

# Building compositional semantics and higher-order inference system for a wide-coverage Japanese CCG parser

Koji Mineshima<sup>1</sup>

mineshima.koji@ocha.ac.jp

Ribeka Tanaka<sup>1</sup>

tanaka.ribeka@is.ocha.ac.jp

Pascual Martínez-Gómez<sup>2</sup>

pascual.mg@aist.go.jp

Yusuke Miyao<sup>3</sup>

yusuke@nii.ac.jp

Daisuke Bekki<sup>1</sup>

bekki@is.ocha.ac.jp

<sup>1</sup>Ochanomizu University  
Tokyo, Japan

<sup>2</sup>AIST  
Tokyo, Japan

<sup>3</sup>National Institute of Informatics  
The Graduate University for Advanced Studies  
Tokyo, Japan

## Abstract

This paper presents a system that compositionally maps outputs of a wide-coverage Japanese CCG parser onto semantic representations and performs automated inference in higher-order logic. The system is evaluated on a textual entailment dataset. It is shown that the system solves inference problems that focus on a variety of complex linguistic phenomena, including those that are difficult to represent in the standard first-order logic.

## 1 Introduction

Logic-based semantic representations have played an important role in the study of semantic parsing and inference. For English, several methods have been proposed to map outputs of parsers based on syntactic theories like CCG (Steedman, 2000) onto logical formulas (Bos, 2015). Output formulas have been used in various tasks, including Question Answering (Lewis and Steedman, 2013) and Recognizing Textual Entailment (RTE) (Bos and Markert, 2005; Beltagy et al., 2013; Bjerva et al., 2014).

Syntactic and semantic parsing for Japanese, by contrast, has been dominated by chunk-based dependency parsing and semantic role labelling (Kudo and Matsumoto, 2002; Kawahara and Kurohashi, 2011; Hayashibe et al., 2011). Recently, the method of inducing wide-coverage CCG resources for English (Hockenmaier and Steedman, 2007) has been applied to Japanese and a robust CCG parser based on it has been developed (Uematsu et al., 2015). However, building a method to map CCG trees in Japanese onto logical formulas is not a trivial task,

mainly due to the differences in syntactic structures between English and Japanese (Section 3).

There are two primary contributions of this paper. First, based on an in-depth analysis of the syntax-semantics interface in Japanese, we present the first system that compositionally derives semantic representations for a wide-coverage Japanese CCG parser. Output representations are formulas in higher-order logic (HOL) combined with Neo-Davidsonian Event Semantics (Parsons, 1990). Second, we demonstrate the capacity of HOL for textual entailment. We evaluate the system on a Japanese textual entailment dataset (Kawazoe et al., 2015), a dataset constructed in a similar way to the FraCaS dataset for English (Cooper et al., 1994; MacCartney and Manning, 2007). Although it is usually thought that HOL is unfeasible for practical applications, the results show that the entire system is able to perform efficient logical inference on complex linguistic phenomena such as generalized quantifiers and intensional modifiers — phenomena that pose challenges to the standard first-order-logic-based approaches.

## 2 Background and system overview

This section provides a brief overview of the entire system as applied to RTE, a task of determining whether a given text ( $T$ ) entails, contradicts, or is just consistent with, a given hypothesis ( $H$ ). In logic-based approaches, the meanings of  $T$  and  $H$  are represented by logical formulas; whether the entailment relation holds is typically determined by checking whether  $T \rightarrow H$  is a theorem in a logical system with the help of a knowledge base.

