Modal Subordination in Dependent Type Semantics

Aoi Iimura Teruyuki Mizuno Daisuke Bekki
Ochanomizu University
{iimura.aoi, bekki}@is.ocha.ac.jp
mizuno.teruyuki@ocha.ac.jp

Abstract

In the field of natural language processing, the construction of “linguistic pipelines”, which draw on insights from theoretical linguistics, stands in a complementary relationship to the prevailing paradigm of large language models. The rapid development of these pipelines has been fueled by recent advancements, including the emergence of Dependent Type Semantics (DTS) — a type-theoretic framework for natural language semantics. While DTS has been successfully applied to analyze complex linguistic phenomena such as anaphora and presupposition, its capability to account for modal expressions remains an underexplored area. This study aims to address this gap by proposing a framework that extends DTS with modal types. This extension broadens the scope of linguistic phenomena that DTS can account for, including an analysis of modal subordination, where anaphora interacts with modal expressions.

1 Introduction

In recent computational linguistics research, a new approach to natural language processing has seem rapid progress: the use of *linguistic pipelines* (Abzianidze, 2015; Mineshima et al., 2015). These pipelines combine theoretical linguistic insights with computational methods. A key driver of this progress is Dependent Type Semantics (DTS) (Bekki and Mineshima, 2017), a framework for natural language semantics that is rooted in Dependent Type Theory (DTT) (Martin-Löf, 1984). Drawing upon the rich tradition of type theory in programming semantics, DTS provides a compositional framework for the analysis of anaphora and presupposition, which exploits theorem provers in analyzing both anaphora resolution and general inference. By a systematic mapping from formal syntax to semantic interpretation, DTS bridges a significant gap between linguistic theories and computational implementation.

In DTS, the semantic representation (SR) of a sentence corresponds to a type in DTT. The dependency of a type on terms allows reference to terms constructed from the context, thereby reducing both anaphora resolution and presupposition binding to problems of proof search. While DTS provides compelling analyses of complex linguistic phenomena, empirical research on modal expressions remains largely unexplored (but see Tanaka et al. 2015), with existing studies primarily focusing on propositions that abstract away from modal expressions. This study aims to extend DTS by providing an analysis of phenomena involving modal expressions.

Modal expressions, which pertain to the notions of possibility and necessity, have been a central research topic in formal semantics. One of the most discussed phenomena is modal subordination (MS), which, since the pioneering work by Roberts (1989), has been investigated by many researchers (Frank and Kamp, 1997; Kaufmann, 2000; van Rooij, 2005; Asher and McCready, 2007; Keshet and Abney, 2024). (1) and (2) illustrate MS.

(1) [A wolf]$_i$ might come in. It$_i$ would growl.

(2) [A wolf]$_i$ might come in. #It$_i$ growls.

As illustrated in (1), an indefinite introduced within the scope of *might* brings a “hypothetical entity” into the discourse[1]. To anaphorically refer to this entity, the subsequent discourse must align with the hypothetical scenario in which the entity is assumed to exist, which is typically signaled by the use of *would* in English. The absence of *would*, as demonstrated in (2), blocks this alignment, thereby preventing the pronoun from referring to the indefinite and resulting in a failure of MS.

[1]Here, we focus on the analysis of the *de dicto* reading. While example (1) also allows a *de re* reading, where *a wolf* scopes over *might*, a detailed analysis of this reading within DTS is beyond the scope of this paper.

Proceedings of the Second Workshop on the Bridges and Gaps between Formal and Computational Linguistics (BriGap-2), pages 15–19
September 24, 2025,

2 Dependent Type Semantics

DTS is a framework developed within the propositions-as-types paradigm. In DTS, the notion of existential quantification $\exists x \in A.B$ is represented by the *dependent product types* $(x : A) \times B$, which are types of pairs (a, b) such that a is of type A and b is of type $B(a)$. The SR of the unmodalized sentence in (3a) is given in (3b). We employ vertical notation for the dependent product type in the subsequent discussion, and π_1 denotes the proof constructor that yields the first projection of such a pair.

(3) a. A wolf came in.

b. $$\left[\begin{array}{l} u : \left[\begin{array}{l} x : \textbf{entity} \\ \textbf{wolf}(x) \end{array}\right] \\ \textbf{comeIn}(\pi_1(u)) \end{array}\right]$$

Bekki (2023) analyzes pronouns as introducing *underspecified types*, written as $(x@A) \times B$. Here, the variable x functions as a placeholder that is to be replaced by a proof of type A from a given context. Example (4) briefly illustrates how anaphora resolution proceeds in DTS.

