UCLA Working Papers in Linguistics, Theories of Everything
Volume 17, Article 12: 80-84, 2012

Proof-theoretic type interpretation: a glimpse to proof-theoretic semantics

Nissim Francez

Dedication: To Ed Keenan, a champion of model-theoretic semantics, on the occasion of his retirement, with best wishes for a long continual of scientific work.

1 Introduction

A foundation of model-theoretic semantics (MTS) for natural language (NL), ever since Montague's seminal work, is the typing of meanings, most often expressed in some variant of the simply-typed λ-calculus. Types are interpreted in what is known as Henkin models, whereby basic types τ are interpreted as denoting arbitrary sets D_τ, except for the type t (of sentential meanings), denoting the two-valued boolean algebra of truth-values $D_t = \{t, f\}$. Functional types (τ, σ) denote $D_\sigma^{D_\tau}$ the collection of all functions from the domain type D_τ to the range type D_σ.

The aim of this note is the presentation of new results; rather, it is the highlighting, in a nutshell, of a proof-theoretic interpretation of types, originating in Francez, Dyckhoff, and Ben-Avi (2010), used by *proof-theoretic semantics (PTS)* for NL, thereby opening a small window to the latter theory of meaning, unfortunately very little known to most linguists.

Before presenting the details of the proof-theoretic type interpretation, I recapitulate the essence of the PTS as applied to NL:

- *For sentences, replace the received approach of taking their meanings as* **truth conditions** *(in arbitrary models) by an approach taking meanings to consist of* **canonical derivability conditions** *(from suitable assumptions).* In particular, this involves a "dedicated" proof-system in natural deduction (ND) form, on which the derivability conditions are based. In a sense, the proof system should reflect the "use" of the sentences, and should allow recovering pre-theoretic properties of the meanings of these sentences such as entailment and assertability conditions. For some discussion of the criticism of MTS as a theory of meaning see Francez and Dyckhoff (2010).

  An important requirement is that the ND-system should be *harmonious* (see Francez and Dyckhoff (2010) for a discussion of harmony of NL ND-rules), in that its rules have a certain balance between introduction and elimination, in order to qualify as meaning conferring.

- For *sub-sentential phrases*, replace their denotations (extensions in arbitrary models) as

their meanings, by their **contributions** to the meanings (in our explication, derivability conditions) of sentences in which they occur. This adheres to Frege's *context principle*, made more specific by the incorporation into a TLG (see Francez et al. (2010) for the process of extracting meanings for sub-sentential phrases from sentential meanings).

2 Sentential meanings: the proof-theoretic type interpretation of type t

The proof-theoretic meaning for NL sentences is based on a "dedicated" natural-deduction proof-system, with *introduction* rules (I-rules) and *elimination* rules (E-rules) for the various constructs of the NL in case. For a sentence S containing such a construct, an I-rule defines how can be S derived from other sentences, while an E-rule defines which (immediate) conclusions can be derived from S (possibly using other auxiliary sentences. For "primitive" sentences (containing *no* construction), the meaning is assumed *given*. In Francez and Dyckhoff (2010), such an ND-system is presented for an extensional fragment of English containing intransitive and transitive verbs, (count) nouns, determiners, (intersective) adjectives, relative clauses, proper names and a copula. The paper also presents an extension with intensional intransitive verbs with an unspecific object.

Suppose such an ND-system N is given. *derivations* (ranged over by $\mathscr{D}$) are defined recursively by iterating applications of rules. A derivation is from a (possibly empty) collection Γ of sentences, to a *conclusion* S. *derivability* (in N) of S from Γ is denoted by $\Gamma \vdash_N S$. Derivations are depicted as a tree, with members of Γ as leaves and S as the root.

There is a special kind of derivations (underlying the definition of sentential meanings) called em canonical derivations. Such derivations consist of the most *direct* way of concluding S.

Canonical derivation: A derivation (in N) is *canonical* iff its last rule application is of an I-rule. Canonical derivability of S from Γ is denoted by $\Gamma \vdash^c_N S$. Let $[\![S]\!]^c_\Gamma$ denote the (possibly empty) collection of canonical derivations of S from Γ.

