

# 信念状態に基づく立論テキスト生成 Argument Text Generations Based on the State of Belief

福島秀顕 †, 萩原馨, 小谷亮 ‡, 池田光生  
Hideaki FUKUSHIMA†, Kaoru HAGIWARA, Akira KOTANI‡, Teruo IKEDA

(財) 新世代コンピュータ技術開発機構

† 国際電信電話株式会社      ‡ 三菱電機株式会社 情報電子研究所  
Institute for New Generation Computer Technology (ICOT)  
†KDD      ‡Mitsubishi Electric Corporation

## 1 はじめに

自然言語生成処理は、一般に、生成する内容を決定するプロセス (what\_to\_say 部) と、実際にその内容を表層表現に変換するプロセス (how\_to\_say 部) とから成ると考えられる。本稿では、ある「信念」状態における「立論」に基づいて生成する内容を決定する方式について述べる。ここでいう「立論」とは、システムに与えられた立論ゴールの正当性を論証することである。その論証過程を基にして生成する内容が決定され、最終的に得られる表層表現はそのゴールの正当性を主張するテキストとなる。

生成内容を決定するアプローチとしては、McKeown のスキーマのようにスクリプト的なテキスト構造を用いるもの [1] と、Mann の RST のように、より抽象的なテキスト構造を用いるもの [2] とがある。前者は構成的で、テキスト構造に文内容決定の手続きが記述されていると考えられるのに対して、後者はより分析的であり、テキスト構造としては文間の機能的な関係が記述される。本稿で述べる方式は後者に属するものであり、RST において RST スキーマの展開として行われていた処理を、システムが信念として保持している事象間の因果関係や、事象に対する判断を用いた推論により表現している。即ち、システムは、与えられたゴールのタイプに応じて決められた必要な条件を満足するまで、システムの信念内容を収集しながらテキスト構造を生成していく。

出力されるテキスト構造は、FTS (Functional Text Structure) と呼ばれるデータ構造によって記述される。スクリプト的なテキスト構造と異なり、FTS では文間の機能的な関係が記述されているだけなので、1つの FTS から統語的に多様な複数の文章を生成することができる。

第2章では、現在、われわれが開発中の立論テキスト生成システム DULCINEA の全体構成について述べる。DULCINEA は、与えられたゴールから FTS を構成する立論サブシステム、FTS からテキストの統語構造を生成する統語構造生成サブシステム、更に、統語構造から1文ごとに表層表現を生成する表層表現サブシステムの3つのモジュールから構成される。また、信念内容の記述や表層文の意味表

現などを記述する共通の知識表現言語として *QUIXOTE* [3] を用いている。

第3章では、DULCINEA の生成内容決定プロセスである立論サブシステムについて述べる。システム信念の構成と信念内容の記述形式を示し、次に立論ゴールの正当性を主張するための方策について述べる。さらに、テキスト構造の生成過程で得られた情報が表層表現の生成に反映されることについて説明する。

第4章で例題を示し、最後に第5章でまとめと今後の課題について述べる。

## 2 立論テキスト生成システム DULCINEA の構成

### 2.1 立論ゴール

DULCINEA に与えられる立論ゴールは、3.2節で述べるような様相表現を含んだものであり、事態とその否定に3種類の様相的なオペレータが付加された6種類を取り扱う。

must(P): “P しなければならない”  
must(¬P): “P してはならない”  
hb(P): “P した方がよい”  
hb(¬P): “P しない方がよい”  
may(P): “P してもよい”  
may(¬P): “P しなくてもよい”

### 2.2 立論サブシステムと信念

立論サブシステムは、立論ゴールに対して、「信念」と呼ばれる知識ベース内の規則を用いて推論を行い、推論の際の規則の適用過程に基づいて、その立論ゴールの正当性を主張する立論のテキスト構造を構成する。信念は、通常の規則や事実以外に、ある事態に対する判断(その事態が望ましい/望ましくない)を含んでいる。推論は原則として前向きに行われ、適宜後ろ向き推論も行われる。即ち、立論サブシステムは、立論ゴールの内容から信念内の判断に到達するまで前向き推論による予測を行う。信念の内容は知識表現言語

QUINOTEによって記述されており、また、立論サブシステムの出力であるテキスト構造は、2.3節で述べるFTSによって記述されている。立論サブシステムと信念の詳細については3節で述べる。

## 2.3 統語構造生成サブシステムと表層表現生成サブシステム

### 2.3.1 FTSと統語構造生成サブシステム

統語構造生成サブシステムの入力であるテキスト構造の記述にはFTS (Functional Text Structure)を用いる[4]。FTSは、単文における格表現との類推から、テキスト中の各文の間の機能的な関係を表現したものである。即ち、本システムで取り扱う何らかの主張を行うテキストでは、主張の内容である「結論」やその「理由」などが各文の機能となる。FTSは、各文の機能的な関係を記述したFTS項、順序制約および引力制約からなる。

#### 1. FTS項 (FTS term)

テキストの機能的な係り受け構造をあらわす素性構造であり、以下の3つの素性を持つことができる：

- thesis:** そのレベルにおける結論となる中心的な文
- reason:** thesisに対する理由づけをあらわす文
- anti\_t:** thesisとは逆の内容が予測されるような文

#### 2. 順序制約 (precedence)

2つの文の間の順序関係を表す制約。‘ $S_1 < S_2$ ’という順序制約は、テキストの中で $S_1$ よりも $S_2$ の方が後に現れなければならないという制約を表す。

