{cat}^{+})))(\mathsf{see}_1^{+}(\mathsf{every}^{+}(\mathsf{cat}^{-})))$$

$$\mathsf{every}^{+}(\mathsf{see}_1^{-}(\mathsf{any}^{-}(\mathsf{cat}^{+})))(\mathsf{see}_1^{+}(\mathsf{any}^{+}(\mathsf{cat}^{-})))$$

What is the point of this logic? Any logic?

For me:

- It would be a step towards a complete logic for a significant language

For those in RTE:

- The sound principles give transformation rules.
- Completeness would be secondary.
- Logical systems are often implemented, and then this could be useful.

Living in two worlds

Work in natural logic continues the ideas of Aristotle and Leibniz, but also hopes to have something to say to Watson