(4) [A wolf]$_i$ came in. It$_i$ growled.

a. $$\left[\begin{array}{l} v : \left[\begin{array}{l} u : \left[\begin{array}{l} x : \textbf{entity} \\ \textbf{wolf}(x) \end{array}\right] \\ \textbf{comeIn}(\pi_1(u)) \end{array}\right] \\ \left[\begin{array}{l} w @ \left[\begin{array}{l} z : \textbf{entity} \\ \neg\, \textbf{human}(z) \end{array}\right] \\ \textbf{growl}(\pi_1(w)) \end{array}\right] \end{array}\right]$$

b. $$\left[\begin{array}{l} v : \left[\begin{array}{l} u : \left[\begin{array}{l} x : \textbf{entity} \\ \textbf{wolf}(x) \end{array}\right] \\ \textbf{comeIn}(\pi_1(u)) \end{array}\right] \\ \textbf{growl}(\pi_1\pi_1(v)) \end{array}\right]$$

The underspecified type in (4a) is eliminated through *type-checking*, a process that validates whether an SR is a well-formed type under a given context. Upon encountering $w@((z : \textbf{entity}) \times \neg\textbf{human}(z))$, the type-checking algorithm attempts to find a proof of type $(z : \textbf{entity}) \times \neg\textbf{human}(z)$. In this specific case, such a proof is successfully found and substituted for the variable x. Subsequently, $\pi_1\pi_1(v)$, which corresponds to the entity x (i.e., the first element of $\pi_1(v)$), serves as the argument of the predicate **growl**, thereby resolving the pronoun *it*.

3 Modal DTS

To account for modal expressions within DTS, we propose Modal DTS, an extension grounded in Contextual Modal Type Theory (CMTT) (Nanevski et al., 2008). Modal DTS introduces two novel type constructors: $[\Psi]$ for necessity and $\langle\Psi\rangle$ for possibility, both of which are parameterized by a context Ψ. In a manner analogous to possible worlds semantics, Ψ serves as a proxy for a domain of possible worlds; accordingly, $[\Psi]$ and $\langle\Psi\rangle$ indicate that the propositions within their scope hold in all or some worlds, respectively, where Ψ is true. As an example, Figure 1 illustrates the SR of (1).

$$\left[\begin{array}{l} v \colon \langle\Psi\rangle \left[\begin{array}{l} u : \left[\begin{array}{l} x : \textbf{entity} \\ \textbf{wolf}(x) \end{array}\right] \\ \textbf{comeIn}(\pi_1(u)) \end{array}\right] \\ [\Psi] \left[\begin{array}{l} w @ \left[\begin{array}{l} z : \textbf{entity} \\ \neg\, \textbf{human}(z) \end{array}\right] \\ \textbf{growl}(\pi_1(w)) \end{array}\right] \end{array}\right]$$

Figure 1: SR of (1) before anaphora resolution

As described in § 2, a dependent product type is a type of pairs where the second conjunct depends on the first element of the pair, i.e., the second conjunct is within the scope of the dependent product type. Accordingly, in Figure 1, where the SR of the first sentence forms the first conjunct, and that of the second sentence the second conjunct, the continuation *it would growl* quantifies over the subset of possible worlds in which *a wolf came in*.

3.1 Contextual Modal Type Theory

Intuitionistic modal logic for necessity is founded on the judgmental notion of categorical truth. Nanevski et al. (2008) examined the consequences of relativizing these notions of categorical truth to explicitly specified contexts, resulting in the formulation of contextual modal logic and its type-theoretic counterpart. Nanevski et al. (2008) advanced the structural approach to intuitionistic modal logic by allowing arbitrary contexts to be internalized within propositions. From a type-theoretic standpoint, CMTT is based on contextual modal logic and provides formal definitions for proof term assignment, substitution on terms, proof reductions and expansions, as well as strong normalization. From a logical standpoint, CMTT constitutes a relativized variant of the intuitionistic modal logic S4.

Modal DTS is a framework that uniquely integrates the dependent types of DTS with the modal types of CMTT. The newly introduced types are grounded in the notions of *contextual necessity* and *contextual possibility* as defined in CMTT. Contextual necessity, denoted as $[\Psi]A$, indicates that

$$
\dfrac{\dfrac{v : \langle\Psi\rangle \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \qquad \dfrac{\dfrac{\sigma' : \Psi \quad \overline{\sigma : \Psi}^{\,1}}{\sigma' : \Psi}\,(w) \qquad \overline{v' : \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix}}^{\,1}}{\langle \sigma', v' \rangle : \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \ \mathsf{poss}\ \langle \sigma' : \Psi \rangle}\,(\mathsf{poss}\ I)}{\texttt{letdia}\ (\, v, \langle \sigma, v' \rangle . \langle \sigma', v' \rangle \,) : \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \ \mathsf{poss}\ \langle \sigma' : \Psi \rangle}\,(\Diamond E),1}{\texttt{p2}\ (\,\texttt{letdia}\ (\, v, \langle \sigma, v' \rangle . \langle \sigma', v' \rangle \,)\,)\ (\equiv D) : \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix}}\,(\mathsf{poss}\ E)
$$

$$
\vdots
$$

$$
(\, \pi_1\pi_1(D), f(\pi_1(D)) \,) : \begin{bmatrix} z : \textbf{entity} \\ \neg\ \textbf{human}(z) \end{bmatrix}
$$

Figure 2: A part of the proof search for (1) A wolf might come in. It would growl.

