

# Compositional Semantics and Inference System for Temporal Order based on Japanese CCG

Tomoki Sugimoto and Hitomi Yanaka

The University of Tokyo

{sugimoto.tomoki, hyanaka}@is.s.u-tokyo.ac.jp

## Abstract

Natural Language Inference (NLI) is the task of determining whether a premise entails a hypothesis. NLI with temporal order is a challenging task because tense and aspect are complex linguistic phenomena involving interactions with temporal adverbs and temporal connectives. To tackle this, temporal and aspectual inference has been analyzed in various ways in the field of formal semantics. However, a Japanese NLI system for temporal order based on the analysis of formal semantics has not been sufficiently developed. We present a logic-based NLI system that considers temporal order in Japanese based on compositional semantics via Combinatory Categorical Grammar (CCG) syntactic analysis. Our system performs inference involving temporal order by using axioms for temporal relations and automated theorem provers. We evaluate our system by experimenting with Japanese NLI datasets that involve temporal order. We show that our system outperforms previous logic-based systems as well as current deep learning-based models.

## 1 Introduction

Natural Language Inference (NLI) is the task of determining whether a premise entails a hypothesis. In particular, NLI involving temporal expressions is crucial. (1) is an example of English NLI involving temporal expressions.

- (1)  $P$ : I arrived in April 2021.  
 $\overline{H}$ : I arrived before May 2021. (entailment)

The inference example with temporal expressions is challenging. This is because we need to represent the meaning of sentences that contain temporal adverbs like *before* and *in*, temporal expressions like *April 2021*, and verb tenses like *arrived*, and to compute temporal order of events written in the sentences.

Thukral et al. (2021) showed that deep learning-based models (Liu et al., 2019; He et al., 2020)

trained on a standard NLI dataset such as Multi-Genre Natural Language Inference (MultiNLI; Williams et al. (2018)) failed to perform simple temporal inference as in (1). Furthermore, deep learning-based models have performed poorly on challenging NLI datasets that involve various temporal inferences such as FraCaS (Cooper et al., 1996) for English and JSeM (Kawazoe et al., 2015) for Japanese.

Recently, logical inference systems based on compositional semantics (Bos and Markert, 2005; Abzianidze, 2015; Mineshima et al., 2015, 2016; Bernardy and Chatzikyriakidis, 2017, 2020; Onishi et al., 2020) (i.e., semantics in which the meaning of a phrase is determined compositionally from the syntax and the meaning of the lexicon contained in the phrase) achieved high accuracy in FraCaS and JSeM. However, most previous systems did not cover temporal inference.

In addition, because most previous research on NLI has focused on English, research on other languages is desirable. In particular, research on NLI in Japanese is still in its infancy and is limited to deep learning-based systems using pre-trained language models and a few logical inference systems (Mineshima et al., 2016; Onishi et al., 2020). Onishi et al. (2020) attempted to implement a Japanese logical inference system for temporal inference. However, the focus of this previous research was limited to a few temporal clauses in Japanese, and temporal adverbs are out of scope. Thus, there is still room for improvement in the accuracy of temporal inference in Japanese.

In this study, our aim is to realize the compositional semantics and a logical inference system for temporal inference in Japanese based on Combinatory Categorical Grammar (CCG) (Steedman, 2000; Bekki, 2010) to derive a transparent syntax-semantics interface and the analysis of tense and aspect studied in formal semantics (Kamp and Reyle, 1993; Yoshimoto, 2000; Kaufmann and Miyachi,

2011; Utsugi and Bekki, 2015; Ogihara, 2017; Jacobsen, 2018). We focus on temporal order and develop a Japanese logical inference system for temporal order.

In our system, a CCG parser first parses the premise and hypothesis sentences and converts them into CCG trees. Based on the analysis of the compositional semantics, we then modify the obtained CCG trees. Next, using *ccg2lambda* (Martínez-Gómez et al., 2016), the meaning of the whole sentence is derived as a logical form. Finally, we attempt to prove the entailment relations between the obtained logical forms by an automated theorem prover Vampire (Kovács and Voronkov, 2013).

We experiment with two NLI datasets involving temporal order in Japanese: JSeM and a Japanese translation of the NLI dataset focusing on temporal inference (Thukral et al., 2021). We compare our system with the previous Japanese logical inference system (Onishi et al., 2020) and the Japanese BERT model (Devlin et al., 2019). Our experiments show that our system outperforms previous logical inference systems as well as current deep learning-based models. Our system will be available for research use at <https://github.com/ynklab/ccgtemp>.

## 2 Background

Tense and aspect are important linguistic phenomena related to temporal expressions. This section provides standard background on the semantics of temporal expressions in Japanese, which have been analyzed in previous studies (Yoshimoto, 2000; Kaufmann and Miyachi, 2011; Utsugi and Bekki, 2015; Ogihara, 2017; Jacobsen, 2018).

In Japanese, verb tense is classified into past (*-ta*) and non-past (*-ru*), and aspect is classified into stative (like *iru*) and non-stative (like *kuru*). The temporal interpretation of a matrix clause (i.e., a clause that contains a subordinate clause) is determined by the combination of tense and aspect, and is expressed by the constraints imposed on the relation between speech time and reference time. Speech time represents the time that a sentence is uttered, and reference time is a concept proposed by Reichenbach (1947) and refers to the time used with location time (i.e., time when an event occurs) and speech time to represent the meaning of tense. Table 1 shows the temporal interpretation of a matrix clause determined by the combination of tense

and aspect and example sentences corresponding to each combination.

