Remarks on Categorial Semantics of Natural Language:

Are Propositions Propositions?

Carl Pollard

Department of Linguistics
Ohio State University

November 13, 2010

Point of Departure: Montague Semantics

- Montague was first to systematically apply the methods of mathematical logic to the analysis of natural language (NL) meaning. (late 1960s).
- A great deal of the subsequent history of (NL) semantics has consisted of attempts to repair, improve, or elaborate on Montague semantics (MS).
- Lambek made a significant though little-known contribution to this enterprise (Tuscon Categorial Grammar Conference, 1985).
- In this talk I will try to describe Lambek's contribution and what became of it.

Why I Care about This (1/2)

- Jon Barwise taught a course on MS at Stanford c. 1981.
- Barwise didn't like MS. He called it
  - a Rube Goldberg machine
  - a three-headed monster
  - a hodge-podge of Frege, Carnap, and Kripke
- Barwise was teaching MS in order to explain why it should be replaced by **situation semantics** (SS), the theory that he was developing together with John Perry.
- But unlike MS, the technical details of SS never got worked out enough to make it usable for linguistics.

Why I Care about This (2/2)

- Sometime in the late 1990's Shalom Lappin and I thought the time had come to fill in the missing details about SS and make it usable by linguists.
- But it soon became evident that it would be easier to just fix what was wrong with MS.
- Surveying the literature, we saw that Lambek's 1985 Tuscon talk (published 1988) made a pretty good start.
- So we started there. More about that in due course, but first we must backtrack.

Sources of Montague Semantics: Frege 1892

- A (NL) expression has a **sense** (which doesn't depend on how things are) and a **reference** (which does).
- For a declarative sentence, the sense is a **proposition** and the reference is that proposition's **truth value**.
- The sense of an expression is a function of the senses of its syntactic constituents.

  (Frege thought references worked this way too, but that was not such a good idea.)

Sources of Montague Semantics: Carnap 1947

- Worlds are **complete state descriptions**, sets of closed formulas in a certain logical language.
- Senses of (NL) expressions are **intensions**, functions whose domain is the set of worlds.
- For sentences, the intensions map to truth values.
- To get the reference of an expression at a world, apply its intension to that world as an argument.

Kripke 1963

Worlds are taken as unanalyzed **primitives** (contra Carnap and earlier Kripke)

Montague Grammars

- Montague's logical theory of meanings is part of an account of how words strings get associated with intensions.
- The relationship between strings and intensions is mediated by a primitive categorial grammar (primitive in the sense of lacking hypothetical proof).
- The grammar defines a set of ordered triples of (1) a string, (2) a syntactic type, and (3) an intension.
- Some of the triples are given in advance (the **lexicon**).
- Each grammar rule is equipped with a recipe for combining the strings of the constituents to get a new string (usually by concatenation), and another recipe for combining the intensions of the constituents to get a new intension (usually by function application).

The Types of Montague's Semantic Theory

- The theory was written in an idiosyncratic higher-order language (no proof theory) called IL.
- But Gallin (1975) showed how to translate IL into Henkin's (1950) HOL, so we'll ignore IL.
- Besides the truth value type t provided by the logic, there are two basic types:
  - e, the type of **entities**
  - w (Montague's s), the type of **worlds**. (Here inspired by Kripke 1963, not Carnap or Kripke 1959.)
- The type p of propositions is *defined* to be w → t (sets of worlds). This follows Carnap, modulo replacement of complete state descriptions by primitive worlds.

Propositions in Montague Semantics

- For p to be **true** at w is for w to be a member of p.
- The intensions for the NL ‘logic words’ are the expected boolean operation on propositions, e.g.:

$$\mathit{and} \rightsquigarrow \lambda_{wpq}.(p\,w) \wedge (q\,w) : \mathrm{p} \rightarrow \mathrm{p} \rightarrow \mathrm{p}$$

$$\mathit{implies} \rightsquigarrow \lambda_{wpq}.(p\,w) \rightarrow (q\,w) : \mathrm{p} \rightarrow \mathrm{p} \rightarrow \mathrm{p}$$

  i.e. intersection and relative complement of sets of worlds.
- The centrally important relation of NL semantics, **entailment**, is modelled by subset inclusion in w → t:

$$\mathsf{entails} =_{\mathrm{def}} \lambda_{pq}.\forall_{w}.(p\,w) \rightarrow (q\,w) : \mathrm{p} \rightarrow \mathrm{p} \rightarrow \mathrm{t}$$

- At first blush, this seems right because intuitively, for p to entail q is supposed to mean that, no matter how things are, if p is true with things that way, then so is q.

A Granularity Problem: Logical Omniscience

But there are problems with assuming propositions are sets of worlds. Here we can consider just a couple.

- As just shown, in MS entailment is modelled as the subset inclusion order in the set of sets of worlds.
- So mutually entailing propositions are equal.
- Then there is only one necessary truth, namely the set of all worlds.
- So anybody who knows some necessary truth (e.g. that Sarkozy is Sarkozy) knows them all (e.g. the Riemann Hypothesis or its denial, whichever is true).