A is true in all worlds in Ψ. Contextual possibility introduces a new judgment, $A\ \mathsf{poss}\langle\Psi\rangle$, which expresses existential quantification over possible worlds: the judgment $A\ \mathsf{poss}\langle\Psi\rangle$ holds if there exists a world in which both Ψ and A are simultaneously true. This judgment is internalized via the type operator $\langle\Psi\rangle A$.

3.2 New rules added to DTS

DTT defines types using three rules: the formation rule, the introduction rule, and the elimination rule. The formation rule specifies the conditions under which a type is well-formed; the introduction rule describes how a type can be constructed; and the elimination rule explains how a type can be used.

For the modal types $[\Psi]A$ and $\langle\Psi\rangle A$, we define their formation rules independently, while adopting their introduction and elimination rules from CMTT. In the case of $A\ \mathsf{poss}\langle\Psi\rangle$, we define its formation and elimination rules independently, and adopt its introduction rule from CMTT [2].

4 Analysis

Our account treats anaphora resolution as the specification of an underspecified type by proof construction. The issue of anaphora accessibility in MS, as observed in examples (1) and (2), is thereby reduced to the question of whether the placeholder can be filled by replacing the variables of underspecified types with concrete proof terms during type-checking.

4.1 Sentence to be modally subordinated

As an example of successful proof construction, we consider (1). Figure 2 illustrates how the variable w associated with the underspecified type in Figure 1 is removed by a proof of type $(z : \textbf{entity}) \times \neg\textbf{human}(z)$. Here, `letdia` denotes the proof term for the elimination rule of $\langle\Psi\rangle$, and `p2` denotes that for the elimination rule of poss.

The strategy for constructing the proof is to first obtain D (`p2` (`letdia` ($v, \langle \sigma, v' \rangle . \langle \sigma', v' \rangle$))) in Figure 2, which is a proof of the SR of the first sentence with the modal expression eliminated, i.e., $(u : ((x : \textbf{entity}) \times \textbf{wolf}(x))) \times \textbf{comeIn}(\pi_1(u))$. The proof search then proceeds as a standard procedure in DTT without involving modal expressions. Consequently, as illustrated by the dotted line in Figure 2, it is sufficient to find a proof of type $(z : \textbf{entity}) \times \neg\textbf{human}(z)$ using only the existing rules for dependent types.

Let us follow the proof diagram in Figure 2 step by step. Since the SR of the second sentence can depend on the proof of the first, the SR of the first sentence can be used in the proof of the second. To utilize the SR of the first sentence within the scope of possibility, we apply the elimination rule for $\langle\Psi\rangle$ (= $\Diamond E$). This application requires that the SR of the sentence be introduced within a poss environment, as derived from CMTT. Therefore, we apply the introduction rule for poss (= $\mathsf{poss}\ I$) to introduce the poss environment, using the context $\sigma' : \Psi$ provided by the formation rule for necessity associated with the second sentence's modal expression $[\Psi]$. Finally, the elimination rule for poss (= $\mathsf{poss}\ E$) is applied to eliminate the poss environment, and we seek the proof D of the SR of the first sentence, stripped of its modal expression.

After returning to a general proof search without modal expressions, we search for a proof of type $(z : \textbf{entity}) \times \neg\textbf{human}(z)$, using the information "if something is a wolf, then it is not human" as a

[2] For details on the new rules added to DTS, please refer to https://github.com/iimuraaoi/modal_dts.git

$$
\dfrac{v : \langle \Psi \rangle \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \qquad \dfrac{\overline{\sigma : \Psi}^{\,1} \qquad \overline{v' : \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix}}^{\,1}}{\langle \sigma, v' \rangle : \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \ \mathsf{poss}\ \langle \sigma : \Psi \rangle}\ (\mathsf{poss}\ I)}{\texttt{letdia}\ (\, v, \langle \sigma, v' \rangle . \langle \sigma, v' \rangle\,) : \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \ \mathsf{poss}\ \langle \sigma : \Psi \rangle}\ (\Diamond E), 1
$$

Figure 3: A part of the proof search for (2) A wolf might come in. #It growls.

global function f in the context. As a result, the variable w associated with the underspecified type in Figure 1 is removed, yielding Figure 4.

