Inquisitive meaning with DTS(need reconsideration)

Kazuki Watanabe, Koji Mineshima, Daisuke Bekki
University of Tokyo
Ochanomizu University
watanabe-kazuki163@g.ecc.u-tokyo.ac.jp
mineshima.koji@ocha.ac.jp
bekki@is.ocha.ac.jp

1 Introduction

A number of studies have analyzed interrogative sentences. Recently, Inquisitive semantics (Ciardelli et al. (2012)) was proposed and some studies (Ciardelli et al. (2017); Champollion et al. (2015)) based on this framework have been reported. Inquisitive semantics analyzes declarative sentences and interrogative sentences in the same way. However, this framework is based on model theory. Therefore, it is not trivial that how to apply this framework to computational semantics. On the other hand, there are some frameworks (Ranta (1994); Ginzburg (2005)) based on dependent type theory (Martin-Löf (1984)). These frameworks can deal with not only declarative sentences but also some interrogative sentences. However, these frameworks don't meet the principle of compositionality and use some meta rules just for describing the relation between questions and answers. we will explain only inquisitive semantics in more detailed.

In this study, we show how to express the semantic representations of interrogative sentences in the Dependent Type Semantics (DTS; Bekki and Mineshima (2017)) and we modify some semantic representations of declarative sentences. DTS is based on dependent type theory and provides the unified analysis of semantic representations of declarative sentences, presupposition and anaphora. In this paper, we propose the extension of DTS to express meanings of interrogative sentences, which also meets the principle of compositionality and uses no meta rules.

The framework that we propose in this paper is different from these points below.

1. The semantics is based on proof theory, while inquisitive semantics is based on model theory.

2. The compositionality principle is achieved without meta rules, while other frameworks using dependent type theory are not.

In section 2, we will explain some features of inquisitive semantics which are related to our study. In section 3 we explain the semantics of DTS. The main part of this paper is section 4. In this section, we propose original semantic representations for some interrogative sentences with DTS and we succeed at extending DTS. Moreover, this extension retains the structure of presupposition and anaphora in DTS.

2 Inquisitive semantics

Inquisitive semantics (Ciardelli et al. (2012)) is based on model theory and uses the first-order system as InqB. InqB is a logic and equivalent to intuitionistic logic, which enables InqB to distinguish meanings of declarative sentences from one of interrogative sentences. Propositions of declarative sentences must be negated twice as the following diagram.

(1) John is tall or short.

(2) $$\neg\neg(\mathbf{tall}(j) \vee \mathbf{short}(j))$$

On the other hand, propositions of interrogative sentences are expressed in the following way. For instance, (4) is a semantic representation of (3) when (3) is considered as a alternative question, not as a polar question.

(3) Is John tall or short?

(4) $$\mathbf{tall}(j) \vee \mathbf{short}(j)$$

Double negation is a key to distinguish declarative sentences from interrogative sentences in InqB, but double negation causes problems in the DTS (Bekki and Mineshima (2017)) when we handle presupposition and anaphora. we will discuss this problem in section 3, section 4.

Another feature of inquisitive semantics is entailment. Inquisitive semantics defines entailment for not only declarative sentences but also interrogative sentences in the same way. In inquisitive semantics, some interrogative sentences entail another interrogative ones and some declarative sentences do another interrogative ones. Some examples (Ciardelli et al. (2012)) are shown as follows.

(5) What is the number of planets? $\Rightarrow$ Is the number of planets even?

(6) John is tall. $\Rightarrow$ Is John tall?