Another Granularity Problem: Donkeys and Asses

- *Chiquita is a donkey* and *Chiquita is an ass* express mutually entailing propositions (call them p and q), so MS treats them as identical.
- But maybe Pedro believes the first but not the second.
- Under the standard assumption that belief is a relation between entities and propositions, that is only possible if p and q are distinct.
- The moral: entailment better not be antisymmetric.

Lambek's (1985) Categorical Semantics (CS)

Lambek's contribution to semantics was to observe that MS is anything but a three-headed monster. Instead:

- Expressions are morphisms in a **residuated category**.
- Senses are morphisms in a **topos**.
- Semantic interpretation is a **residuated functor** from expressions to senses.
- Sentence senses are morphisms $: 1 \to \Omega$
  "(Linguists') propositions are (topos) propositions."

Let's unpack this a little.

Expressions are Morphisms in a Residuated Category

- Replace Montague's categorial grammar with the **Lambek calculus**, the (⊗, /, \, I)-fragment of intuitionistic noncommutative linear logic.
- Then upgrade it to a residuated category (RC).
- RC's are related to Lambek calculus just as CCC's are related to positive $(\wedge \rightarrow, 1)$-intuitionistic logic:
  - Just as a CCC order is a heyting semilattice, an RC order is a residuated monoid.
  - Just as a CCC is equivalent to a positive typed λ-calculus, an RC is equivalent to a (Buszkowski) calculus (with left and right versions of **eval** and λ and no structural rules).
  - Just as in a CCC, morphisms in an RC are reifications of (equivalence classes of) proofs.
- Hence expressions are reified proofs of a Lambek calculus, and the syntactic types (sentence, noun phrase, etc.) are the objects of the RC.

Senses are Morphisms in a Topos

- Replace Montague's IL (or Henkin's HOL) with an intuitionistic type theory along the lines of Lambek and Scott (LS) 1986, and Montague's Henkin models with the topos which has that type theory as its internal language.
- We could add LEM, but don't insist on it.
- Recall that a topos is a CCC, and therefore an RC with $\otimes = \times$, $/ = \backslash = \rightarrow$, $\mathrm{I} = 1$.
- Model semantic types as objects of the topos.
- Model senses themselves as morphisms of the topos.

Semantic Interpretation is a Residuated Functor

- Model semantic interpretation as a **residuated** functor **Sem** from the RC of expressions to the topos of senses.
- Hence at the level of objects/types, **Sem** translates $\otimes$ as $\times$, $/$ and $\backslash$ as $\rightarrow$, and I as 1.
- And at the level of morphisms/terms, **Sem** translates ${}_r\lambda$ and ${}_l\lambda$ as λ, $\mathbf{eval}_r$ and $\mathbf{eval}_l$ as **eval**, and the senses of **words** (morphisms corresponding to nonlogical axioms) have to be stipulated (the **lexicon**).
- "Le sens provient toujours de la traduction d'une théorie dans une autre."

Are Propositions Propositions?

- Up to this point, CS has to be judged a success: it shows that Montague was well on the way to having an elegant and natural theory. But:
- Lambek proposes that the sense of a declarative sentence is a morphism $p : 1 \to \Omega$.
- But Ω is an (internal) heyting algebra, which we further require to be boolean (not necessarily insisting on LEM).
- So CS also suffers from the antisymmetry of entailment.
- But in fact things are far worse than that.

Entailment vs. Implication in MS

- Recall that in MS, entailment is

$$\textsf{entails} =_{\text{def}} \lambda_{pq}.\forall_w.(p\,w) \to (q\,w) : \text{p} \to \text{p} \to \text{t}$$

  In particular, it is a **relation** on propositions.

- Whereas implication is

$$\mathit{implies} \rightsquigarrow \lambda_{wpq}.(p\,w) \to (q\,w) : \text{p} \to \text{p} \to \text{p}$$

  In particular, it is an **operation** on propositions.

- This is as it should be.

Implication vs. Entailment in CS

- In CS, $\mathsf{entails}$ is the order in the internal BA Ω and $\wedge$ is the meet, and so we must define entailment as

$$\mathsf{entails} =_{\text{def}} \lambda_{pq}.(p \wedge q) = p$$

- But in LS type theory, $p \to q$ is also defined as $(p \wedge q) = p$.
- And so if ‘propositions are propositions’, the distinction between implication and entailment collapses!
- One example suffices to illustrate what a disaster that is.

The Total Omniscience Problem

- Let p, q, r, and s be the propositions expressed by the following four sentences:
  1. Chiquita is a donkey
  2. Frances is a mule.
  3. Pedro knows Chiquita is a donkey.
  4. Pedro knows Frances is a mule.

  so that $p = (\mathsf{donkey\ chiquita})$, $q = (\mathsf{mule\ frances})$, $r = (\mathsf{know}\ p\ \mathsf{pedro})$, and $s = (\mathsf{know}\ q\ \mathsf{pedro})$.
- Then we can easily prove (internally):

  $$\vdash (p \wedge q \wedge r) \rightarrow s$$

  (Hint: $p \wedge q$ is defined as $\langle p, q \rangle = \langle \mathsf{true}, \mathsf{true} \rangle$, and so $\vdash (p \wedge q) \rightarrow (p = q)$.)
- More generally: anyone who knows some (possibly contingent) truth is totally omniscient (knows *every* truth, not just necessary ones)

Denouement

- I noticed the Total Omniscience problem in 1999.
- Shalom Lappin became disillusioned with toposes and developed a different approach with Chris Fox (*Foundations of Intensional Semantics*, 2005).
- I was lucky to get some good advice:
  - Drew Moshier and Bill Lawvere (separately) suggested (2000) using an internal algebra p distinct from Ω for sentence senses.