Past	Stative	Relation	Example
+	+	$r < s$	<i>Taro-ga ita</i> 'Taro was here'
	-	$r < s$	<i>Taro-ga kita</i> 'Taro came'
-	+	$r \geq s$	<i>Taro-ga iru</i> 'Taro is here'
	-	$r > s$	<i>Taro-ga kuru</i> 'Taro is coming'

Table 1: Constraints imposed on the relation between speech time  $s$  and reference time  $r$  by tense and aspect and example sentences

To analyze the temporal interpretation of embedded clauses, the concepts of absolute tense and relative tense are necessary. Absolute tense means that the temporal interpretation is determined by the relation between the speech time and the reference time, as in the matrix clause. However, relative tense means an interpretation in which the temporal interpretation does not depend on the relation between the speech time and the reference time. We explain the details with examples in Section 3.2.

This paper uses CCG to formalize the syntactic analysis of our method and analyzes the compositional semantics of temporal expressions based on the analysis by Kaufmann and Miyachi (2011).

## 3 Compositional Semantics and Inference for Tense

### 3.1 Semantic Representations for Verb Tense

This section explains the semantic representations for verb tense. Consider the following sentences.

- (2) a. Taro-ga kuru  
Taro-NOM come-NP  
'Taro is coming'
- b. Taro-ga kita  
Taro-NOM come-P  
'Taro came'

(2a) is non-past tense (NP), and (2b) is past tense (P). (2a) means that the event of Taro's coming occurs after the speech time, whereas (2b) means that the event occurred before the speech time. Thus, for the speech time  $s$  and the reference time  $r$ ,  $r > s$  in (2a) and  $r < s$  in (2b). Here,  $r$  and  $s$  both

$$\begin{array}{c}
\frac{\text{太郎 (Taro)}}{NP_{[nc, nm, f]}} \quad \frac{\text{が}^s \text{(NOM)}}{NP_{[ga, nm, f]} \setminus NP_{[nc, nm, f]}} \\
\frac{\lambda NF.(N(\lambda x. \top, \text{Taro}) \wedge F(\text{Taro}))}{\lambda NF.(N(\lambda x. \top, \text{Taro}) \wedge F(\text{Taro}))} \quad \frac{\lambda Q.Q}{\lambda Q.Q} < \\
\frac{NP_{[ga, nm, f]}}{\lambda NF.(N(\lambda x. \top, \text{Taro}) \wedge F(\text{Taro}))} < \\
\frac{\text{来る (come-NP)}}{S_{[nm, base, f]} \setminus NP_{[ga, nm, f]}} \\
\frac{\lambda QC1C2C3Ki1j1.Q(\lambda I.I, \lambda x. \exists e1.(K(\lambda e2i2j2.(come(e2)) \wedge \text{during}(\text{time}(e2), j2) \wedge \text{after}(j2, i2)), e1, i1, j1) \wedge C1(x, e1, \text{Nom})))}{\lambda QC1C2C3Ki1j1.Q(\lambda I.I, \lambda x. \exists e1.(K(\lambda e2i2j2.(come(e2)) \wedge \text{during}(\text{time}(e2), j2) \wedge \text{after}(j2, i2)), e1, i1, j1) \wedge C1(x, e1, \text{Nom})))} < \\
\frac{S_{[nm, base, f]}}{\lambda C1C2C3Ki1j1.(\top \wedge \exists e1.(K(\lambda e2i2j2.(come(e2)) \wedge \text{during}(\text{time}(e2), j2) \wedge \text{after}(j2, i2)), e1, i1, j1) \wedge C1(\text{Taro}, e1, \text{Nom})))} < \\
\frac{S_{[nm, base, t]}}{\exists sr.(\top \wedge \exists e1.(come(e1) \wedge \text{during}(\text{time}(e1), r) \wedge \text{after}(r, s) \wedge (\text{Nom}(e1) = \text{Taro})))} <
\end{array}$$

Figure 1: CCG derivation tree for *Taro-ga kuru* (*Taro is coming*).  $\top$  denotes the tautology.

represent intervals and  $r < s$  means the end of the interval  $r$  is before the beginning of the interval  $s$ . Another interpretation of time is instance semantics, which treats time as an instance, but in this study, we follow the standard treatment of time as an interval (Kamp and Reyle, 1993; Bernardy and Chatzikyriakidis, 2020).

Following Kamp and Reyle (1993), in this study, the time of an event is represented by its relationship with the reference time. Then, the meaning of (2a) and (2b) can be expressed by the following logical expressions, where  $\text{tgk}$  is the predicate that represents the event Taro’s coming, time is the function that returns the time when the event occurred and  $e$  is a variable representing the event.

- (3) a.  $\exists e.(\text{tgk}(e) \wedge \text{time}(e) \subseteq r \wedge r > s)$   
b.  $\exists e.(\text{tgk}(e) \wedge \text{time}(e) \subseteq r \wedge r < s)$

The meanings of (3a) and (3b) are as shown in the Figure 2 and Figure 3. Figure 1 shows the CCG derivation tree for (2a).

