

TM-0470

文脈理解のための知識の表現と推論

木下 聡, 佐野 洋, 浮田輝彦, 住田一男
大野真家

March, 1988

©1988. ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03) 456-3191-5
Telex ICOT J32964

Institute for New Generation Computer Technology

文脈理解のための知識の表現と推論[†]

木下 聡・佐野 洋・浮田輝彦・住田一男・天野真家
柳東芝 総合研究所

1 はじめに

文章を入力してその内容に対する質問に答えるシステムや、対話によってユーザと協調して問題解決を行っていくシステム、代名詞や省略などを含む文も正しく翻訳できる機械翻訳システムなどを考えた場合、それらにおいて中心となる技術は文脈処理技術であり、日本でも近年、文脈理解をテーマとした研究が活発に行われている【向井85、石崎86、伊東87】。

文脈処理をする上での最も基本的な問題は文中の代名詞や省略の処理と言っても過言ではなからう、Hobbs らによれば、英語の文の解析では、シンタクスによる解析だけでテキストでは90%（会話では70%）の同定を行うことが可能であるとの報告もある【Hobbs79】。しかし、裏を返せば、残りの処理を行うには、なんらかの意味的な、もしくは知識に基づいた処理を行う必要があるわけで、文脈処理においては、文法規則など言語的知識に加え、文の表現対象となる世界に関する知識（非言語的知識）とその知識を使った推論の重要性が指摘されている。

例えば、Schankらは、文を理解するうえで、スクリプトと呼ばれる典型的なイベントの系列に関する知識を用いて文の解析を行った【Schank77】。しかし、このスクリプトに基づいた文の解析は、あらかじめ用意された典型例にうまくあてはまる場合にしか解析できない、柔軟性に欠ける点を補うために、Wilenskyらは、ゴールとプランというレベルで文と文の間の関係を揃えることで、文の理解を行っている【Wilensky83】。これらの研究では、文と文との間の関係を調べ、全体として文がどのように関係し、接続されているかを認識するかが主眼となっている。もちろん、この様なレベルの理解は、対話においては焦点を把握した一貫性のある対話を行うことを可能とする。

現在我々は、文脈理解の研究の一環として、ユーザとの対話により機器の操作法のコンサルテーションを行う対話システムを開発中である【浮田87】。対話システムでは、扱う文も主として会話文となるが、照応参照を扱うメカニズムは文章理解システムと基本的に同じであり、むしろ省略などが多用される点で、文脈への依存性も大きいと思われる。

本論文では、まず質問応答システムにおいてユーザの発話を理解するために行われる談話処理のための知識の表現と推論について考察する。その後現在開発中のシステムにおける知識処理系と応用例として対話システムの概略について報告する。

[†] 本研究はICOTの委託研究として行っている

2 文脈理解のための知識の表現と推論

知識表現に関しては、知識をどのような形式で表現するかといった表現形式の問題や、時間、限量などをどのように表現するかなど様々な議論がなされている。ここでは、文脈理解に必要な知識の分類と推論について述べる。

2.1 文脈理解のための知識

文脈理解のための「知識」について考える前に「文脈理解」についてふれておきたい。文脈理解には2つの意味があると考えられる。1つは文脈そのものを理解すること、すなわち、文章を読んで文章全体でどのようなことがらを述べているかを把握したり会話かどのような話題のもとに行なわれているかを把握したりすることである。もう1つは、個々の文を文脈の中で理解すること、すなわち、文中の代名詞や省略などを文脈から補って文の意味を把握することである。これらの2つの考え方は相反するものではない、個々の文が理解できなければ文章全体を正確に理解できない場合もあろうし、ある特定の文がトリガーとなって文脈を描えることができる場合もあろう。また、逆に、個々の文に曖昧さがある解釈を一意に決定できない場合でも文脈によって決定できる場合もあろう。会話文では、発話は文脈への依存性が大きいので、特に後者が重要である。ここでは、文脈理解を後者に重点を置いて考えていく。

さて文を理解するための知識は言語における文法などの言語的知識と、我々を取りまく世界に関する知識である非言語的知識に分かれると考えられる。前者の重要性をここで改めて議論する必要はないであろう。以下では後者を指して単に知識と呼ぶ。

我々は、文脈理解に必要な知識として、対象とする世界における知識を4種類に分けて考えている。

- ・オブジェクトに関する知識
- ・イベントに関する知識
- ・イベントとイベントの関係
- ・概念体系（シソーラス）

①オブジェクトに関する知識

世界に存在する有形物や無形物などの「もの」をここでは「オブジェクト」と呼ぶ。我々は文を読んだり聞いた時、個々の単語の意味は解るが、文全体の意味が理解できないということが多々ある。例えば、あるオブジェクトについてなんらかの事柄を述べている文がある場合、仮に対象がそのオブジェクトであると認識できたとしても、そのオブジェクトについて何を述べているかよく分からないことがある（もちろん、文の表現対象となっているオブジェクトがどれかを認識できない場合もある）。このような状況は、そのオブジェクトが本質的にどのよ