3 DTS

DTS (Bekki and Mineshima (2017)) is a discourse semantic framework based on dependent type theory (Martin-Löf (1984)). The following three type constructors

$$\dfrac{A : type_i \quad \begin{array}{c}\overline{x : A}^{\,k} \\ \vdots \\ B : type_j\end{array}}{(x : A) \to B : type_{max(i,j)}}\ (\Pi F), k \qquad \dfrac{A : type_i \quad \begin{array}{c}\overline{x : A}^{\,k} \\ \vdots \\ M : B\end{array}}{\lambda x.M : (x : A) \to B}\ (\Pi I), k \qquad \dfrac{M : (x : A) \to B \quad N : A}{MN : B[N/x]}\ (\Pi E)$$

$$\dfrac{A : type_i \quad \begin{array}{c}\overline{x : A}^{\,k} \\ \vdots \\ B : type_j\end{array}}{\begin{bmatrix} x : A \\ B \end{bmatrix} : type_{max(i,j)}}\ (\Sigma F), k \qquad \dfrac{M : A \quad N : B[M/x]}{(M, N) : \begin{bmatrix} x : A \\ B \end{bmatrix}}\ (\Sigma I) \qquad \dfrac{M : \begin{bmatrix} x : A \\ B \end{bmatrix}}{\pi_1(M) : A}\ (\Sigma E) \qquad \dfrac{M : \begin{bmatrix} x : A \\ B \end{bmatrix}}{\pi_2(M) : B[\pi_1(M)/x]}\ (\Sigma E)$$

$$\dfrac{A : type_i \quad A : type_j}{A \uplus B : type_{max(i,j)}}\ (\uplus F) \qquad \dfrac{M : A}{\iota_1(M) : A \uplus B}\ (\uplus I) \qquad \dfrac{M : A}{\iota_2(M) : A \uplus B}\ (\uplus I)$$

$$\dfrac{L : A \uplus B \quad C : (A \uplus B) \to type_i \quad \begin{array}{c}\overline{x : A}^{\,k} \\ \vdots \\ M : C(\iota_1(x))\end{array} \quad \begin{array}{c}\overline{x : B}^{\,k} \\ \vdots \\ N : C(\iota_2(x))\end{array}}{case\ L\ of\ (\lambda x.M; \lambda x.N) : C(L)}\ (\uplus E), k$$

Figure 1: Inference rules (fix layout)

are proposed in these studies (Bekki and Mineshima (2017), Ranta (1994)) .

(7) Π-Type: $(x : A) \to B(x)$

(8) Σ-Type: $\begin{bmatrix} x : A \\ B(x) \end{bmatrix}$

(9) $\biguplus$-Type: $A \biguplus B$

Π-Type can be considered as a generalized function type and Σ-Type can be also considered as a generalized product type. In dependent type theory, these two types depend on terms and this property makes a difference from simple type theory. $\biguplus$-Type can be considered as a generalized disjoint union type. $\biguplus$-Type don't depend on term, unlike Π-Type and Σ-Type. Figure 1 shows some inference rules which we must use to express the inquisitive meaning in DTS.

While other frameworks (Ranta (1994); Ginzburg (2005)) based on dependent type theory don't meet the principle of compositionality, DTS meets this principle with the syntax of CCG (Steedman (1996)). In section 4, we will show the way to compose the meanings of some sentences.

One of the another significant features of DTS is that presupposition and anaphora are represented by underspecified terms $@_i$. DTS also considers presupposition and anaphora resolution as proof search of underspecified term $@_i$.

$$\dfrac{A : type_i \quad \begin{matrix}\overline{x : A}^{\,k} \\ \vdots \\ B : type_j\end{matrix}}{(x\colon A) \bigoplus B : type_{max(i,j)}} \,(\bigoplus F), k \qquad \dfrac{t : A \quad u : B[x/t]}{[t,u] : (x\colon A) \bigoplus B} \,(\bigoplus I) \qquad \dfrac{[t,u] : (x\colon A) \bigoplus B \quad \begin{matrix}\overline{x : A}^{\,k} \quad \overline{y : B(x)}^{\,k} \\ \vdots \\ m : C\end{matrix}}{case_{[t,u]} m : C} \,(\bigoplus E), k$$

Figure 2: Inference rules of $\bigoplus$-Type. Elimination rule can be applied if $m : C$ and open assumptions which depend on $m : C$ don't have x, y as free variables.(fix layout)

4 Inquisitive meaning with DTS

Semantic representations of interrogative sentences such as polar question, alternative question and wh question can't be represented in former DTS. In this paper, we introduce existential type (Luo (1994)) $\bigoplus$-Type in DTS to express the meaning of wh question. $\bigoplus$-Type is also called as weak-sigma type and inference rules of $\bigoplus$-Type is shown in Figure 2.