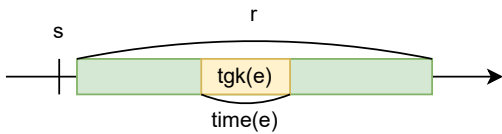


Figure 2: Temporal interpretation of (2a)

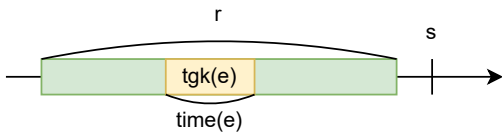


Figure 3: Temporal interpretation of (2b)

### 3.2 Semantic Representations for Temporal Clause

Next, consider the following sentences with an embedded clause.

- (4) a. Taro-ga kuru mae-ni oyoida  
Taro-NOM come-NP before-LOC swim-P  
‘I swam before Taro came’  
b. Taro-ga kita ato-ni oyoida  
Taro-NOM come-NP after-LOC swim-P  
‘I swam after Taro came’

In (4a), the embedded clause is the non-past tense, and in (4b), the embedded clause is the past tense. As mentioned in Section 2, the temporal meaning of embedded clauses is interpreted using “relative tense.” Thus, the temporal meaning of embedded clauses is determined not by the relation between the speech time and the reference time of the embedded clause but by the relation between the reference time of the matrix clause and the reference time of the embedded clause. For the reference time of the embedded clause  $t$  and the reference time of the matrix clause  $r$ , we then have  $t > r$  in (4a), and  $t < r$  in (4b).

Therefore, using the same predicates and functions as Section 3.1, the meaning of the embedded clauses can be expressed by the following logical formulas.

- (5) a.  $\exists e.(\text{tgk}(e) \wedge \text{time}(e) \subseteq t \wedge t > r)$   
b.  $\exists e.(\text{tgk}(e) \wedge \text{time}(e) \subseteq t \wedge t < r)$

By combining these logical formulas with the meanings of the matrix clauses interpreted in the same way as Section 3.1, the meanings of sentences with the embedded clauses can be expressed by the following logical formulas, where  $o$  is the predicate that represents the event of my swimming.

- (6) a.  $\exists t.(\exists e1.(\text{tgk}(e1) \wedge \text{time}(e1) \subseteq t \wedge t > r) \wedge \exists e2.(o(e2) \wedge \text{time}(e2) \subseteq r \wedge r < s))$   
b.  $\exists t.(\exists e1.(\text{tgk}(e1) \wedge \text{time}(e1) \subseteq t \wedge t < r) \wedge \exists e2.(o(e2) \wedge \text{time}(e2) \subseteq r \wedge r < s))$

The meanings of (6a) and (6b) are as shown in the Figure 4 and Figure 5.

This study interprets the temporal meaning of sentences with embedded clauses in this way.

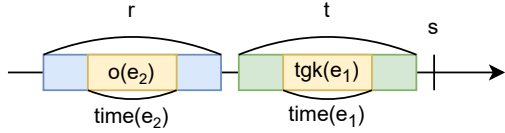


Figure 4: Temporal interpretation of (4a)

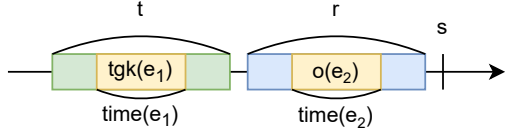


Figure 5: Temporal interpretation of (4b)

### 3.3 Semantic Representations for Temporal Adverb

#### 3.3.1 Syntactic analysis

An example of the temporal adverbs targeted in this paper is shown in bold in the following.

- (7) Taro-ga **4 gatsu 3 nichi izen-ni** kita  
 Taro-NOM **4 month 3 day before** come-P  
 ‘Taro came **before April 3**’

More generally, we analyze temporal adverbs comprising various types of absolute temporal expressions (e.g., date, day of the week, and time) and temporal connectives *izen* (before) and *ikou* (after). Absolute temporal expressions are temporal expressions that do not depend on the speech time, in contrast to relative temporal expressions such as *today* that depend on the speech time. In this study, temporal adverbs containing relative temporal expressions are out of scope and left for future work.

In temporal adverbs containing absolute temporal expressions, the particle *-ni* is unnecessary. For example, the following three sentences are all acceptable and have the same meaning.

- (8) a. 4 gatsu 3 nichi ni Taro-ga kita  
 4 month 3 day on Taro-NOM come-P  
 ‘Taro came on April 3’  
 b. 4 gatsu 3 nichi, Taro-ga kita  
 4 month 3 day Taro-NOM come-P  
 ‘Taro came on April 3’  
 c. 4 gatsu 3 nichi Taro-ga kita  
 4 month 3 day Taro-NOM come-P  
 ‘Taro came on April 3’

Thus, *-ni* can be analyzed as a separation of clauses like a comma and does not have any meaning. Before considering the syntactic category of *-ni*, let us

consider absolute temporal expressions. As shown in (8c), absolute temporal expressions are combined with sentences such as *Taro-ga kita*. Therefore,  $S/S$  is assigned as the syntactic category of the absolute temporal expression *4 gatsu 3 nichi*. As mentioned above, because *-ni* plays the role of connecting the preceding and following clauses,  $(S/S) \setminus (S/S)$  is appropriate as its syntactic category.