#### 3. 引力制約 (gravity)

2つの文が近接して現れる方が好ましいことを表す制約。‘ $S_1 - S_2$ ’という引力制約が指定されているときには、文 $S_1$ と文 $S_2$ が近接しているようなテキストが優先される。

### 2.3.2 統語構造生成サブシステムの処理概要

統語構造生成サブシステムは、立論サブシステムの出力であるテキスト構造からテキスト全体の統語的な構造(文の順序、接続表現)を決定する。

一般に、1つのFTSからは複数の文章を生成することが可能であるが、語用論的な観点からみたと、それらの文章の好ましきは様々である。そこで本サブシステムでは、これらの生成可能な文章のうちから、最も好ましいと考えられる文章を、語用論的な評価基準を用いて選択する。

本サブシステムでは、以下のような処理を行い、表層文生成サブシステムの入力である統語的構造化意味表現を生成する。

1. 与えられたFTSから生成可能なすべての接続木を生成する。接続木とは、テキストの構造を木構造で表現したものであり、文の順序、重文化、接続詞などに関する情報を含んでいる。
2. 前段で求められた接続木のそれぞれを語用論的に評価し、それらを評価値によってソートする。

ここで用いられる評価値の計算は、さまざまな評価基準による評価結果の重みつき総和によって行なう。このときの評価基準には次のようなものがある。

- 記憶のスタックの深さ
- テキスト中に含まれる亀裂(逆茂木)の数
- テキストとFTS項の間の形態的類似性
- 文間の引力

3. 前段においてもっとも評価値が優れていた接続木の統語的構造化意味表現を生成し、表層文生成サブシステムに渡す。

### 2.3.3 統語的構造化意味表現と表層文生成サブシステム

表層文生成サブシステムでは、文の意味的な情報と統語構造とを統合した表現から表層表現を生成する。この意味表現を統語的構造化意味表現と呼ぶ[5]。統語的構造化意味表現は5つの階層を用いて1つの文を表現し、生成処理はこの階層に基づいて行われる。5つの階層の内容を以下に示す。

#### 第1層 基本概念の記述

#### 第2層 基本概念間の関係の記述

#### 第3層 基本概念間の関係が成立する時相状況と、時相状況に対する話し手の判断の記述

#### 第4層 時相状況間の関係の記述

#### 第5層 話し手が着目する対象とその着目対象についての記述内容の組と、それらと談話構造との関係の記述

第1層は基本的な概念の言語的な知識の記述であり、いわゆる辞書に相当する。第2層は基本概念間の関係を表す項の記述である。

第3層で記述する時相状況とは、あることがらを時間的視野・視点によって時間的に相対化した状況として見ることによりテンス・アスペクトの情報抜き取りのために取り入れた概念である。第3層で、第2層の項が成立する時相状況を考えることにより、テンス・アスペクトなどの時間的表現が生じる。

第4層では時相状況間の関係を記述する。提題の「は」や提題の名詞句などは、第5層で着目対象として記述する。着目対象とは話し手の注意が置かれている対象であり、その着目対象についての記述の表現を記述内容と呼ぶ。第4層の時相状況間の関係の記述は、1つの着目対象についての記述の範囲内にあると考える。このことは提題の「は」や提題の名詞句などが、個々の時相状況を越えて文全体にかかることに対応する。

## 3 「信念」と立論サブシステム

立論サブシステムは、2.1節で述べた様相的オペレータを持つ立論ゴールに対して、「信念」内の規則を適用することにより、立論ゴールの正当性を主張するテキスト構造(FTS)を生成する。

### 3.1 信念の内容

#### 3.1.1 信念のモジュール構成

信念には立論開始前にあらかじめ与えられている固定的な信念と、立論過程において生成されていく動的な信念とがある。

固定的な信念としては、fact、rule、judgementの3種類がある。factはシステムが既に生起した(している)ことを知っている事態の集合である。ruleは、推論に用いられる規則の集合であり、事態の間の因果関係が記述されている。judgementは、システムが望ましい(あるいは望ましくない)と考える事態の集合であり、goodとno-goodの2種類に類別される。3.1.2節で述べるように、judgementには条件付きでgood/no-goodとなるものもあり、その場合は規則として記述される。立論サブシステムはこのjudgementを用いることで、3.2節で述べるような推論を行い、様相オペレータ付き立論ゴールの説明テキストを生成する。

動的な信念にはpredict、assume、qfactの3種類がある。predictは3.1.2節で述べる予測型規則の前向き適用によって生成される結論であり、いわば「予測される事態」の集合である。assume/qfactは、ある予測における前提である。この前提が証明された場合(後ろ向き推論によってfactに到達する場合)にはqfactになり、証明されない(できない)場合はassumeとなる。即ち、qfactは「事実から導出される事態」の集合であり、assumeは「予測の過程で置かれた仮定」の集合である。

#### 3.1.2 信念内の規則

3.1.1節で述べたように、信念内の規則としては、事態間の因果関係を記述したもの(rule)と、ある事態が望ましくなる(望ましくなくなる)条件を記述したもの(judgement)とがある。ruleはさらに予測型規則(cause)と前提型規則(precond)の2種類に類別できる。

予測型規則は、ある原因となる事態群と、その事態群の結果となる事態を記述したものである。

$$P_1 \wedge P_2 \wedge \dots \Rightarrow Q \\ (\text{cause}([P_1, P_2, \dots], Q))$$

例えば、「駐車場の不足  $\wedge$  自動車の使用  $\wedge$  ...  $\Rightarrow$  違法駐車」のような規則であり、これは「駐車場が不足し、かつ、自動車を使用し、かつ... のときに、違法駐車をする」という意味である。