うな性質を持ちうるかといった知識が実際には不足しているからである。このようなことは、Minskyが物体の認識のためにフレームの概念を導入した理由と同じであると考えられる [Minsky75]。極端な言い方をすれば、我々は自分が知っているレベルでしか文を理解することが出来ないのである。したがって、文を理解するための知識としてまず第1に、オブジェクトがどのような属性を持ちうるかなどといったオブジェクトに関する知識が必要である。

② イベントに関する知識

オブジェクトが行う動作やオブジェクトに対して行われる動作などをここでは「イベント」と呼ぶ。文の表現対象となる世界においてどのようなイベントがあり、どのようなオブジェクトがその主体や対象となりうるかといった知識をイベントに関する知識として持つ必要がある。

③ イベントとイベントの関係

文脈理解での最も基本的な問題は、照応参照処理であるといっても決して過言ではなからう。それには、先行する文と現在処理を行っている文とのつながりを認識する必要がある。1でも触れたように、Schankらは、スクリプトと呼ばれるイベントの典型的な系列に関する知識を用いて文の解析を行った。しかし、このスクリプトに基づいた文の解析は、予め用意された典型例にうまくあてはまる場合にしか解析できない。この柔軟性に欠ける点を補うために、Wilenskyらは、ゴールとそれに対するプランという形でイベント間の関係を補えようとした。このように、通常我々はイベント間に成り立っている様々な関係についての知識を持っており、これを文脈理解に使用している。

④ 概念体系

次のような対話でのやりとりを考えてみよう。

「再生スイッチを押しましたか」

「そのスイッチは押しました」

会話での応答にしては少し不自然な答えかもしれないが、このように対話や文章の中では、同一のオブジェクトを示すのに異なる語が使用される場合が多い。もちろん、ビデオとVTRなどのように同一の概念を表すものもあるが、上の例のように、異なる概念（スイッチは再生スイッチの上位概念）である場合も多い。このような問題を扱うには、知識として概念体系を持っている必要がある。

2.2 文脈理解における推論

文脈理解において行われる推論にはどのようなものがあるかについては、Riegerや石崎らが現象面から分類を行っている [Rieger75, 石崎86]。ここではこの様な観点から推論を考えるのではなく、文脈処理をする上で生じる解釈の曖昧さをどう扱うかといった機能的な観点から推論について考えてみたい。

エキスパートシステムの分野では、TMSまたはATMSと呼ばれる、仮説に基づいた推論方式が提案されている [Doyle79, deKleer86]。この方式は、情報の不足などによる不確かな事実を正しいものと仮定をして推論を進め、矛盾が検出された場合に、推論の過程で行っ

てきた仮説を自動的に修正して推論を継続していく推論方式である。

自然言語理解の分野でもこのような推論方式を利用した研究が行われ始めている [劉87, 島津88]。自然言語理解では、もともと解析文に曖昧性が内在する場合や、照応参照処理を行っても解釈が一意に決定できない場合も多い。例えば、対話システムにおいてユーザが(1)のように発話したとする。

カセットをビデオに入れて再生スイッチを (1)
押したが、(φ)が動かない

仮にφの検出として考えられるものをカセット、ビデオ、再生スイッチとしよう。ユーザが発話した際のφはもちろんそれらのうちの1つであるが、解釈する側ではこれら3つの解釈を考えることが出来る。このように文を解釈する上で考えることが出来る世界をここでは「仮説的文脈世界（または単に文脈世界）」と呼ぶ(図1)。結局、このような問題に対する文脈処理の目的は、それらの文脈世界の中から最も適当と思われる世界を選択することに帰着される。そして、それ以降の文の解析は選択された文脈世界の上で行われる。

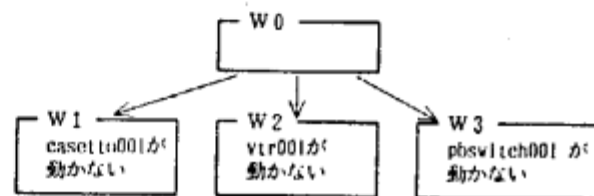


図1 ワールドを用いた談話処理

いずれにせよ、解析時点で最も適当と思われる解析結果を採用する必要があるが、対話を進めたり文章を読み進んで処理を続けていったある時点で矛盾が検出された場合、その矛盾が何によるものかを解析しそれに対応した処理を行うことができる枠組が必要である。