Currently, first-order logic (FOL) is the most pop-

ular logical system used for RTE (Bos and Markert, 2005; Lewis and Steedman, 2013; Bjerva et al., 2014). One advantage of systems based on FOL is that practical general-purpose theorem provers and model-builders are available. However, a drawback is that there are linguistic phenomena that cannot be represented in the standard FOL; a typical example is a generalized quantifier such as *most* (Barwise and Cooper, 1981). Accordingly, it has been standard in formal semantics of natural language to use HOL as a representation language (Montague, 1974). Although HOL does not have general-purpose theorem provers, there is room for developing an automated reasoning system specialized for natural language inference. In general, a higher-order representation makes the logical structure of a sentence more explicit than a first-order encoding does and hence can simplify the process of proof search (Miller and Nadathur, 1986). Recently, based on the evaluation on the FraCaS dataset (Cooper et al., 1994), Mineshima et al. (2015) showed that a higher-order inference system outperformed the Boxer/Nutcracker’s first-order system (Bos, 2008) in both speed and accuracy. Likewise, Abzianidze (2015) developed a higher-order prover based on natural logic tableau system and showed that it achieved high accuracy comparable to state-of-the-art results on the SICK dataset (Marelli et al., 2014).

There are three main steps in our pipeline. The focus of this paper is on the last two components.

**1. Syntactic parsing** Input sentences are mapped onto CCG trees. We use a Japanese CCG parser Jigg (Noji and Miyao, 2016)<sup>1</sup>, a statistical parser based on Japanese CCGbank (Uematsu et al., 2015).

**2. Semantic parsing** CCG derivation trees are compositionally mapped onto semantic representations in HOL. The compositional mapping is implemented via simply typed  $\lambda$ -calculus in the standard way (Bos, 2008; Martínez-Gómez et al., 2016).

**3. Logical inference** Theorem proving in HOL is performed to check for entailment and contradiction. Axioms and proof-search procedures are largely language-independent, so we use the higher-order inference system of Mineshima et al. (2015)<sup>2</sup> and adapt it for our purpose.

<sup>1</sup><https://github.com/mynlp/jigg>

<sup>2</sup><https://github.com/mynlp/ccg2lambda>

Syntactic category	Semantic representation
$NP$	$\lambda NF.\exists x(N(\text{Base}, x) \wedge F(x))$
$S \setminus NP_{ga}$	$\lambda QK.Q(\lambda I.I, \lambda x.\exists v(K(\text{Base}, v) \wedge (\text{Nom}(v) = x)))$
$S \setminus NP_{ga} \setminus NP_o$	$\lambda Q_2 Q_1 K.Q_1(\lambda I.I, \lambda x_1.Q_2(\lambda I.I, \lambda x_2.\exists v(K(\text{Base}, v) \wedge (\text{Nom}(v) = x_1) \wedge (\text{Acc}(v) = x_2))))$
$S/S$	$\lambda SK.S(\lambda Jv.K(\lambda v'.(J(v') \wedge \text{Base}(v')), v))$
$NP/NP$	$\lambda QNF.Q(\lambda Gx.N(\lambda y.(\text{Base}(y) \wedge G(y)), x), F)$

**Table 1:** Examples of semantic templates. Base is the position in which the base form of a word appears.

### 3 Compositional Semantics and HOL

#### 3.1 CCG and semantic lexicon

Combinatory Categorical Grammar (CCG) (Steedman, 2000) is a lexicalized grammar formalism suitable for implementing a compositional mapping from syntax to semantics. A syntactic category of CCG is either a basic category such as  $S$  and  $NP$  or a functional category of the form  $X/Y$  or  $X \setminus Y$ . The meaning of a sentence is computed from a small number of combinatory rules and the meanings of constituent words. In addition to standard combinatory rules, the Japanese CCG parser uses a small number of unary type-shifting rules (e.g., the relativization rule that changes the category  $S \setminus NP$  to  $NP/NP$ ), to which suitable meaning composition rules are given.

We follow the standard method of building a semantic lexicon in CCG-based logical semantics (Bos, 2008). There are two kinds of lexical entries: (1) *semantic templates* that are schematic entries assigned to syntactic categories, possibly with syntactic features and (2) *lexical entries* directly assigned to a limited number of logical and functional expressions. Lexical entries can be sensitive to a POS tag, a surface form, and other information contained in the parser output. Table 1 shows semantic templates for main syntactic categories. More details will be provided in Section 3.2 and 3.3.