前提型規則はある事態とその事態の前提である事態群を記述したものである。

$$Q \Rightarrow P_1 \wedge P_2 \wedge \dots \\ (\text{precond}([P_1, P_2, \dots], Q))$$

例えば、「自動車の使用  $\Rightarrow$  道路の整備  $\wedge$  石油の使用  $\wedge$  ...」のような規則であり、これは「自動車を使用するためには、道路の整備が必要であり、かつ、石油を使用することが前提であり、...」という意味である。

3.3節で述べるように、立論の過程の予測においては主に予測型規則が用いられる。前提型規則は3.4節で述べるような特殊な使い方をする。

判断型規則(judgement)は次のような形式をしている。

$$(P_1 \in M_1) \wedge (P_2 \in M_2) \wedge \dots \Rightarrow (Q \in M) \\ (\text{judgement}([M_1(P_1), M_2(P_2), \dots], M(Q)))$$

ここで $M, M_1, M_2, \dots$ はqfact、good、no-goodのいずれかであり、左辺が成立したとき右辺が成立し、 $Q$ は望ましい( $\in$  good)か望ましくない( $\in$  no-good)事態となる。右辺が常に成立する場合は左辺は空である。例えば、「(有害( $X \in$  qfact)  $\wedge$  ...  $\Rightarrow$  (放出( $X \in$  no-good))」という規則は、「ある $X$ が有害であり、かつ、... のとき、 $X$ を放出するのは望ましくない」という意味である。3.3節で述べるように、判断型規則は予測の停止条件として用いられる。

#### 3.1.3 事態間の時間関係の記述

factや規則中の事態の間には時間関係が記述される。本システムにおいては、2つの事態 $S_1, S_2$ の間の時間関係として、t\_precとincludeの2種類を取り扱っている。

t\_prec( $S_1, S_2$ )は2つの事態の間に時間的な前後関係が成り立っているものであり、先行する事態 $S_1$ が終了してから後続の事態 $S_2$ が生起する。include( $S_1, S_2$ )は、2つの事態の間に包含関係がある場合で、事態 $S_2$ は事態 $S_1$ が生起している間に生起する。

### 3.2 様相表現のモデル化

システムは事態の真偽の他に、その事態の内容に関する価値基準も持つことができる。現在のところ、以下の二つの価値基準を持っている。

- $A \in$  good ... 事態 $A$ は望ましいことである
- $A \in$  no-good ... 事態 $A$ は望ましくないことである

「事態 $A$ が望ましい」という信念を持っている場合、 $A \in$  goodは真になる。逆に「 $A$ が好ましくない」という信念を持っている場合は、 $A \in$  no-goodが真になる。

ある事態 $A$ に対して、これらの価値基準のオペレータと否定のオペレータを組み合わせることで、以下の8通りの信念が表現可能である。

- goodに関するもの:  
 $A \in$  good,  $\neg A \in$  good,  
 $\neg(A \in$  good),  $\neg(\neg A \in$  good)
- no-goodに関するもの:  
 $A \in$  no-good,  $\neg A \in$  no-good,  
 $\neg(A \in$  no-good),  $\neg(\neg A \in$  no-good)

これらの信念の表現のうち、表1に掲げる6種類は自然言語の様相表現に自然に対応させられる。

### 3.3 推論の方式

ある事態に対するシステムの価値判断(good, no-good)は、システムの信念から推論することができる。ここでは、ある命題 $A$ に対する価値判断(good, no-good)を決定する手続きを説明する。われわれの計算モデルの基礎になっているのは以下の推論規則である。

信念の表現	様相表現	表層表現
$\neg A \in \text{no\_good}$	$\text{must}(A)$	Aしなければならぬ
$A \in \text{no\_good}$	$\text{must}(\neg A)$	Aしてはならぬ
$A \in \text{good}$	$\text{hb}(A)$	Aした方がよい
$\neg A \in \text{good}$	$\text{hb}(\neg A)$	Aしない方がよい
$\neg(A \in \text{no\_good})$	$\text{may}(A)$	Aしてもよい
$\neg(\neg A \in \text{no\_good})$	$\text{may}(\neg A)$	Aしなくてもよい

表 1: 信念と様相表現の対応

$B \Rightarrow A$	$B \Rightarrow A$
$A \in \text{good}$	$A \in \text{no\_good}$
$B \in \text{good}$	$B \in \text{no\_good}$

これらの推論規則は、それぞれ以下の内容を表している。

- 「 $B \Rightarrow A$  という予測型の規則があるとき、 $A$  が good と判断されたら、 $B$  も good と判断してよい」
- 「 $B \Rightarrow A$  という予測型の規則があるとき、 $A$  が no-good と判断されたら、 $B$  も no-good と判断してよい」

これらの推論規則を基にすると、与えられた事象  $A$  の価値判断を決定するアルゴリズムは以下のようになる。

$prove\_g(A)$  { $A$  が good であることの推論手続き }

1. ( $A \in \text{good}$ )  $\leftarrow P_1, P_2, \dots, P_n$  という判断型ルールがあるとき:  $prove\_goals([P_1, P_2, \dots, P_n])$
2.  $B \leftarrow P_1, P_2, \dots, P_n, A$  という予測型ルールがあるとき:
  - (a)  $prove\_goals([P_1, P_2, \dots, P_n])$
  - (b)  $prove\_g(B)$

$prove\_ng(A)$  { $A$  が no-good であることの推論手続き }

1. ( $A \in \text{no\_good}$ )  $\leftarrow P_1, P_2, \dots, P_n$  というルールがあるとき:  $prove\_goals([P_1, P_2, \dots, P_n])$
2.  $B \leftarrow P_1, P_2, \dots, P_n, A$  というルールがあるとき:
  - (a)  $prove\_goals([P_1, P_2, \dots, P_n])$
  - (b)  $prove\_ng(B)$