In addition, absolute temporal expressions like *4 gatsu 3 nichi* can be a noun phrase  $NP$ , as in Figure 6. In this example, the syntactic category of *4 gatsu 3 nichi* is  $NP$ , and the syntactic category of *izen* is  $(S/S) \setminus NP$ . We explain the reason why absolute temporal expressions are used as both  $NP$  and  $S/S$  from a semantic perspective in the next paragraph.

$$\frac{\frac{4\text{月}3\text{日 (April 3)}}{NP} \quad \frac{\text{以前 (before)}}{(S/S) \setminus NP}}{S/S} \quad \frac{\text{に}}{(S/S) \setminus (S/S)}}{S/S}$$

Figure 6: CCG derivation tree for *4 gatsu 3 nichi izen ni* (before April 3).

#### 3.3.2 Semantic analysis

We treat absolute temporal expressions (e.g., *4 gatsu 3 nichi* (April 3)) as multi-word expressions. Consider the expression *4 gatsu 3 nichi*. We can decompose the expression into four constituents as follows.

$$\begin{aligned} [4 \text{ gatu } 3 \text{ nichi}] &= [4 \text{ gatu}][3 \text{ nichi}] \\ &= [[4][\text{gatu}]] [[3][\text{nichi}]] \end{aligned}$$

A current Japanese CCG parser (Yoshikawa et al., 2017) analyzes each constituent as the syntactic category  $4 = NP$ ,  $\text{gatsu} = (NP/NP) \setminus NP$ ,  $3 = NP$ , and  $\text{nichi} = NP/NP$ , respectively. The semantic template for  $NP$  is  $\lambda E N F. \exists x. (N(E, x) \wedge F(x))$ , which means ‘‘some bound variable  $x$  is associated with the word  $E$ .’’ Now 4 and 3 are both  $NP$ , so 4 and 3 have different bound variables associated with them. This bound variable refers to the interval. Essentially, because *4 gatsu 3 nichi* refers to only one interval, 4 and 3 need to be associated with the same interval. The correct meaning cannot be derived when 4 and 3 are associated with different bound variables.

Thus, we treat temporal expressions such as *4 gatsu 3 nichi* as multi-word expressions and set

Category	Expression	Semantic Template
$S \setminus NP$	来る (is coming)	$\lambda Q C1 C2 C3 K i1 j1.Q(\lambda I.I, \lambda x.\exists e1.(K(\lambda e2 i2 j2.(come(e2) \wedge during(time(e2), j2) \wedge after(j2, i2)), e1, i1, j1) \wedge C1(x, e1, Nom)))$
$S \setminus NP$	来た (came)	$\lambda Q C1 C2 C3 K i1 j1.Q(\lambda I.I, \lambda x.\exists e1.(K(\lambda e2 i2 j2.(come(e2) \wedge during(time(e2), j2) \wedge before(j2, i2)), e1, i1, j1) \wedge C1(x, e1, Nom)))$
$NP$	4月3日 (April 3rd)	$\lambda N F.\exists x.(N(\lambda y.(normalized\_time(y) = 40300), x) \wedge F(x))$
$S/S$	4月3日 (April 3rd)	$\lambda S C1 C2 C3 K i1 j1.S(C1, C2, C3, \lambda J e1 i2 j2.K(\lambda e2 i3 j3.(J(e2, i3, j3) \wedge \exists x.((normalized\_time(x) = 40300) \wedge (x = j3))), e1, i2, j2), i1, j1)$
$(S/S) \setminus NP$	以前 (before)	$\lambda Q S C1 C2 C3 K i1 j1.S(C1, C2, C3, \lambda J e1 i2 j2.K(\lambda e2 i3 j3.(J(e2, i3, j3) \wedge Q(\lambda I.I, \lambda x.before(j3, x))), e1, i2, j2), i1, j1)$
$(S/S) \setminus (S/S)$	に (on)	$\lambda V3.V3$

Table 2: Examples of semantic templates.

up a semantic template as shown in Table 2. This semantic template allows us to derive the meaning of a temporal expression associated with only one bound variable. In this template, the function `normalized_time` takes interval as an argument and returns its actual time, which can be set in the format `YYYYMMDDHH` from absolute temporal expressions. For example, for interval  $x$ , which represents *April 3*, the value is `normalized_time(x) = 0000040300`. In this example, year and hour are not explicitly written, so zero-padding is applied to them.

As shown in Figure 6, *4 gatsu 3 nichu* functions as  $NP$  when connected to *izen* and as  $S/S$  when used by itself. This phenomenon can be analyzed as follows. Temporal expressions such as *4 gatsu 3 nichu* and *4 gatsu 3 nichu izen* play the role of representing the time of the sentence. Consider the following sentences.

(9) *4 gatsu 3 nichu ni Taro-ga kita*  
4 month 3 day on Taro-NOM come-P  
'Taro came on April 3'

(10) *4 gatsu 3 nichu izen-ni Taro-ga kita*  
4 month 3 day before Taro-NOM come-P  
'Taro came before April 3'

In (9), the location time of the event *Taro-ga kita* (*Taro came*) is *4 gatsu 3 nichu* (*April 3*), and in (10), the location time of the event *Taro-ga kita* (*Taro came*) is *4 gatsu 3 nichu izen* (*before April 3*). The expressions that represent temporal adverbs such as *4 gatsu 3 nichu* (*April 3*) and *4 gatsu 3 nichu izen* (*before April 3*) must have the syntactic category of  $S/S$ , so *4 gatsu 3 nichu* changes from  $NP$  to  $S/S$ .