    (This is reminiscent of Thomason's 1980 intentional logic, with a basic type p distinct from t),
  - Howard Gregory (2001) emphasized that although this solves Total Omniscience, it's no help with Logical Omniscience because entailment is still antisymmetric.

Hyperintensional Categorical Semantics (HCS)

- We stick with CS, except we model sentence senses using an internal boolean *pre*order object p, rather than Ω.
- We still have basic type e for entities.
- We rename Ω to t (for ‘truth values’), reserving the term ‘proposition’ for morphisms $1 \to \mathrm{p}$.
- We also require that p have ‘weakly enough’ ultrafilters, i.e. enough to separate any two propositions which are not mutually entailing. (We could have this for free if the topos had IAC, but that’s more than we need.)
- So we have a weak form of Stone duality for propositions, weak in the sense that the Stone embedding is not monic.
- This is what we need for the preorder on p to be able to model entailment.

Worlds in HCS

- We *define* the type w to be the subobject $\mu : \mathrm{w} \rightarrowtail (\mathrm{p} \to \mathrm{t})$ of the ultrafilters of p.
- In the internal language, we define

  $$@ =_{\mathrm{def}} \lambda_{pw}.(m\ w\ p) : \mathrm{p} \to \mathrm{w} \to \mathrm{t}$$

  where $m : \mathrm{w} \to (\mathrm{p} \to \mathrm{t})$ denotes $\ulcorner \mu \urcorner$.
- So the way to say 'p is true at w' is $p@w$. This says p is one of the propositions belonging to the ultrafilter w.
- Then the axiom that p have weakly enough ultrafilters is:

  $$\vdash \forall_{pq}.\neg(p \ \mathsf{entails}\ q) \to \exists_w.p@w \wedge \neg q@w$$

Extensional and Intensional Types in HCS

- We define the **hyperintensional** types to be p, e, and types obtained from these using the type constructors.
- For each hyperintensional type A, the corresponding **extensional** type $\mathrm{Ext}(A)$ is defined as follows:

$$\begin{aligned}
&\mathrm{Ext}(\mathrm{p}) = \mathrm{t}\\
&\mathrm{Ext}(\mathrm{e}) = \mathrm{e}\\
&\mathrm{Ext}(1) = 1\\
&\mathrm{Ext}(A \times B) = \mathrm{Ext}(A) \times \mathrm{Ext}(B)\\
&\mathrm{Ext}(A \to B) = A \to \mathrm{Ext}(B)
\end{aligned}$$

- and the corresponding **intensional** type $\mathrm{Int}(A)$ is defined as $\mathrm{w} \to \mathrm{Ext}(A)$.
- So there are intensions.
- But they aren't the senses; hyperintensions are.

Extensions at Worlds in HCS

- The **extension** of a hyperintension $a : A$ at a world w, written $a@w : \mathrm{Ext}(A)$, is defined as follows:
  - This was already defined for $A = \mathrm{p}$.
  - $a@w = a$ for $A = \mathrm{e}$.
  - $*@w = *$
  - $\langle a, b\rangle @w = \langle a@w, b@w\rangle$
  - $a@w = \lambda_x.(a\,x)@w$ for $A = B \to C$.
    (This one is why I don't like Frege's idea that reference is compositional.)
- For each hyperintensional type A, the **intensionalizer** morphism is

$$\mathbf{int}_A =_{\mathrm{def}} \lambda_{xw}.x@w : A \to \mathrm{Int}(A).$$

- $\mathbf{int}\,a$ is called the **intension corresponding to** a.

Propositions and their Intensions

- $\mathbf{int}_{\mathrm{p}} : \mathrm{p} \rightarrow \mathrm{w} \rightarrow \mathrm{t}$ is the Stone dual mapping that maps each proposition to the set of worlds which contain it
- Hence the family of morphisms $\mathbf{int}_A$ amounts to a generalized Stone dual at all hyperintensional types.
- For each $p : \mathrm{p}$, $\mathbf{int}\, p$ is a morphism from worlds to truth values.
- Hence $\mathbf{int}\, p$ is much like a Carnapian proposition, modulo the replacement of 'complete state descriptions' of (syntactic!) formulas by (semantic!) ultrafilters.

The Big Differences between HCS and MS

- MS is written in classical HOL; HCS in LS type theory.
- In MS propositions are sets of worlds; in HCS it is the other way around.
- More generally: in MS meanings are intensions; in HCS meanings are hyperintensions and their Stone duals are intensions.
- The reason intensional semantics is not fine-grained enough is because the Stone dual mapping isn't monic.