$prove\_goals(\text{ゴール列})$

1. ゴール列が空のとき: 成功
2. ゴール列が  $[G_1, G_2, \dots, G_n]$  のとき:
  - (a)  $prove(G_1)$  または  $prove\_g(G_1)$
  - (b)  $prove\_goals([G_2, G_3, \dots, G_n])$

$prove(G)$

$G \leftarrow G_1, G_2, \dots, G_n$  というルールがあるとき:  
 $prove\_goals([G_1, G_2, \dots, G_n])$

### 3.4 推論過程の立論テキストへの反映

#### 3.4.1 推論過程のテキスト構造 (FTS) への反映

立論のゴールとして  $\text{must\_not}(P)$  が与えられた場合を考える。更に、rule としては予測型規則のみを考える。3.3 節で述べたように、立論サブシステムは  $P \in \text{no\_good}$  を導出して、その導出過程から立論のテキスト構造を構成する。即ち、 $P$  から  $Q \in \text{no\_good}$  なる  $Q$  が導出されるまで rule (cause) の前向き適用を繰り返す。例えば、以下のような規則が適用されて、1 ステップの前向き推論で  $P \in \text{no\_good}$  が導出されたとする。

$$\begin{array}{l} P \wedge P_a \Rightarrow Q \\ Q \in \text{no\_good} \end{array} \quad \begin{array}{l} (r_1) \\ (j_1) \end{array}$$

ここで、 $P_a$  は後向き推論によって fact から証明できたものとし ( $P_a \in \text{qfact}$ )、そのテキスト構造を  $TS(P_a)$  とすると、この導出過程を表すテキスト構造は、 $(r_1)$  の成功と  $(j_1)$  の存在に対応した 2 つの reason を持つ構造として、次のように FTS で記述される。

$$\begin{array}{l} \{t/P \in \text{no\_good}, \\ r/\{\{t/P \Rightarrow Q, r/TS(P_a)\}, \\ \{t/Q \in \text{no\_good}\}\} \end{array} \quad (f_1)$$

ただし、2.3.1 節で述べた FTS の thesis, reason, anti.t という要素は、ここでは  $\{t/\langle \text{thesis} \rangle, r/\langle \text{reason} \rangle, a/\langle \text{anti.t} \rangle\}$  のような記法で表現している。

次に前向き推論が 1 ステップ長くなった以下のような場合を考える。

$$\begin{array}{l} P \wedge P_a \Rightarrow Q \\ Q \wedge Q_a \Rightarrow R \\ R \in \text{no\_good} \end{array} \quad \begin{array}{l} (r_1) \\ (r_2) \\ (j_2) \end{array}$$

ここでは、 $P$  に対して規則  $(r_1)$  が適用されることによって、まず  $Q$  が導出され、その  $Q$  に対して規則  $(r_2)$  が適用されて望ましくない事象  $R$  が導出される。この導出過程のテキスト構造は  $(f_1)$  をより複雑にしたものになるが、その際、新たに付け加わったステップを  $(f_1)$  のどこに付加するかによって、以下の  $(f_2)$ 、 $(f_3)$  の 2 通りが考えられる。 $(f_2)$  は 1 つの前向き推論のステップごとにテキスト構造を構成するものであり、これに対して  $(f_3)$  は、前向き推論の出発点  $P$  からの到達可能パスとしてテキスト構造を生成するものである。

$$\begin{array}{l} \{t/P \in \text{no\_good}, \\ r/\{\{t/P \Rightarrow Q, \\ r/TS(P_a)\}, \\ \{t/Q \in \text{no\_good}, \\ r/\{\{t/Q \Rightarrow R, \\ r/TS(Q_a)\}, \\ \{t/R \in \text{no\_good}\}\}\} \end{array} \quad (f_2)$$

$$\begin{array}{l} \{t/P \in \text{no\_good}, \\ r/\{\{t/P \Rightarrow R, \\ r/\{\{t/P \Rightarrow Q, \\ r/TS(P_a)\}, \\ \{t/TS(Q_a)\}\}, \\ \{t/R \in \text{no\_good}\}\} \end{array} \quad (f_3)$$

- (f<sub>2</sub>) と (f<sub>3</sub>) は論理的には等価であるが、(f<sub>2</sub>) は
- 中間の事態 (Q) の結論 (Q ∈ no-good) を構成する
  - 各ステップごとのテキスト構造が P と無関係

であることから、must\_not(P) の立論のテキスト構造としては、わかりにくいものになると考えられる。従って、本立論サブシステムの出力としては (f<sub>3</sub>) を採用する。前向き推論のステップがより長くなった場合も、同様にテキスト構造を構成する。

規則の前向き適用において、その前提となる事態 ((r<sub>1</sub>) の P<sub>a</sub>, (r<sub>2</sub>) の Q<sub>a</sub>) は、原則として後向き推論によって fact から証明することになる。例えば、以下のような規則がある場合、

$$\begin{array}{ll} S_1 \wedge S_2 \Rightarrow P_a & (r_3) \\ S_3 \wedge S_4 \Rightarrow S_2 & (r_4) \\ S_1 \in \text{fact} & (\text{fa}_1) \\ S_3 \in \text{fact} & (\text{fa}_2) \\ S_4 \in \text{fact} & (\text{fa}_3) \end{array}$$