Next, the semantic template for *izen* was determined as shown in Table 2. The temporal meaning of *izen* is represented as the lambda expression

$\lambda x.before(j3, x)$ , which indicates that the expression “doing before  $x$ ” means “doing in  $j3$  before  $x$ .” Finally, the meaning of temporal expressions can be derived by setting up a template with *-ni* and a comma as meaningless words, as described in Section 3.3.1.

### 3.4 Inference with Tense

We introduce a set of axioms for temporal relations and temporal expressions to perform inference for temporal order. Allen (1983) defined 13 relations between time intervals. The previous logic-based inference system (Onishi et al., 2020) introduced 169 axioms for these 13 temporal relations. Six of the 13 temporal relations, *meets*, *met\_by*, *starts*, *started\_by*, *finishes*, and *finished\_by* are special cases of other relations in implementing axioms. For example, *meets* is a special case of *before* where the end of the preceding interval coincides with the beginning of the following interval. *meets* is necessary for inferences involving temporal clauses such as *soon after*. Thus, we consider that those six relations are redundant in performing the temporal inference involving temporal order in this study. We therefore merged them into the most similar relations: merged *meets* into *before*, *met\_by* into *after*, *starts* into *during*, *started\_by* into *contains*, *finishes* into *during*, and *finished\_by* into *contains*, respectively. In summary, we introduce 49 axioms corresponding to seven temporal relations: *before*, *after*, *overlaps*, *overlapped\_by*, *during*, *contains*, and *equal*.

In addition, we speculate 30 additional axioms for temporal expressions in Japanese such as *izen* (*before*) and *ikou* (*after*), and those for identity conditions of speech times between premises and hypotheses. Table 3 shows examples of the axioms.

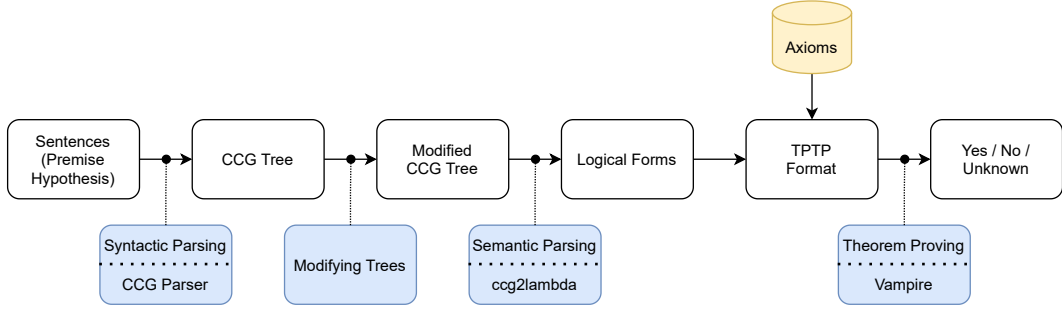


Figure 7: Overview of our system

Pattern	Axiom
transitivity of <i>before</i> relations	$\forall A, B, C. (\text{before}(A, B) \wedge \text{before}(B, C) \rightarrow \text{before}(A, C))$
insertion of <i>izen</i>	$\forall I, X, R. ((\text{nort}(X) = I \wedge (X = R)) \rightarrow (\forall J. ((I \leq J) \rightarrow (\exists Y. (\text{nort}(Y) = J \wedge \exists Z. (\text{before}(Z, Y) \wedge (Z = R)))))))$
replacement of <i>izen</i>	$\forall I, X, R. ((\text{nort}(X) = I \wedge \text{before}(R, X)) \rightarrow (\forall J. ((I \leq J) \rightarrow (\exists Z. (\text{nort}(Z) = J \wedge \text{before}(R, Z))))))$
identity condition of speech times	$\forall S_1, S_2. (\text{speech\_time}(S_1) \wedge \text{speech\_time}(S_2) \rightarrow S_1 = S_2)$

Table 3: Examples of axioms. nort indicates a normalized\_time function.

## 4 System Overview

Figure 7 shows the pipeline of our system. Our system consists of three main steps. First, natural language sentences of premises and hypotheses are converted into modified CCG trees by CCG parsing and modifying trees. Next, a meaning from the semantic templates is assigned to each lexical item. The semantics in lexical items are then composed by *ccg2lambda* to derive a logical formula that represents the meaning of the whole sentence. Finally, an automated theorem prover determines whether the logical formula of the hypothesis is provable from the logical formula of the premises. In this section, we describe each of these steps.

### 4.1 Syntactic Analysis

The syntactic analysis, which obtains CCG parsing trees of input sentences, consists of two steps. First, we use the tokenizer to tokenize sentences and a CCG parser to obtain a CCG tree. We use *depccg* (Yoshikawa et al., 2017), a standard Japanese CCG parser, trained on the Japanese CCG-Bank (Uematsu et al., 2013) for the first step.