We use a language of standard higher-order logic (simple type theory) (Carpenter, 1997) as a representation language. Expressions in HOL are assigned semantic types. We use three basic types: E (Entity), Ev (Event), and Prop (Proposition). Thus, the semantic types of expressions in our system are defined by the rule

$$T ::= E \mid \text{Ev} \mid \text{Prop} \mid T_1 \Rightarrow T_2$$

where  $T_1 \Rightarrow T_2$  is a function type.

First-order language can be taken as a fragment of this system; apart from logical connectives and

$$\begin{aligned}
NP^\bullet &= ((E \Rightarrow \text{Prop}) \Rightarrow E \Rightarrow \text{Prop}) \Rightarrow (E \Rightarrow \text{Prop}) \Rightarrow \text{Prop} \\
S^\bullet &= ((E \Rightarrow \text{Prop}) \Rightarrow E \Rightarrow \text{Prop}) \Rightarrow \text{Prop} \\
(C1/C2)^\bullet &= (C1 \setminus C2)^\bullet = C2^\bullet \Rightarrow C1^\bullet
\end{aligned}$$

**Figure 1:** The mapping from syntactic categories to semantic types.  $\Rightarrow$  is right-associative.

quantifiers, all primitive expressions in first-order logic are confined to constant symbols of type  $E$  and predicates of type  $E \Rightarrow \text{Prop}$ ,  $E \Rightarrow E \Rightarrow \text{Prop}$ , and so on. Thus, adopting higher-order language does not lead to the loss of the expressive power of first-order language.

The Japanese CCG parser simplifies the standard CCG and uses two basic categories,  $S$  and  $NP$ . Accordingly, a mapping  $(\cdot)^\bullet$  from syntactic categories to semantic types can be defined as in Figure 1. Keeping the correspondence between syntactic categories and semantic types in the semantic lexicon guarantees that a well-formed formula is compositionally derived from the meaning assignment to each leaf of a CCG derivation tree.

### 3.2 Semantic composition for VPs

To model a semantics for VPs in Japanese, we adopt Neo-Davidsonian Event Semantics (Parsons, 1990; Jurafsky and Martin, 2009), which is widely used in the NLP field. For instance, the sentence (1) is analyzed as having the logical form in (2):

- (1) ジョンが ゆっくり 歩いた。  
 John NOM slowly walk PAST  
 ‘John walked slowly’

(2)  $\exists v(\text{walk}(v) \wedge (\text{Nom}(v) = \text{john}) \wedge \text{slow}(v) \wedge \text{Past}(v))$

In this approach, verbs are analyzed as 1-place predicates over events; arguments and adjuncts of VPs are also analyzed as event predicates. This semantic uniformity is suitable to handling Japanese syntactic structures in which the arguments of a VP is often implicit and thus the argument-adjunct distinction is less transparent than languages like English (Pietroski, 2005). As is seen in (2), we adopt the unique-role requirement for case markers (Carlson, 1984); for instance, the nominative case marker does not denote the relation  $\text{Nom}(v, x)$ , as in the event semantics in Boxer (Bos, 2008), but the function  $\text{Nom}(v) = x$ . This treatment allows us to make use of logical properties of equality and hence is more suited to theorem-proving in our setting.

To derive a semantic representation in event semantics compositionally, we adopt the compositional semantics of VPs in Champollion (2015) and analyze VPs themselves as introducing existential quantification over events. To derive the correct meaning for VP modifiers, the semantic type of a verb is raised so that the verb takes a modifier as argument but not vice versa. Figures 2 and 3 give example derivations.

VP modifiers such as *slowly* license an inference from *John walked slowly* to *John walked*, an inference correctly captured by the formula in (2). In English and Japanese, however, there are intensional VP modifiers that do not license this inference pattern. Thus, the sentence *John almost walked* does not entail *John walked* (Dowty, 1979). While it is not easy to provide a desirable analysis in first-order language (Hobbs, 1985), HOL gives a perspicuous representation:

(3)  $\exists v(\text{almost}(\text{walk}, v) \wedge (\text{Nom}(v) = \text{john}) \wedge \text{Past}(v))$

Here, *almost* is a higher-order predicate having the semantic type  $(E \Rightarrow \text{Prop}) \Rightarrow E \Rightarrow \text{Prop}$ . The meaning assignment to VP modifiers of category  $S/S$  in Table 1 is for extensional modifiers; an intensional modifier is assigned the representation  $\lambda SK.S(\lambda Jv.K(\text{Base}(J), v))$  in the lexical entry, which results in a representation as in (3).