その後向き導出の過程から構成される TS(P<sub>a</sub>) というテキスト構造は、以下のような FTS で記述される。

$$\begin{array}{l} \{t/P_a, \\ r/\{t/S_1\}, \\ \{t/S_2, \\ r/\{t/S_3, \{t/S_4\}\}\} \} \end{array} \quad (f_4)$$

### 3.4.2 表層表現決定に利用されるその他の情報

立論サブシステムで行われる推論の過程は、3.4.1 節で述べたように立論のテキスト構造に反映されるが、その他に、立論の過程で得られる以下のような情報が、アスペクト表現などの立論テキストの表層表現決定のために利用できる。立論サブシステムは、これらの情報を信念や出力される FTS に書き込んで、後段の統語構造生成サブシステムや表層表現生成サブシステムが利用できるようにする。

#### (1) 規則の適用結果

3.1.1 節で述べたように、「信念」は複数のモジュールで構成されているが、モジュールの中で「動的」なもの (predict, qfact, assume) は、推論の進展に伴って書き込まれていく。即ち、前向き推論によって予測された事態は predict となり、前向き推論の過程における前提は assume となる。さらに、前提は後ろ向き推論によって証明されることによって qfact となる。例えば、

$$\begin{array}{l} \text{道路が狭い} \wedge \text{違法駐車} \Rightarrow \\ \text{交通渋滞が起きる} \end{array} \quad (r_3)$$

という規則を前向きに適用することによって、「違法駐車」という事態からの予測を表層表現として生成する場合を考える。ここで、「道路が狭い」という事態が assume であれば、次の FTS で記述されるテキスト構造が生成される。

$$\begin{array}{l} \{t/\text{道路が狭い} \Rightarrow \\ (\text{違法駐車} \Rightarrow \text{交通渋滞が起きる}) \} \end{array} \quad (f_5)$$

ここでは「道路が狭い」は条件として扱われている。一方、この事態が証明されて qfact となれば、テキスト構造は以下のようになり、「道路が狭い」は理由となる。

$$\begin{array}{l} \{t/\text{違法駐車} \Rightarrow \text{交通渋滞が起きる}, \\ r/\text{道路が狭い} \} \end{array} \quad (f_6)$$

また、「交通渋滞が起きる」という事態は predict であるが、その上望ましくない事態 (∈ no-good) であるとする、表層表現では「交通渋滞が起きてしまう」と生成されるべきであろう。結局、(f<sub>5</sub>) と (f<sub>6</sub>) に対応した表層表現は、例えば以下のようなものになる。

(f<sub>5</sub>) “道路が狭いと、違法駐車すれば交通渋滞が起きてしまう”

(f<sub>6</sub>) “道路が狭いので、違法駐車すれば交通渋滞が起きてしまう”

#### (2) 規則の種類 (cause と precondition)

3.1.2 節で述べたように、rule K は cause と precondition の 2 種類があるが、今まで取り上げてきた例 ((r<sub>1</sub>), (r<sub>2</sub>), (r<sub>3</sub>)) は全て cause である。3.2 節でも述べたように、precond は推論の過程で cause とは異なった用いられ方をする。即ち、precond は目標が望ましい事態 (∈ good) である予測の過程においてのみ後ろ向きに適用される。例えば、hb(A<sub>0</sub>) という立論ゴールが与えられた時、A<sub>0</sub> からの予測の過程で、「道路の整備」という事態に対して以下のような precondition が適用されたとする。

$$\begin{array}{l} \text{自動車の使用} \Rightarrow \\ \text{道路の整備} \wedge \text{石油の使用} \wedge \dots \end{array}$$

これにより生成されるテキスト構造は (f<sub>7</sub>) のようになり、その表層表現は例えば以下ようになる。

$$\begin{array}{l} \{t/\text{hb}(A_0), \\ r/\{t/A_0 \Rightarrow \text{道路の整備}, r/\dots\}, \\ \{t/\text{precond}(\text{道路の整備}, \\ \text{自動車の使用}), \\ \{t/\text{hb}(\text{自動車の使用}), r/\dots\} \} \} \end{array} \quad (f_7)$$

(f<sub>7</sub>) “A<sub>0</sub> すれば道路が整備される。また、道路が整備されれば、自動車を使うことができる。また、自動車は使った方がよい。なぜならば... したがって、A<sub>0</sub> した方がよい”

即ち、precond の適用によってもたらされた予測は、あくまで「その状態が可能になる」ということであり、cause の場合の予測とは異なる。従って、3.4.1 節で述べた構造 (「原因 ⇒ 結果」という cause の連鎖) が、いったん切断された構造となる。

#### (3) 時間関係 (t-prec と include)

3.1.3 節で述べたように、信念内に記述された事態間の時間関係は、テンス / アスペクト情報の生成に利用される。例えば、「自動車の使用」という事態からの予測において、

$$\begin{array}{l} \text{駐車場所の不足} \wedge \text{自動車の使用} \Rightarrow \\ \text{違法駐車} \end{array}$$

という規則が適用され、また「駐車場所の不足」という事態が fact であり、かつ現時点を含むような時間幅を持った事態である場合、「駐車場所が不足しているので、自動車を使うと違法駐車してしまう」という表層表現が生成される。

#### (4) 発話順序 (gravity と precedence)

(1)~(3)までは、個々の事態の表現に関する情報であったが、事態間の論理的な関係によって、それらの事態間の発話順序に関する制約が発生する場合がある。

例えば、「ディーゼル車は煤を放出する」という事態と「ディーゼル車はガソリン車より多くの煤を放出する」という事態を考えると、前者は後者に包含されている(前者は後者の前提である)。一方、1つのテキスト内で、先に発話されている事態から当然推測できることを、後で改めて発話するのは不自然である。従って、この2つの事態が1つのテキスト内で発話される場合は、必ず、前者を後者よりも先に発話しなくてはならない。また、次節で述べる反駁の場合のように、ある2つの事態の間に意味的に深い関係があり、表層テキスト中ではなるべく近接して発話した方がよい場合もある。