Second, if the sentence contains temporal expressions, we extract the subtrees in which the leaves are temporal expressions from the CCG tree of the whole sentence. The extracted CCG subtree is then transformed into an appropriate form. Figure 8 and Figure 9 show the temporal expression subtrees *4 gatsu 3 nichi ni* (*on April 3*) before and after the conversion. As another possible way of implemen-

tation for obtaining correct CCG trees for temporal expressions, we can improve the CCG parser itself. However, to do that, we need to re-train the morphological analyzer and the CCG parser to correctly handle a variety of temporal expressions. We do not take this approach because it is too costly.

$$\frac{\frac{\frac{4}{NP/NP} \quad \frac{\text{月(month)}}{NP/NP}}{NP/NP} >_B \quad \frac{\frac{3}{NP} \quad \frac{\text{日(day)}}{NP \setminus NP}}{NP} <}{NP} > \quad \frac{\text{に}}{(S/S) \setminus NP} <$$

Figure 8: CCG derivation tree before conversion.

$$\frac{\frac{4\text{月}3\text{日(April 3)}}{S/S} \quad \frac{\text{に}}{(S/S) \setminus (S/S)}}{S/S} <$$

Figure 9: CCG derivation tree after conversion.

### 4.2 Semantic Analysis

In semantic analysis, each leaf (lexical item) of the CCG tree obtained in the syntactic analysis is assigned a meaning from the semantic templates. The lexical items are then combined according to the CCG derivation tree to derive a logical formula that expresses the meaning of the entire sentence. The composition is performed using *ccg2lambda* in Japanese (Mineshima et al., 2016).

In order to assign meaning to the temporal expressions, we set up semantic templates for lexical items such as absolute temporal expressions and *izen*. We provide a set of semantic templates, which

contains 150 lexical entries. The number of lexical entries assigned to CCG categories is 92, and the number of entries directly assigned to specific words is 58. Table 2 shows the examples of semantic templates.

As a representation language, we use the typed first-order form of the Thousands of Problems for Theorem Provers (TPTP; Sutcliffe (2017)) format. We use standard interval semantics (Dowty, 1979; Bennett and Partee, 1978) and introduce an interval type to express time instances as intervals and their relations in logical expressions. We use four basic types: E (Entity), Ev (Event), Prop (Proposition) and I (Interval). The types of expressions we adopt are defined by

$$T ::= E \mid Ev \mid Prop \mid I \mid T1 \Rightarrow T2$$

where  $T1 \Rightarrow T2$  is a function type. Because the logical expressions derived by `cgg2lambda` are not typed, we implement automatic completion of variable types, predicate types, and definitions of predicates.

### 4.3 Theorem Proving

In theorem proving, we use the state-of-the-art first-order logic automated theorem prover Vampire (Kovács and Voronkov, 2013) which accepts TPTP formats to determine whether or not a hypothesis is provable from premises using the logical formula derived in Section 4.2. The system outputs “yes” (entailment) when the hypothesis can be proved from the premises, “no” (contradiction) when the negation of the hypothesis can be proved from the premises, and “unknown” (neutral) when neither can be proved. We use the fastest mode, CASC mode, and set the timeout of Vampire to a maximum of 300 sec for our experiments.

Even though Vampire is a fast theorem prover, it takes too long to prove the problems, whose premises and hypothesis are too complex. When proving the negation of a hypothesis, it turns out that simply negating the logical formula increases the complexity. Therefore, this study uses the symmetrical relationship between *ikou* and *izen* to replace *izen* and *ikou* in the hypothesis with *ikou* and *izen*, respectively, to negate the logical formula without increasing the complexity.

## 5 Experiments

### 5.1 Experimental Setup

We evaluate our system on two datasets. First, JSeM (Kawazoe et al., 2015) is a Japanese version of the FraCaS (Cooper et al., 1996) test suite, which consists of nine sections, each containing representative problems of semantically challenging inferences involving various linguistic phenomena. In this study, we use 23 problems involving temporal order in temporal reference section. The distribution of gold answer labels for the problems is (yes/no/unknown) = (12/4/7).

PLMUTE Section: time_multi, No. 11, Gold answer: yes	
<i>P</i>	午後7時以降ロビンは両親を訪ねた。 (After 7 p.m. Robin visited her parents.)
<i>H</i>	16時以降ロビンは両親を訪ねた。 (After 16:00 Robin went to visit her parents.)
PLMUTE Section: day, No. 239, Gold answer: no	
<i>P</i>	月曜日以前、食料品店が閉店した。 (Before Monday, the grocery store was closed.)
<i>H</i>	火曜日以降、食料品店が閉店した。 (After Tuesday, the grocery store was closed.)
JSeM No. 645, Gold answer: yes	
<i>P</i>	1992年以来、ITELはバーミンガムにある。 (Since 1992 ITEL has been in Birmingham.)
	現在、1996年である。 (It is now 1996.)
<i>H</i>	ITELは1993年にはバーミンガムにあった。 (ITEL was in Birmingham in 1993.)

Table 4: Examples of problems from JSeM and PLMUTE\_ja.

Second, we created an NLI dataset focusing on temporal order in Japanese from the existing NLI dataset (which we refer to as PLMUTE) for temporal inference in English proposed by Thukral et al. (2021) because Japanese NLI datasets involving diverse temporal adverbs were not well developed. We used the ordering section of PLMUTE, which collects problems related to ordering various temporal adverbs for a date, day of the week, and time. The original PLMUTE is automatically generated from 71 templates by a program. Thus, we manually translated the templates into Japanese and modified the program to generate the dataset to make the generated dataset natural in Japanese. We automatically generated a Japanese translation of the original PLMUTE by using the translated templates and modified program. We call the dataset PLMUTE\_ja. PLMUTE\_ja consists of nine sections: year (340 problems), month (480 problems), date (560 problems), date\_DMY (340 problems), date\_MY (340 problems), day

System	year	month	date	date _dmy	date _my	day	time _12	time _24	time _multi
Majority	.382	.421	.425	.403	.379	.396	.368	.415	.418
BERT	JSNLI	.394	.413	.382	.400	.400	.378	.415	.368
	few	.509	.517	.509	.491	.476	.518	.440	.453
	all	.997	1.000	.998	.985	.982	1.000	1.000	.998
Onishi et al. (2020)	.238	.265	.239	.206	.244	.291	.290	.225	.253
<b>Our system</b>	1.000	1.000	.980	.971	.974	.984	.943	.970	.953

Table 5: Accuracy on the PLMUTE\_ja test suite.