### 3.3 Semantic composition for NPs

The quantificational structure of an NP plays a crucial role in capturing basic entailment patterns such as monotonicity inference. In the case of English, quantificational structures are specified by the type of determiners (e.g. *a*, *the*, *every*, *some*, *no*); together with the category distinction between  $N$  and  $NP$ , which is supported in English CCGbank (Hockenmaier and Steedman, 2007), one can provide a correct representation for NPs.

By contrast, Japanese is a classifier language, where NPs freely occur without determiners in argument position (Chierchia, 1998). For example, the subject in (4) appears in argument position without accompanying any determiner.

- (4) 小さな犬が 吠えた。  
 small dog NOM bark PAST  
 ‘A small dog barked’

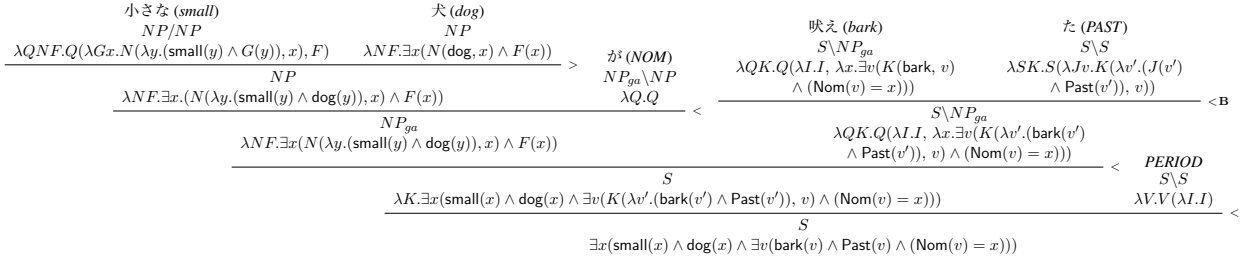


Figure 2: A CCG derivation tree for the sentence “A small dog barked”.

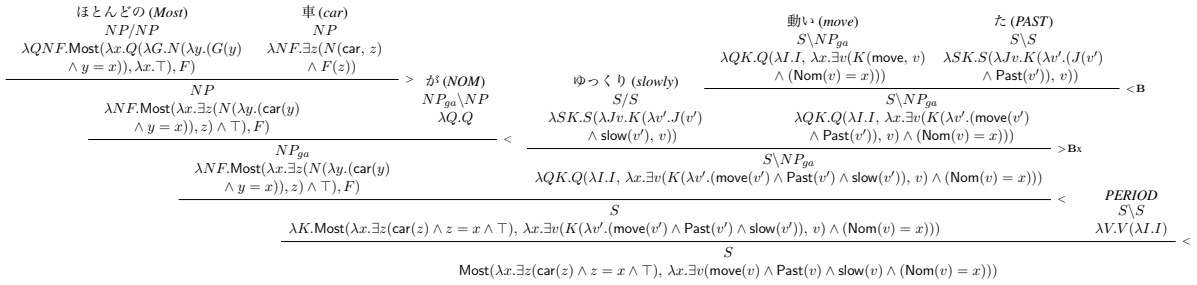


Figure 3: A CCG derivation tree for the sentence “Most cars moved slowly”.  $\top$  denotes the tautology.

Bekki (2010) provides a comprehensive CCG grammar for Japanese that adopts the  $N$ - $NP$  distinction and analyzes Japanese bare NPs as accompanying the null determiner. The Japanese CCGbank, by contrast, simplifies Bekki’s (2010) grammar and avoids the use of the null determiner; it does not use the category  $N$  and takes all NPs in Japanese to have the syntactic category  $NP$ . This discrepancy in NP-structure between English and Japanese poses a challenge to the standard approach to building compositional semantics.