これらの発話順序に関する制約は、2.3節で述べた FTS 中の素性(順序制約, 引力制約)として、統語構造生成サブシステムに与えられる。

#### 3.4.3 補強と反駁

3.4.2節で述べたように、立論における予測の前提は、証明されない(できない)場合、仮定 (assume) として残されるが、そのような部分は立論テキストとしては弱い部分と言えるので補強した方がよい。即ち、その事態  $P$  に関して、 $\text{must}(P)$  や  $\text{hb}(P)$  の立論を行う。例えば、 $(f_8)$  のようなテキスト構造において、 $\text{must}(P)$  の立論が成功した場合、 $(f'_8)$  のテキスト構造が得られる。

$$\{t/P \Rightarrow (A_0 \Rightarrow A)\} \quad (f_8)$$

$$\{t/A_0 \Rightarrow A, r/\{t/\text{must}(P), r/\dots\}\} \quad (f'_8)$$

反駁はある立論テキストに対する異議の表明であるが、どのように反駁を行うかによって以下の3種類がある。

##### (1) 強い反駁 1

これは、反駁対象の立論テキストの主張と反対の主張:

hb に対しては  $\text{must\_not}$ ,  
hb\_not に対しては  $\text{must}$ ,  
 $\text{must}$  に対しては  $\text{hb\_not}$ ,  
 $\text{must\_not}$  に対しては  $\text{hb}$

を行うものである。具体的には、反駁対象の立論のテキスト構造中の predict から、その立論とは逆の目標:

good に対しては  $\text{no\_good}$ ,  
 $\text{no\_good}$  に対しては  $\text{good}$

に向かう予測パスが構成できればよい。例えば、 $(f_9)$  は  $\text{hb}(A_0)$  を主張する立論のテキスト構造であるが、これに対する強い反駁は  $(f'_9)$  のようになる。即ち、 $(f_9)$  中の予測パス中の事態  $A_1$  から  $E'$  ( $\in \text{no\_good}$ ) なる事態を予測して、 $\text{hb}(A_0)$  とは正反対の結論である  $\text{must\_not}(A_0)$  を結論としている。反駁の効果を上げるためには、 $E$  の望ましい程度に比べて、甚だしく望ましくない  $E'$  を選ぶ。

$$\begin{aligned} &\{t/\text{hb}(A_0), \\ &r/\{\{t/A_0 \Rightarrow E, \\ &r/\{\dots\{t/A_0 \Rightarrow A_1, r/\dots\}\dots\}\}, \\ &\{t/E \in \text{good}\}\} \end{aligned} \quad (f_9)$$

$$\begin{aligned} &\{t/\text{must\_not}(A_0), \\ &r/\{\{t/A_1 \Rightarrow E', r/\dots\}, \\ &\{t/E' \in \text{no\_good}\}\}, \\ &a/\{t/\text{seem\_hb}(A_0), \\ &r/\{\{t/A_0 \Rightarrow E, \\ &r/\{\dots\{t/A_0 \Rightarrow A_1, r/\dots\}\dots\}\}, \\ &\{t/E \in \text{good}\}\} \end{aligned} \quad (f'_9)$$

“自動車を使えば輸送に便利であるから、自動車は使った方がよいように見える。しかし、自動車を使うと排気ガスを放出するので大気汚染が進み、環境が悪化する。従って、自動車は使うべきではない”

##### (2) 強い反駁 2

(1)と同様、反駁対象の立論テキストと反対の主張:

hb に対して  $\text{hb\_not}$ ,  
hb\_not に対して  $\text{hb}$ ,  
 $\text{must}$  に対して  $\text{must\_not}$ ,  
 $\text{must\_not}$  に対して  $\text{must}$

を行うものである。具体的には反駁対象の立論のテキスト構造中の predict の negation から、その立論と同じ目標 (good または  $\text{no\_good}$ ) が予測できればよい。以下の  $(f_{10})$  は  $\text{must\_not}(A_0)$  を主張するものであるが、それに対する反駁 2 である  $(f'_{10})$  によって  $\text{must}(A_0)$  を主張している。(1)と比較した場合、 $\text{must}$  と  $\text{must\_not}$  についてはより強い反駁となっており、 $\text{hb}$  と  $\text{hb\_not}$  についてはより弱い反駁になっている。(1)と同様に、 $E \in \text{no\_good}$  より  $E' \in \text{no\_good}$  の程度が強いのが必要である。

$$\begin{aligned} &\{t/\text{must\_not}(A_0), \\ &r/\{\{t/A_0 \Rightarrow E, \\ &r/\{\dots\{t/A_0 \Rightarrow A_1, r/\dots\}\dots\}\}, \\ &\{t/E \in \text{no\_good}\}\} \end{aligned} \quad (f_{10})$$

$$\begin{aligned} &\{t/\text{must}(A_0), \\ &r/\{\{t/\neg A_1 \Rightarrow E', r/\dots\}, \\ &\{t/E' \in \text{no\_good}\}\}, \\ &a/\{t/\text{seem\_must\_not}(A_0), \\ &r/\{\{t/A_0 \Rightarrow E, \\ &r/\{\dots\{t/A_0 \Rightarrow A_1, r/\dots\}\dots\}\}, \\ &\{t/E \in \text{no\_good}\}\} \end{aligned} \quad (f'_{10})$$