$$
\begin{bmatrix} v : \langle \Psi \rangle \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \\ [\Psi]\ \ \textbf{growl}(\pi_1 \pi_1(D)) \end{bmatrix}
$$

Figure 4: SR of (1) after anaphora resolution

4.2 Sentence not to be modally subordinated

On the other hand, the anaphora in (2) is unacceptable. Modal DTS analyzes anaphora accessibility, as DTS does, in terms of proof constructability.

$$
\begin{bmatrix} v : \langle \Psi \rangle \begin{bmatrix} u : \begin{bmatrix} x : \textbf{entity} \\ \textbf{wolf}(x) \end{bmatrix} \\ \textbf{comeIn}(\pi_1(u)) \end{bmatrix} \\ \begin{bmatrix} w @ \begin{bmatrix} z : \textbf{entity} \\ \neg\ \textbf{human}(z) \end{bmatrix} \\ \textbf{growl}(\pi_1(w)) \end{bmatrix} \end{bmatrix}
$$

Figure 5: SR of (2) before anaphora resolution

The reason why proof construction is blocked in the anaphora resolution of (2) lies in the rule (poss E), which is defined independently in Modal DTS. The application of (poss E) imposes a restriction: the proof term corresponding to the context used to introduce the poss environment must appear as a free variable in the overall proof term prior to the application of (poss E). In Figure 2, for example, the application is permitted because this condition is satisfied.

$$\sigma' \in \texttt{letdia}\ (\, v, \langle \sigma, v' \rangle . \langle \sigma', v' \rangle\,)$$

Stated differently, this constraint effectively requires that if the antecedent sentence contains a modal expression, then the consequent sentence must also contain a modal expression. In Figure 3, which illustrates the proof search for (2), the only candidate for a modal expression introduced into the poss environment is the antecedent *might*, which fails to satisfy the condition necessary for eliminating the poss environment.

$$\sigma \notin \texttt{letdia}\ (\, v, \langle \sigma, v' \rangle . \langle \sigma, v' \rangle\,)$$

In other words, the underspecified types in Figure 5 cannot be removed during the type-checking process, which accounts for the unacceptability of the sentence.

5 Conclusion

Modal DTS is more than a computational framework for modal expressions; it also offers a theoretical contribution that puts forth empirical claims within formal linguistics, thereby bridging the computational and empirical domains of natural language semantics.

Our future work will extend Modal DTS to provide a unified account of sentences involving modal expressions such as *may* and *will*, which present distinct contexts from those of *might* and *would*. Furthermore, given a variety of analyses of modal subordination proposed in formal semantics, a next step will be to conduct empirical comparisons with these alternative accounts.

Acknowledgments

We are grateful to the two anonymous reviewers for their insightful comments. This research was partially supported by JST CREST Grant Number JPMJCR20D2, Japan, and JSPS KAKENHI Grant Number JP23H03452, Japan.

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.

Nicholas Asher and Eric McCready. 2007. Were, would, might and a compositional account of counterfactuals. In *Journal of Semantics*, volume 24, pages 93–129. Oxford University Press.

Daisuke Bekki. 2023. A proof-theoretic analysis of weak crossover. In *New Frontiers in Artificial Intelligence*, pages 228–241. Springer.

Daisuke Bekki and Koji Mineshima. 2017. Context-passing and underspecification in dependent type semantics. In Stergios Chatzikyriakidis and Zhaohui Luo, editors, *Studies of Linguistics and Philosophy*, pages 11–41. Springer International Publishing.

Anette Frank and Hans Kamp. 1997. On context dependence in modal constructions. In *Proceedings of Semantics and Linguistic Theory*, volume 7, pages 151–168. Linguistic Society of America.

Stefan Kaufmann. 2000. Dynamic context management. In *Formalizing the Dynamics of Information*, pages 171–188. The University of Chicago Press.

Ezra Keshet and Steven Abney. 2024. Intensional anaphora. In *Semantics and Pragmatics*, volume 17, pages 1–54. Linguistic Society of America.

Per Martin-Löf. 1984. Intuitionistic type theory. volume 17. Bibliopolis.

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.

Aleksandar Nanevski, Frank Pfenning, and Brigitte Pientka. 2008. Contextual modal type theory. In *ACM Transactions on Computational Logic*, volume 9, pages 1–49. Association for Computing Machinery.

Craige Roberts. 1989. Modal subordination and pronominal anaphora in discourse. In *Linguistics and Philosophy*, volume 12, pages 683–721. Springer.

Ribeka Tanaka, Koji Mineshima, and Daisuke Bekki. 2015. Resolving modal anaphora in dependent type semantics. In *New Frontiers in Artificial Intelligence*, pages 83–98. Springer.

Robert van Rooij. 2005. A modal analysis of presupposition and modal subordination. In *Journal of Semantics*, volume 22, pages 281–305. Oxford University Press.