To provide a compositional semantics adapted for the Japanese CCG, we take NPs themselves as introducing quantification over individuals, along the same lines as the semantics for VPs. The semantic type of NPs needs to be raised so that they take NP-modifiers as argument (cf. the template for  $NP$  in Table 1). Figure 2 shows a derivation for the sentence in (4), where the adjective *small* modifies the NP *dog* to form a bare NP *small dog*. It should be noted that the predicate  $small(x)$  is correctly inserted inside the scope of the existential quantification introduced by the NP *dog*. The so-called privative adjectives (e.g. *fake* and *former*) are analyzed in the same way as intensional VP modifiers.

Following the analysis in Mineshima et al. (2015), we analyze non-first-order generalized quantifier *most* as having the higher-order logical form

$Most(F, G)$ , where  $Most$  has the type of generalized quantifier  $(E \Rightarrow Prop) \Rightarrow (E \Rightarrow Prop) \Rightarrow Prop$ . Figure 3 shows an example derivation for a sentence containing a generalized quantifier *most*. Our system also handles floating quantifiers in Japanese.

## 4 Experiments

We evaluate our system<sup>3</sup> on Japanese Semantics test suite (JSeM)<sup>4</sup> (Kawazoe et al., 2015), a Japanese dataset for textual entailment designed in a similar way to the FraCaS dataset for English. These datasets focus on the types of logical inferences that do not require world knowledge. JSeM has Japanese translations of FraCaS problems and an extended set of problems focusing on Japanese syntax and semantics. Each problem has one or more premises, followed by a hypothesis. There are three types of answer: *yes* (entailment), *no* (contradiction), and *unknown* (neutral). Each problem is annotated with the types of inference (logical entailment, presupposition, etc.) and of linguistic phenomena.

We evaluate the system on 523 problems in the dataset. We focus on problems tagged with one of the five phenomena: generalized quantifier, plu-

<sup>3</sup>The system will be available at <https://github.com/mynlp/ccg2lambda>.

<sup>4</sup><http://researchmap.jp/community-inf/JSeM/?lang=english>

Section	#Problem	Gold	System	SLC
Quantifier	337	<b>92.3</b>	78.0	88.4
Plural	41	<b>68.3</b>	56.1	51.2
Adjective	65	<b>67.7</b>	63.1	44.6
Verb	36	<b>77.8</b>	75.0	55.5
Attitude	44	<b>88.6</b>	86.4	75.0
Total	523	<b>86.0</b>	75.0	76.7

**Table 2:** Accuracy on each section of JSeM.

	Acc.	Prec.	Rec.	Time
Gold parses	<b>86.0</b>	<b>94.9</b>	<b>81.3</b>	3.30s
w/o HOL axioms	69.8	93.3	56.5	2.47s
System parses	75.0	92.7	65.4	3.58s
SLC	76.7	77.5	79.3	n/a
Most common class (yes)	56.8	56.8	85.6	n/a

**Table 3:** Accuracy, precision, recall, and average proof time.

ral, adjective, verb, and attitude. We use problems whose inference type is logical entailment, excluding anaphora and presupposition. We use Kuromoji<sup>5</sup> for morphological analysis. To focus on the evaluation of semantic parsing and inference, we use gold syntactic parses, which show an upper bound on the performance of the semantic component. Gold syntactic parses are manually selected from  $n$ -best outputs of the CCG parser. For the higher-order inference system, we use the axioms presented in Mineshima et al. (2015) adapted with the necessary modification for our event semantics.

Given premises  $P_1, \dots, P_n$  and a hypothesis  $H$ , the system outputs *yes* ( $P_1 \wedge \dots \wedge P_n \rightarrow H$  is proved), *no* ( $P_1 \wedge \dots \wedge P_n \rightarrow \neg H$  is proved), or *unknown* (neither is proved in a fixed proof-search space).<sup>6</sup> We set a 30 seconds timeout for each inference run; the system outputs *unknown* after it. The current semantic lexicon has 36 templates and 113 lexical entries.