“自動車を使うと環境が悪化するから自動車は使うべきではないように見える。しかし、自動車を使わなければ、輸送コストが増加して物価が増加する。従って、自動車は使わねばならない”

### (3) 弱い反駁

(1) や (2) とは異なり、反対の主張をするのではなく、反駁対象の立論テキストが論理的に成立しないことを主張するものである。反駁対象の様相が *must* と *must\_not* の場合は、3.2 節でも述べたように、反駁の様相表現はそれぞれ *may* と *may\_not* となる。これは、反駁対象のテキスト構造中の *assume* に対して、その *negation* を証明する (*qfact* であることを示す) ことによって行う。以下の  $(f_{11})$  では、 $\text{must\_not}(A_0)$  の中に  $A_2$  という *assume* がある。 $(f'_{11})$  では  $\neg A_2$  を証明することによって、 $(f_{11})$  に対する反駁を行っている。なお、*assume* は予測の前提として反駁対象のテキスト構造に直接現れていなくても、予測される *predict* に対して *precond* を前向きに適用することによって導かれたものでもよい ( $(f''_{11})$ )。

$$\begin{aligned} & \{t/\text{must\_not}(A_0), \\ & r/\{\{t/A_0 \Rightarrow E, \\ & \quad r/\{\dots\{t/A_2 \Rightarrow (A_0 \Rightarrow A_1), \quad (f_{11}) \\ & \quad \quad r/\dots\}\dots\}, \\ & \quad \{t/E \in \text{no\_good}\}\}\} \end{aligned}$$

$$\begin{aligned} & \{t/\text{may}(A_0), \\ & r/\{t/\neg A_2, r/\dots\}, \\ & a/\{t/\text{seem\_must\_not}(A_0), \\ & \quad r/\{\{t/A_0 \Rightarrow E, \quad (f'_{11}) \\ & \quad \quad r/\{\dots\{t/A_2 \Rightarrow (A_0 \Rightarrow A_1), \\ & \quad \quad \quad r/\dots\}\}\}, \\ & \quad \{t/E \in \text{no\_good}\}\}\} \end{aligned}$$

$$\begin{aligned} & \{t/\text{may}(A_0), \\ & r/\{t/\neg A_3, r/\dots\}, \\ & a/\{t/\text{seem\_must\_not}(A_0), \\ & \quad r/\{\{t/A_0 \Rightarrow E, \\ & \quad \quad r/\{\dots\{t/A_2 \Rightarrow (A_0 \Rightarrow A_1), \quad (f''_{11}) \\ & \quad \quad \quad r/\dots\}\}\}, \\ & \quad \{t/E \in \text{no\_good}\}\}\} \end{aligned}$$

where  $\text{precond}(A_3 \wedge \dots, A_1)$

“自動車を使うと環境を悪化させるので、自動車を使用してはならないように見える。なぜならば、排気ガスが放出されるからである。しかし、自動車の排気ガスは排気ガス規制で抑えられている。従って、自動車を使用してもよい”

反駁のテキスト構造の *reason* は、反駁対象のテキスト構造中のある事態 (またはその *negation*) に対する別の予測や証明から構成されている。従って、3.4.2 節でも述べたように、反駁の *reason* と反駁の起点となった事態は、表層テキスト中ではできるだけ近接して生成されるように FTS に指定する。

## 4 動作例

DULCINEA は、現在、逐次型推論マシン PSI の上で開発中である。本章では、DULCINEA の実際の動作例を示す。

与えられる立論ゴールは、「違法駐車を厳罰に処してはならない」というものである。ただし、「信念」には以下のよう規則と事態が含まれているものとする。

```
rule::
  cause(使用 (agt=X, obj= 自動車)
        ^ 不足 (obj2= 駐車スペース),
        違法駐車 (agt=X)).
  cause(使用 (obj= 自動車, agt=X),
        便利 (for= 輸送, obj2= 自動車, pat=X)).

fact::
  不足 (obj2= 駐車スペース).

good::
  便利 (for=X, obj2=Y, pat= 多くの人).

no-good::
  厳罰に処する (obj= 多くの人).
```

立論サブシステムには、*QUIXOTE* のオブジェクトとして立論ゴールが与えられる。今の例では、このゴールは次のように表される:

$$\begin{aligned} & \text{must\_not}(\text{cont}=\text{cause}(1=\text{違法駐車 (agt=X)}, \\ & \quad 2=\text{厳罰に処する (obj=X)})) \end{aligned}$$

この例の場合、立論は以下のように行われる:

この立論ゴールの内容には、「違法駐車 (agt=X)  $\Rightarrow$  厳罰に処する (obj=X)」という関係が含まれているので、「違法駐車 (agt=X)」という事態が証明されれば、そこから「厳罰に処する (obj=X)」という事態が予測されることになる。従って、立論サブシステムはまず、「違法駐車 (agt=X)」という事態から後向き推論を行い、この事態の証明を試みる。この場合、その証明は失敗し、「自動車の使用」という事態が仮定 (*assume*) として残される。そのため、立論サブシステムはこの部分の補強を行うために、「自動車の使用」という事態から前向き推論による予測を行い、「多くの人が便利である」という事態に達して成功する。従って、「多くの人が自動車を使用したほうがよい」という主張が生成され、「違法駐車 (agt= 多くの人)」という事態が十分予測されることになる。結局、立論サブシステムから出力される FTS は次のようなものになる。