Sentential meanings: The (reified) *meaning* of a sentence S is defined by

$$[\![S]\!] =^{df.} \lambda\Gamma.[\![S]\!]^c_\Gamma$$

Thus, the meaning of S consists of *all* its canonical derivations from arbitrary Γs. Some properties of this prof-theoretic meanings are summarized below.

- The meaning of a sentence does not depend on any special "logical form", different from its surface form.
- The meaning is of a finer granularity then the MTS truth-conditions, for example nor rendering logically equivalent sentences as having the same meaning.
- Such meanings may serve as more adequate arguments for propositional attitudes than the corresponding truth-conditions of MTS.
- Most importantly, such meanings do not impose any *ontological commitments* like the ones that are imposed by the structure of models. They are expressed using purely syntactic, formal expressions.

Based on these reified proof-theoretic meanings, the proof-theoretic interpretation of type t can now be defined as follows.

Proof-theoretic interpretation of type t:

$$D_t =^{df.} \{[\![S]\!] \mid S \text{ in the language}\}$$

Thus, the inhabitants of type t are all the sentential meanings.

For some purposes, those proof-theoretic meanings are *too fine grained*. There is a natural equivalence relation that can be imposed to somewhat coarsen the granularity of meanings.

Grounds of assertion: Every Γ s.t. $\Gamma \vdash^c_N S$ is a *grounds for assertion* of S. Let $G[\![S]\!] =^{df.} \{\Gamma \mid \Gamma \vdash^c_N S\}$ be the (possibly empty) collection of all grounds of assertion for S.

Thus, S is warranty asserted by anyone in posession[1] of some $\Gamma \in [\![S]\!]$. When '$\vdash_N$' is decidable (which most often is the case), warranted assertion is effective.

Using grounds of assertion, the following natural equivalence relation on meanings can be imposed.

$$S_1 \equiv_G S_2 \text{ iff } G[\![S_1]\!] = G[\![S_2]\!]$$

Thus, sentences with identical grounds of assertion are rendered as having equivalent meanings.

3 Sub-sentential meanings: more types and their proof-theoretic interpretation

As described in detail in Francez and Dyckhoff (2010), a natural ND-system for NL uses a denumerable collection $\mathscr{P}$ of *individual parameters*. These are syntactic objects, not used in the NL itself, only in its extension for purposes of expressing rules and derivations. Meta-variables in boldface font, **j, k**, range over individual parameters; syntactically, such parameters are dps; $S[\mathbf{j}]$, containing a parameter in some dp-position, is a *pseudo-sentence*, present only in the proof-language extending the NL. Let p be a basic type, with $D_p = \mathscr{P}$. Type p is the counterpart of the the Montagovian type e; however, while D_e is *arbitrary*, D_p is *fixed*, containing only syntactic inhabitants. The general type of a predicate is the functional type (p,t).

There is a means for forming certain *subtypes*, for some of the more frequently used functional types, where the argument parameter has to occupy some position in a pseudo-sentence type (i.e., preventing constant functions).

- t_p is a subtype of (p,t), s.t. $D_{t_p} = \{\lambda\mathbf{j}.[\![S[\mathbf{j}]]\!] \mid S[\mathbf{j}]\ a\ (pseudo)sentence\}$.
- $t_{p,p}$ is a subtype of $(p,(p,t))$, s.t. $D_{t_{p,p}} = \{\lambda\mathbf{k}\lambda\mathbf{j}.[\![S[\mathbf{j},\mathbf{k}]]\!] \mid S[\mathbf{j},\mathbf{k}]\ a\ (pseudo)sentence\}$.
- n is a subtype of (p,t), s.t. $D_n = \{\lambda\mathbf{j}.[\![\mathbf{j}\ \ is\ a\ X]\!] \mid X\ a\ noun\}$.

Note that there are two different predicate types. One, t_p is *verbal*, and the other, n, which is basic, is *nominal*. This distinction plays a major role in the definition of the type of *determiners* (see below).

[1] A cognitive term, left unexplicated here.

3.1 Meanings of nouns and verbs

The meanings of nouns and verbs originate from (given) meanings of ground pseudo-sentences. For verbs, the ground sentence is the sentence headed by the verb. Accordingly, the meaning of an intransitive verb P of type t_p is $[\![P]\!] = \lambda\mathbf{j}.[\![\mathbf{j}\ P]\!]$. Similarly, the meaning of a transitive verb R of type $t_{p,p}$ is $[\![R]\!] = \lambda\mathbf{k}\lambda\mathbf{j}.[\![\mathbf{j}\ R\ \mathbf{k}]\!]$. The meaning of a noun X of type n is given by $[\![X]\!] = \lambda\mathbf{j}.[\![\mathbf{j}\quad \textit{is a}\ X]\!]$.

