

# 6th International Workshop on Logic and Engineering of Natural Language Semantics (LENLS 6)

Daisuke Bekki

Ochanomizu University, 2-1-1 Ohtsuka, Bunkyo-ku, Tokyo 112-8610, Japan  
bekki@is.ocha.ac.jp

The annual international workshop LENLS (Logic and Engineering of Natural Language Semantics) covers topics in formal linguistics and related fields, such as theoretical computer science, mathematical logic and formal philosophy, and includes the following:

- ✂ Dynamic syntax/semantics/pragmatics of natural language
- ✂ Categorical/topological/coalgebraic approaches to natural language syntax/semantics/pragmatics
- ✂ Logic and its relation to natural language and linguistic reasoning (especially dynamic logics)
- ✂ Type-theoretic approaches to natural language
- ✂ Formal philosophy of language
- ✂ Formal pragmatics (especially game- and utility-theoretic approaches)
- ✂ Substructural expansion of Lambek Lambda Calculi
- ✂ Many-valued/Fuzzy and other non-classical logics and natural language

Formal linguistics is intrinsically an interdisciplinary field. Indeed, most of the formalisms which have been adopted in formal linguistics have their origin in the field of theoretical computer science, and in programming semantics especially. To take a few examples:

**Lambda calculus** originally appeared in Church (1941) and was subsequently incorporated into Montague Grammar (Montague (1973)).

**Dynamic logics** has its root in the programming semantics of the mid 1970s such as in the work of Harel (1979), which inspired the seminal paper by Groenendijk and Stokhof (1991).

**Continuations** first appeared in Plotkin (1975) and later in Danvy and Filinski (1990) before being extended to the field of formal semantics by de Groote (2001) and Barker (2001).

The average time for these notions to jump the boundary from the originating field into another was around 23 years. During such time they matured, became widely accepted in their respective fields as important notions, were introduced in many lectures, before eventually becoming known to formal linguists.

Besides programming semantics, other fields of theoretical computer science have influenced formal linguistics. Game theory, Bayesian networks and stochastic processes are among the many examples, and have helped to shape the discipline over time and remain influential in formal linguistics today.

On the other hand, only a very small number of theories emerging from formal linguistics have influenced theoretical computer science, to say nothing of to mathematical logic. Algebraic automata theory, which originates in automata theory, is one of the few exceptions. Lambek calculus is another exception, if we take the view that it comes from categorial grammar, which, together with linear and fuzzy logics, gives its own insights into substructural/algebraic logics.

This means that formal linguistics has been, rightly or wrongly, almost a ‘net-importer of formal theory’. Whether we accept this situation and want to accelerate the cycle of importing new formal theories, or whether we regard it as a problem and try to develop linguistic-oriented formal theories, it seems that one of our prime tasks is to promote and maintain people-to-people and theory-to-theory exchanges between formal linguistics and adjacent fields.

The aim of LENLS, in the spirit of its founder Norihiro Ogata, is to revitalize such exchanges, rediscover the connections between the formal disciplines (theoretical computer science, logic and mathematics) and empirical disciplines (syntax, semantics, pragmatics and philosophy), and revive the interdisciplinary nature of formal linguistics to what it once was.

In LENLS6, we focused particularly on work related to the interplay between logic, philosophy of language and formal semantics and pragmatics. This post-proceedings volume contains 8 papers, selected from the 20 presentations given at the LENLS 6 workshop.

LENLS6 was held as part of JSAI International Symposia on AI (JSAI-isAI 2009). I hope that LENLS will continue its international development, alongside the other workshops of JSAI-isAI, within this new scheme.

## References

- Barker, C.: Introducing Continuations. In: Hastings, R., Jackson, B., Zvolenszky, Z. (eds.) *The Proceedings of SALT 11*. CLC Publications, Ithaca (2001)
- Church, A.: *The Calculi of Lambda Conversion*. *Annals of Mathematical Studies*, vol. 6, ii+77. Princeton University Press, Princeton (1941)
- Danvy, O., Filinski, A.: Abstracting Control. In: *The Proceedings of LFP 1990, the 1990 ACM Conference on Lisp and Functional Programming*, pp. 151–160 (1990)
- de Groote, P.: Type raising, continuations, and classical logic. In: van Rooij, R., Stokhof, M. (eds.) *The Proceedings of the 13th Amsterdam Colloquium*, Institute for Logic, Language and Computation, Universiteit van Amsterdam, pp. 97–101 (2001)
- Groenendijk, J., Stokhof, M.: Dynamic Predicate Logic. *Linguistics and Philosophy* 14, 39–100 (1991)
- Harel, D. (ed.): *First-Order Dynamic Logic*. LNCS, vol. 68. Springer, Heidelberg (1979)
- Montague, R.: The Proper Treatment of Quantification in Ordinary English. In: Hintikka, J., Moravcsic, J., Suppes, P. (eds.) *Approaches to Natural Language*, pp. 221–242. Reidel, Dordrecht (1973)
- Plotkin, G.D.: Call-by-Name, Call-by Value and the Lambda Calculus. *Theoretical Computer Science* 1, 125–159 (1975)