$$\begin{aligned} & \{t/\text{must\_not}(\text{cont}=\text{cause}(1=\text{m\_s1}(\text{cont}=I_1), \\ & \quad 2=\text{m\_s2}(\text{cont}=P_1))), \\ & r/\{t/\text{cause}(1=\text{m\_s1}(\text{cont}=\text{cause}(1=\text{m\_s1}(\text{cont}=I_1), \\ & \quad 2=\text{m\_s2}(\text{cont}=P_1))), \\ & \quad 2=\text{m\_s2}(\text{cont}=P_2)), \\ & \quad r/\{\{t/\text{cause}(1=\text{m\_s1}(\text{cont}=U_1), \\ & \quad 2=\text{m\_s2}(\text{cont}=I_2)), \\ & \quad \quad r/\{t/S_1\}\}, \\ & \quad \{t/\text{hb}(\text{cont}=U_1), \\ & \quad r/\{\{t/\text{cause}(1=\text{m\_s1}(\text{cont}=U_1), \\ & \quad 2=\text{m\_s2}(\text{cont}=C_1)\}\}\}\} \end{aligned}$$

ただし、上の構造中に現れる  $I_1, I_2$  などの変数の内容は以下のとおりである:

$$\begin{aligned} I_1 &= \text{違法駐車 (pol=1)} \\ I_2 &= \text{違法駐車 (agt= 多くの人, pol=1)} \\ P_1 &= \text{厳罰に処する (pol=1)} \end{aligned}$$

$P_2 =$  厳罰に処する (obj= 多くの人, pol=1)  
 $U_1 =$  使用 (agt= 多くの人, obj= 自動車, pol=1)  
 $C_1 =$  便利 (for= 輸送, obj2= 自動車,  
           pat= 多くの人, pol=1)  
 $S_1 =$  不足 (obj2= 駐車スペース, pol=1)

統語構造生成サブシステムは、この FTS を受け取って文の順序と文間の接続表現を決定し、表層テキストの統語構造を生成する。その際、2.3 節で述べたような評価関数を用いて、可能な統語構造の中から最尤のものを選ぶ。この例の場合、可能な統語構造は 138 個ある。更に、統語構造生成サブシステムは各文の attention や視点を決定して、提題化や態の決定を行い、最終的に各文の統語的構造化意味表現を構成する。以下に、統語的構造化意味表現の例を示す。これは、統語構造生成サブシステムが選んだ最尤テキストの第 1 文に対応したものである。

```

m_content(id=1, sid=1)::
  S1.
m_sb::
  include(s1=m_content(id=1, sid=1), s2=m_now).
m_content(id=1, sid=2)::
  cause(1=m_s1(cont=U1),
        2=m_s2(cont=I2)).
m_gen(id=1)::
  top(1=top(attention= 駐車スペース,
             cont=m_content(id=1, sid=1)),
      2=top(cont=m_content(id=1, sid=2),
            connective= から)).
  
```

表層文生成サブシステムは、各文ごとの統語的構造化意味表現から、2.4 節で述べたような各階層ごとの処理を行い、順次表層文を生成していく。この例で最終的に生成される表層テキストを以下に示す。

“駐車スペースは不足しているから、多くの人々が自動車を使用するれば、違法駐車してしまう。また、多くの人にとって自動車が輸送に便利だから、多くの人々は自動車を使用するほうがよい。したがって、違法駐車で厳罰に処すれば、多くの人々が厳罰に処されてしまう。したがって、違法駐車で厳罰に処してはならない”

## 5 おわりに

本稿では、与えられた立論ゴールに対して、その正当性を主張するテキスト(立論テキスト)の内容を、信念に基づいた推論により決定する方式について述べた。立論ゴールは事態に様相的なオペレータの付いたものであるが、その内容から信念に含まれる判断(judgement)に達するまで、前向き推論による予測と後ろ向き推論による証明を行うことで立論ゴールの正当性を論証し、その推論過程から立論のテキスト構造を構成する。テキスト構造は、文間の機能的な関係を記述する FTS を用いて表現され、基本的に推論過程はそのような機能的な関係にマッピングされるが、その他に、アスペクト情報など文そのものの表層表現を決定する情報や、文の発話順序についての制約が得られる場合もある。本

稿では、さらにある主張に対する補強と反駁の方法についても検討した。

今後の課題としては以下のものが挙げられる。

- 複数の立論が可能な場合  
一般に、ある立論ゴールに対する立論テキストは複数存在しているであろう。そのような場合に生成する立論テキストを選択する評価基準について検討する必要がある。評価基準においては聞き手についての知識も必要であろう。
- 推論ステップの省略  
現行のシステムでは、推論ステップは全てテキスト構造に記述される。しかし、よく知られている規則が適用された場合など、発話するとかえってくどくなることもあるだろう。これについても、前項と同様聞き手が何を知っているか、という聞き手についての知識が必要である。

## 参考文献

- [1] K. R. McKeown, Discourse Strategies for Generating Natural Language Text, *Artificial Intelligence*, 27-1985, pp.1-41, 1985.
- [2] W. C. Mann, Text Generation: The Problem of Text Structure, In D. D. McDonald and L. Bolc, editors, *Natural Language Generation Systems*, Springer-Verlag Publishers, pages 47-68, 1988.
- [3] H. Yasukawa and K. Yokota, Labelled Graphs as Semantics of Objects, 情報処理学会, データベース研究会資料, DB-80-13, 1990.
- [4] 小谷, 今村, 近藤: 文章の機能的構造からの接続表現生成, 情報処理学会, 自然言語研究会資料, 1990.
- [5] 池田光生: 意味表現の階層に基づく生成システム, 電子情報通信学会, 日本ソフトウェア科学会, 「自然言語処理における統合」シンポジウム論文集, 1991.