(560 problems), time\_12 (400 problems), time\_24 (400 problems), and time\_multi (400 problems). The distribution of gold answer labels for the problems is (yes/no/unknown) = (1353/1502/965). Table 4 shows examples of problems in JSeM and PLMUTE\_ja.

We compared our system with the following previous logic-based inference system and deep learning-based models in Japanese.

**Logic-based inference system** We used the logic-based inference system for temporal inference in Japanese proposed by Onishi et al. (2020). Onishi et al. (2020)’s system used Coq, a higher-order theorem prover based on natural deduction.

**Deep learning-based model** We used the Japanese BERT (Devlin et al., 2019) model (cl-tohoku/bert-base-japanese-whole-word-masking) of Huggingface transformers<sup>1</sup> as a deep learning-based model. This Japanese BERT model is the most commonly used pre-trained language model for Japanese in huggingface/transformers. In this study, we experimented with the following three models: **BERT\_JSNI** is Japanese BERT fine-tuned on a large Japanese NLI dataset JSNLI (Yoshikoshi et al., 2020) (533,005 examples), a Japanese translation of the SNLI dataset (Bowman et al., 2015), which is one of the most widely used NLI datasets. **BERT\_few** is Japanese BERT fine-tuned on the PLMUTE\_ja minimal training set with two examples each of different combinations of tenses and sections (360 examples). **BERT\_all** is Japanese BERT fine-tuned on the entire PLMUTE\_ja training set (11,220 examples).

<sup>1</sup><https://huggingface.co/transformers/>

System	Accuracy	
BERT	JSNLI	.522
	few	.217
	all	.435
Onishi et al. (2020)	.478	
<b>Our system</b>	.783	

Table 6: Accuracy on the problems involving temporal order in the JSeM test suite.

## 6 Results and Discussion

### 6.1 Results

The results on the problems involving temporal order in JSeM are shown in Table 6. As the table shows, our system outperforms all models.

The results on the PLMUTE\_ja test set are shown in Table 5. As the table shows, our system outperforms all models except BERT\_all. Although the performance is slightly inferior to BERT\_all, the performance is comparable to BERT\_all with 11,220 training data. The experiment with Japanese BERT + PLMUTE\_ja reproduced the results of the experiment with English RoBERTa + PLMUTE conducted by Thukral et al. (2021). That is, although the model trained on all of the PLMUTE training sets could achieve high accuracy, the model trained on either the large standard NLI dataset or the minimal training set could only achieve low accuracy.

We also compared the average proof time for all four problems for which both our system and Onishi et al. (2020)’s system output “yes”. Our system was faster than the previous logic-based system: the average proof time for our system was 1.98 seconds, while Onishi et al. (2020)’s system was 3.11 seconds.



$P_1$	ジョーンズが契約書を修正した。 (Jones revised the contract.)
$P_2$	スミスが契約書を修正した。 (Smith revised the contract.)
$P_3$	ジョーンズがスミスより先に契約書を修正した。 (Jones revised the contract after Smith did.)
$H$	スミスはジョーンズより後に契約書を修正した。 (Smith revised the contract before Jones did.)
Gold answer: yes (JSeM No. 659)	

Table 7: An example of problem our system did not solve.

## 6.2 Error Analysis

In this section, we discuss the error analysis in the experiments. Our system did not solve the problems involving comparative deletion and temporal connectives such as *yorī mae* (*before*) and *yorī ato* (*after*), as shown in Table 7.

Although *yorī mae* and *yorī ato* have similar meanings to *izen* and *ikou*, they have different meanings. For example, 4 *gatsu* 3 *nichi izen* includes April 3rd, while 4 *gatsu* 3 *nichi yorī-mae* does not include April 3rd. In addition, *yorī mae* is more difficult to analyze than *izen* because it consists of two words, *yorī* and *mae* that require the analysis of comparative deletion, which we leaves for future work.

## 7 Conclusion

In this paper, we compositionally derived semantic representations of sentences with tense and aspect in Japanese based on CCG. We developed a logic-based NLI system that considers temporal order in Japanese. We evaluated our system by experimenting with two Japanese NLI datasets involving temporal order. Our system performed more robustly than previous logic-based systems as well as current deep learning-based models. The experimental results of our system suggest that a logical NLI system based on an analysis of tense in formal semantics is effective for temporal inference. Other previous studies of logic-based methods have shown the effectiveness of NLI systems based on the analysis of various semantics such as degree semantics (Haruta et al., 2020). By combining them, we will be able to construct one NLI system capable of performing a variety of inferences. In the future, we plan to cover various temporal inferences involving comparative deletion and temporal anaphora. Furthermore, we plan to construct inference test sets for these challenging inferences.