Table 2 and 3 show the results. The system with gold syntactic parses achieved 86% accuracy on the total 523 problems, with high precision and reasonable speed. There was no timeout.<sup>7</sup> The accuracy dropped to 70% when ablating HOL axioms (Table 3). SLC refers to the performance of a supervised learning classifier<sup>8</sup> based on 5-fold cross-validation for each section. Although direct comparison is not

<sup>5</sup><http://www.atilika.org/>

<sup>6</sup>Note that natural-logic-based systems (MacCartney and Manning, 2008) do not handle multi-premised problems.

<sup>7</sup>Our pipeline was run single-threaded on Ubuntu Linux 64 bits with a CPU at 2.67GHz.

<sup>8</sup>We used NTCIR RITE baseline tools (<http://www.cl.ecei.tohoku.ac.jp/rite2/doku.php>).

Section	#Problem	Gold	System	M15
Quantifier	335	92.5	78.2	78.4
Plural	38	65.8	52.6	66.7
Adjective	21	57.1	47.6	68.2
Verb	9	66.7	66.7	62.5
Attitude	14	78.6	78.6	76.9
Total	417	87.3	74.1	73.3

**Table 4:** The results on a subset of JSeM that is a translation of FraCaS. M15 refers to the accuracy of Mineshima et al. (2015) on the corresponding sections of FraCaS.

possible, our system with gold parses outperforms it for all sections.

Out of the 523 problems, 417 are Japanese translations of the FraCaS problems. Table 4 shows a comparison between the performance of our system on this subset of the JSeM problems and the performance of the RTE system for English in Mineshima et al. (2015) on the corresponding problems in the FraCaS dataset. Mineshima et al. (2015) used system parses of the English C&C parser (Clark and Curran, 2007). The total accuracy of our system is comparable to that of Mineshima et al. (2015).

Most errors we found are due to syntactic parse errors caused by the CCG parser, where no correct syntactic parses were found in  $n$ -best responses. Comparison between gold parses and system parses shows that correct syntactic disambiguation improves performance.

## 5 Conclusion

To our knowledge, this study provides the first semantic parsing system based on CCG that compositionally maps real texts in Japanese onto logical forms. We have also demonstrated the capacity of HOL for textual entailment. The evaluation on JSeM showed that our system performs efficient logical inference on various semantic phenomena, including those that challenge the standard FOL. The attractiveness of a logic-based system is that it is highly modular and can be extended with other components such as a robust knowledge base (Lewis and Steedman, 2013; Beltagy et al., 2013; Bjerva et al., 2014). Such an extension will be a focus of future work.

**Acknowledgments** We are grateful to the three anonymous reviewers for their helpful comments and suggestions. This research has been supported by the JST CREST program.