The semantic representations of polar question and alternative question are represented by using $\uplus$-Type. In previous study using DTS, $\uplus$-Type was used for expressing the semantic representations of declarative-or sentences (fix this expression). However, we decide to use $\uplus$-Type for polar question and alternative question and modify the semantic representations of declarative-or sentences (fix this expression). (10) is a simple alternative question and semantic representation is (11).

(10) Does Mary raise a horse or$_{alt}$ a pony?

(11) $$\begin{bmatrix} u : \begin{bmatrix} x : entity \\ \mathbf{horse}(x) \end{bmatrix} \\ \mathbf{raise}(m, \pi_1 u) \end{bmatrix} \uplus \begin{bmatrix} u : \begin{bmatrix} x : entity \\ \mathbf{pony}(x) \end{bmatrix} \\ \mathbf{raise}(m, \pi_1 u) \end{bmatrix}$$

In inquisitive semantics, declarative-or sentences (fix this expression) is distinguished with alternative question by double negation. However,

(12) Mary raises a horse or$_{dcl}$ a pony.

(13) $$\begin{aligned} &(\neg \begin{bmatrix} u : \begin{bmatrix} x : entity \\ \mathbf{horse}(x) \end{bmatrix} \\ \mathbf{raise}(m, \pi_1 u) \end{bmatrix} \to \begin{bmatrix} u : \begin{bmatrix} x : entity \\ \mathbf{pony}(x) \end{bmatrix} \\ \mathbf{raise}(m, \pi_1 u) \end{bmatrix}) \\ &\wedge (\neg \begin{bmatrix} u : \begin{bmatrix} x : entity \\ \mathbf{pony}(x) \end{bmatrix} \\ \mathbf{raise}(m, \pi_1 u) \end{bmatrix} \to \begin{bmatrix} u : \begin{bmatrix} x : entity \\ \mathbf{horse}(x) \end{bmatrix} \\ \mathbf{raise}(m, \pi_1 u) \end{bmatrix}) \\ &\wedge (\begin{bmatrix} x : entity \\ \mathbf{pony}(x) \end{bmatrix} \uplus \begin{bmatrix} x : entity \\ \mathbf{horse}(x) \end{bmatrix}) \end{aligned}$$

....

5 Conclusion

....

References

Bekki, D. and K. Mineshima (2017). Context-passing and underspecification in Dependent Type Semantics. In *Modern Perspectives in Type Theoretical Semantics*, pp. 11–41. Springer.

Champollion, L., I. Ciardelli, and F. Roelofsen (2015). Some questions in typed inquisitive semantics. In *Talk at workshop on Questions in Logic and Semantics, Amsterdam, handout available at illc. uva. nl/inquisitivesemantics/workshops/inqturn-closing-event-2015*.

Ciardelli, I., J. Groenendijk, and F. Roelofsen (2012). Inquisitive semantics. *NASSLLI lecture notes, available via www. illc. uva. nl/inquisitivesemantics*.

Ciardelli, I., F. Roelofsen, and N. Theiler (2017, Feb). Composing alternatives. *Linguistics and Philosophy 40*(1), 1–36.

Ginzburg, J. (2005, 04). Abstraction and ontology: Questions as propositional abstracts in type theory with records. *J. Log. Comput. 15*, 113–130.

Luo, Z. (1994). *Computation and Reasoning: A Type Theory for Computer Science*. New York, NY, USA: Oxford University Press, Inc.

Martin-Löf (1984). *Intuitionistic type theory*, Volume 9. Bibliopolis Naples.

Ranta, A. (1994). *Type-theoretical grammar*. Oxford University Press.

Steedman, M. (1996). *Surface structure and interpretation*. Number 30 in Linguistic inquiry monographs / Samuel Jay Keyser, general editor. MIT Press.