## Acknowledgements

We thank the three anonymous reviewers for their helpful comments and feedback. This work was supported by PRESTO, JST Grant Number JP-MJPR21C8, 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, Lisbon, Portugal. Association for Computational Linguistics.
- James F. Allen. 1983. [Maintaining knowledge about temporal intervals](#). *Commun. ACM*, 26(11):832–843.
- Daisuke Bekki. 2010. *A Formal Theory of Japanese Grammar: The Conjugation System, Syntactic Structures, and Semantic Composition (in Japanese)*. Kuroshio.
- Michael Bennett and Barbara Hall Partee. 1978. *Toward the logic of tense and aspect in English*, volume 84. Indiana University Linguistics Club Bloomington.
- Jean-Philippe Bernardy and Stergios Chatzikyriakidis. 2017. [A type-theoretical system for the FraCaS test suite: Grammatical framework meets coq](#). In *IWCS 2017 - 12th International Conference on Computational Semantics - Long papers*.
- Jean-Philippe Bernardy and Stergios Chatzikyriakidis. 2020. [Fracas: Temporal analysis](#). *arXiv preprint arXiv:2012.10668*.
- Johan Bos and Katja Markert. 2005. [Recognising textual entailment with logical inference](#). In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 628–635, Vancouver, British Columbia, Canada. Association for Computational Linguistics.
- Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. [A large annotated corpus for learning natural language inference](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.
- Robin Cooper, Richard Crouch, Jan van Eijck, Chris Fox, Josef van Genabith, Jan Jaspars, Hans Kamp, Manfred Pinkal, David Milward, Massimo Poesio, Stephen Pulman, Ted Briscoe, Holger Maier, and Karsten Konrad. 1996. [Using the framework](#). Technical report, Technical Report LRE 62-051 D-16, The FraCaS Consortium.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of](#)

- deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- David R Dowty. 1979. *Word meaning and Montague grammar: The semantics of verbs and times in generative semantics and in Montague’s PTQ*. Reidel, Dordrecht.
- Izumi Haruta, Koji Mineshima, and Daisuke Bekki. 2020. Logical inferences with comparatives and generalized quantifiers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 263–270, Online. Association for Computational Linguistics.
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced BERT with disentangled attention. *arXiv preprint arXiv:2006.03654*.
- Wesley M. Jacobsen. 2018. *Tense and Aspect*, page 332–356. Cambridge University Press.
- Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Dordrecht: Kluwer Academic Publishers.
- Stefan Kaufmann and Misa Miyachi. 2011. On the temporal interpretation of japanese temporal clause. *Journal of East Asian Linguistics*, 20(1):33–76.
- 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 *JSAI International Symposium on Artificial Intelligence*, pages 58–65. Springer.
- Laura Kovács and Andrei Voronkov. 2013. First-order theorem proving and Vampire. In *International Conference on Computer Aided Verification*, pages 1–35. Springer.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.
- 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, Berlin, Germany. Association for Computational Linguistics.
- 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, Lisbon, Portugal. Association for Computational Linguistics.
- Koji Mineshima, Ribeka Tanaka, Pascual Martínez-Gómez, Yusuke Miyao, and Daisuke Bekki. 2016. Building compositional semantics and higher-order inference system for a wide-coverage Japanese CCG parser. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2236–2242, Austin, Texas. Association for Computational Linguistics.
- Toshiyuki Ogihara. 2017. Tense and aspect. *The handbook of Japanese linguistics*, pages 326–348.
- Maiko Onishi, Hitomi Yanaka, Koji Mineshima, and Daisuke Bekki. 2020. Recognizing temporal relations in natural language based on ccg and theorem proving (in japanese). In *Proceedings of the Annual Conference of JSAI*, volume JSAI2020, pages 1E3GS904–1E3GS904.
- Hans Reichenbach. 1947. *Elements of Symbolic Logic*. London: Dover Publications.
- Mark Steedman. 2000. *The syntactic process*, volume 24. MIT press Cambridge, MA.
- Geoff Sutcliffe. 2017. The TPTP problem library and associated infrastructure. *Journal of Automated Reasoning*, 59(4):483–502.
- Shivin Thukral, Kunal Kukreja, and Christian Kavouras. 2021. Probing language models for understanding of temporal expressions. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 396–406, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Sumire Uematsu, Takuya Matsuzaki, Hiroki Hanaoka, Yusuke Miyao, and Hideki Mima. 2013. Integrating multiple dependency corpora for inducing wide-coverage Japanese CCG resources. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1042–1051, Sofia, Bulgaria. Association for Computational Linguistics.
- Maika Utsugi and Daisuke Bekki. 2015. Towards an analysis of tense and aspect in japanese by dependent type semantics (in japanese). In *Proceedings of the Annual Conference of JSAI*, volume JSAI2015, pages 2F4OS01a3–2F4OS01a3.
- Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume*

*I (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Masashi Yoshikawa, Hiroshi Noji, and Yuji Matsumoto. 2017. [A\\* CCG parsing with a supertag and dependency factored model](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 277–287, Vancouver, Canada. Association for Computational Linguistics.

Takumi Yoshikoshi, Daisuke Kawahara, and Sadao Kurohashi. 2020. Multilingualization of a natural language inference dataset using machine translation (in japanese). In *The 244th Meeting of Natural Language Processing*, pages 1–8.

Kei Yoshimoto. 2000. *Tense and Aspect in Japanese and English*. Peter Lang Publisher Inc.