3.2 Meaning of determiners

A (regular) determiner combines with a (count) noun and a verb-phrase to form a sentence. The meaning of a determiner is *extracted* (as described precisely in Francez et al. (2010)) from the sentential meanings in which the determiner occurs. Their general form of the proof-theoretic meaning of a basic determiner D is

$$[\![D]\!]^{df.} = \lambda z_1^n \lambda z_2^{t_p} \lambda\Gamma. \bigcup_{\mathbf{j}_1,\ldots,\mathbf{j}_m \in \mathscr{P}} I_D(z_1)(z_2)(\mathbf{j}_1)\cdots(\mathbf{j}_m)(\Gamma)$$

Here z_1 is the meaning of a noun, say X, z_2 is the meaning of a verb-phrase, say V, and I_D is a function applying the I-rule corresponding to D to derivations of the noun and the vp. The result is the meaning of the sentence $S = D\ X\ V$. For example, for $D = \mathsf{every}$, $z_1 = [\![\mathsf{girl}]\!]$ and $z_2 = [\![\mathsf{smiles}]\!]$, one gets

$$[\![\mathsf{every}]\!]([\![\mathsf{girl}]\!])([\![\mathsf{smiled}]\!]) = [\![\mathsf{every\ girl\ smiled}]\!]$$

as expected.

In Francez (2012), determiners are studied in detail. There, the proof-theoretic meaning of complex determiners like *possessives* and *coordinated determiners* is given too. For handling *negative* determiners such as no, the PTS moves to *bilateralism*, where *denial* is taken on par with assertion. I-rules are provided both for asserting and for denial. This is reflected in a change of sentential meanings, "hidden" under the inhabitants of type t.

The main result of Francez (2012) is the following theorem.

Theorem: (conservativity) Every determiner is conservative in at least one of its argument.

Thus, instead of *stipulating* the conservativity of determiners, as is the case in the MTS using generalized quantifiers as dp-denotations, conservativity is proved! Note that the proof-theoretic meaning as defined above is much more restrictive than the MTS counterpart. The is no way to express non-conservative GQs such as the following. Let A and B be arbitrary subsets of the domain E of any model.

$$G_1(A)(B) \Leftrightarrow |A| > |B|, \quad G_2(A)(B) \Leftrightarrow |A| = |B| \quad G_3(A)(B) \Leftrightarrow (E - A) \subseteq B$$

Another discrepancy of determiners cannot arise: dependency of their MT-denotation on the cardinality of the domain. For example, a definition like

$$[\![D]\!] = \{ \begin{array}{ll} [\![\mathsf{every}]\!] & |E| \geq 100 \\ [\![\mathsf{some}]\!] & |E| < 100 \end{array}$$

4 Conclusions

This note presented a proof-theoretic interpretation of types, not using models, entities or any other ontologically committing sort of machinery. Only syntactic expressions, resulting from derivations in an ND-system, are used. Another example, using an additional primitive type (not ontologically committing to anything), handling *non-specific* objects of intensional transitive verbs, such as

every lawyer needs a secretary

known to be hard (and controversial as to the right models and truth-conditions needed) in MTS, can be found in Francez and Dyckhoff (2010).

This note presents in a nutshell only some of the main ideas involved in applying PTS to NL. Readers interested in fuller presentation, including many concrete examples, are encouraged to read the cited papers.

References

Francez, Nissim. 2012. A proof-theoretic reconstruction of generalized quantifiers. *Journal of Semantics* (under refereeing).

Francez, Nissim, and Roy Dyckhoff. 2010. Proof-theoretic semantics for a natural language fragment. *Linguistics and Philosophy* 33:447–477.

Francez, Nissim, Roy Dyckhoff, and Gilad Ben-Avi. 2010. Proof-theoretic semantics for subsentential phrases. *Studia Logica 94* 381–401.

Affiliation

Nissim Francez
Technion - Israel Institute of Technology
francez@cs.technion.ac.il