## References

- Lasha Abzianidze. 2015. A tableau prover for natural logic and language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2492–2502.
- Jon Barwise and Robin Cooper. 1981. Generalized quantifiers and natural language. *Linguistics and Philosophy*, 4(2):159–219.
- Daisuke Bekki. 2010. *A Formal Theory of Japanese Grammar: The Conjugation System, Syntactic Structures, and Semantic Composition*. Kuroshio. (In Japanese).
- Islam Beltagy, Cuong Chau, Gemma Boleda, Dan Garrette, Katrin Erk, and Raymond Mooney. 2013. Montague meets Markov: Deep semantics with probabilistic logical form. In *2nd Joint Conference on Lexical and Computational Semantics: Proceeding of the Main Conference and the Shared Task*, pages 11–21.
- Johannes Bjerva, Johan Bos, Rob van der Goot, and Malvina Nissim. 2014. The Meaning Factory: Formal semantics for recognizing textual entailment and determining semantic similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 642–646.
- Johan Bos and Katja Markert. 2005. Recognising textual entailment with logical inference. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 628–635.
- Johan Bos. 2008. Wide-coverage semantic analysis with Boxer. In *Proceedings of the 2008 Conference on Semantics in Text Processing*, pages 277–286.
- Johan Bos. 2015. Open-domain semantic parsing with Boxer. In *Proceedings of the 20th Nordic Conference of Computational Linguistics*, pages 301–304.
- Greg Carlson. 1984. Thematic roles and their role in semantic interpretation. *Linguistics*, 22(3):259–280.
- Bob Carpenter. 1997. *Type-Logical Semantics*. MIT press.
- Lucas Champollion. 2015. The interaction of compositional semantics and event semantics. *Linguistics and Philosophy*, 38(1):31–66.
- Gennaro Chierchia. 1998. Reference to kinds across language. *Natural Language Semantics*, 6(4):339–405.
- Stephen Clark and James R. Curran. 2007. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33(4):493–552.
- Robin Cooper, Richard Crouch, Jan van Eijck, Chris Fox, Josef van Genabith, Jan Jaspers, Hans Kamp, Manfred Pinkal, Massimo Poesio, Stephen Pulman, et al. 1994. FraCaS — a framework for computational semantics. *Deliverable*, D16.
- David Dowty. 1979. *Word Meaning and Montague Grammar*. Springer.
- Yuta Hayashibe, Mamoru Komachi, and Yuji Matsumoto. 2011. Japanese predicate argument structure analysis exploiting argument position and type. In *Proceedings of IJCNLP 2011*, pages 201–209.
- Jerry R. Hobbs. 1985. Ontological promiscuity. In *Proceedings of the 23rd annual meeting on Association for Computational Linguistics*, pages 60–69.
- Julia Hockenmaier and Mark Steedman. 2007. CCGbank: a corpus of CCG derivations and dependency structures extracted from the Penn Treebank. *Computational Linguistics*, 33(3):355–396.
- Daniel Jurafsky and James H. Martin. 2009. *Speech and Language Processing*. Prentice-Hall, Inc.
- Daisuke Kawahara and Sadao Kurohashi. 2011. Generative modeling of coordination by factoring parallelism and selectional preferences. In *Proceedings of IJCNLP 2011*, pages 456–464.
- Ai Kawazoe, Ribeka Tanaka, Koji Mineshima, and Daisuke Bekki. 2015. An inference problem set for evaluating semantic theories and semantic processing systems for Japanese. In *Proceedings of LENS12*, pages 67–73.
- Taku Kudo and Yuji Matsumoto. 2002. Japanese dependency analysis using cascaded chunking. In *Proceedings of the 6th Conference on Natural Language Learning*, pages 63–69.
- Mike Lewis and Mark Steedman. 2013. Combining distributional and logical semantics. *Transactions of the Association for Computational Linguistics*, 1:179–192.
- Bill MacCartney and Christopher D. Manning. 2007. Natural logic for textual inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200.
- Bill MacCartney and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 521–528.
- Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of LREC2014*, pages 216–223.
- Pascual Martínez-Gómez, Koji Mineshima, Yusuke Miyao, and Daisuke Bekki. 2016. ccg2lambda: a compositional semantics system. In *Proceedings of ACL 2016 System Demonstrations*, pages 85–90.
- Dale Miller and Gopalan Nadathur. 1986. Some uses of higher-order logic in computational linguistics. In *Proceedings of the 24th annual meeting on Association for Computational Linguistics*, pages 247–256.

- Koji Mineshima, Pascual Martínez-Gómez, Yusuke Miyao, and Daisuke Bekki. 2015. Higher-order logical inference with compositional semantics. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2055–2061.
- Richard Montague. 1974. *Formal Philosophy: Selected Papers*. Yale University Press New Haven.
- Hiroshi Noji and Yusuke Miyao. 2016. Jigg: a framework for an easy natural language processing pipeline. In *Proceedings of ACL 2016 System Demonstrations*, pages 103–108.
- Terence Parsons. 1990. *Events in the Semantics of English*. MIT Press.
- Paul Pietroski. 2005. *Events and Semantic Architecture*. Oxford University Press.
- Mark Steedman. 2000. *The Syntactic Process*. MIT Press.
- Sumire Uematsu, Takuya Matsuzaki, Hiroki Hanaoka, Yusuke Miyao, and Hideki Mima. 2015. Integrating multiple dependency corpora for inducing wide-coverage Japanese CCG resources. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 14(1):1–24.