2.3 知識表現のアプローチ

知識表現の表現レベルには種々のものがあり、対象とするアプリケーションに依存する。ここでは機器操作のコンサルテーションを想定して以下のような知識処理を考えた。

(1) 知識表現の枠組

2.1で述べたように、文脈理解のための知識として①オブジェクトに関する知識、②イベントに関する知識、③イベントとイベントの関係、④概念体系などを表現しておかなければならない。知識を表現する枠組には意味ネットワークをはじめ様々な表現形式があり、それぞれの特徴について議論されている [例えば辻井87]。我々は概念体系の表現のしやすさ、管理の容易さ、論理的表現のしやすさなどを考慮して、これらの知識を「フレーム」と「ルール」を用いて表現する。

① オブジェクトに関する知識

オブジェクトに関する知識として次のような知識を表現する。

- (a) オブジェクトの属性と関係
- (b) 動的な性質
- (c) 属性間の論理的關係

(a) オブジェクトの属性と関係

オブジェクトの属性や他のオブジェクトとの関係はフレームのスロットで表現される。また、どのような属性や関係を持ちうるかといった知識は、個々のオブジェクトのクラスのフレームにスロットを設け、スロット値に対する制約条件を記述する。

(b) 動的な性質

オブジェクトが外部からの刺激を受けてどのように変化・反応するかといった知識、いわゆるモデルに相当する知識となる。オブジェクト指向の知識表現では、このような知識はオブジェクトにおけるメソッドとして手続きの形で記述されていたが、これらはルールで記述可能である。

(c) 属性間の論理的關係

(a) で述べた属性の間には論理的關係が成立しているものがある。たとえば、人間における「年齢」と「生年月日」の間には、論理的關係が存在する。このような知識もルールの形で記述する。

② イベントに関する知識

一つのイベントは深層的な格表現を基本として表現する。一般に格表現はそれぞれの格をスロットで表現することで全体をフレームとして表現できるため、フレームとの親和性が良い。各イベントにどのようなオブジェクトが関係するといった知識は、それぞれの格フィルアに対する制約条件であり、そのままスロット値に対する制約条件の形で記述する。

③ イベントとイベントの関係

イベントとイベントの間関係についての知識として我々が持っている主たる知識は、イベント間の因果関係である。因果関係はルールによって記述する。

④ 概念体系

概念体系は、従来よりフレームを用いた知識表現においてスロットを使って表現されており、ここでもこれを踏襲する。オブジェクトの性質を表現する際にも、上位概念からの知識の継承により、記述量を減らすことが可能であり、知識の保守性も向上する。

(2) 推論の枠組

2. 2で述べた処理を可能にするために、各文脈世界ごとにそれらに対応した推論用のワーキングメモリ（以下ワールドと呼ぶ）を設定する。このワールドを用いることで、個々の解釈に対応した文脈世界上で、その解釈結果をもとにさらに前向き推論などを行って、その結果から、文脈との関連性や無矛盾性などをチェックすることも可能となる。

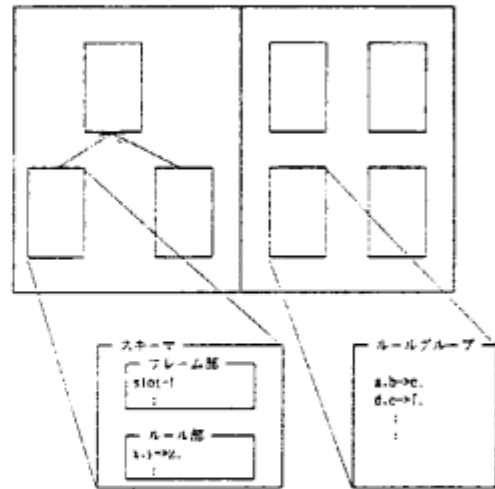


図2 知識ベースの構成

3 知識処理の設計と実現

現在我々は、4章で述べるように、ユーザとの対話によってビデオの操作法のコンサルテーションを行うシステムを開発中である。本章では、ユーザの発話を理解するためにシステムが持つ対象世界の知識の具体的な表現と推論の概略を述べる。

図2は、本処理系における知識ベースの構成を表している。先に述べたようなオブジェクトに関する知識やイベント間の因果関係などの知識は、スキーマとルールによって表現する。

3. 1 知識表現

(1) スキーマによる知識の表現

図3に、スキーマによるビデオの知識表現の一部を示す。スキーマはフレーム部とルール部に別れている。フレーム部には、上位-下位関係、部分-全体関係、属性関係（属性名-属性値）など静的な知識が記述されている。ルール部には、ビデオの動的な性質などが記述されている。

スキーマはインスタンス・スキーマとクラス・スキーマに分けられる。直観的には、インスタンスはこれまで述べてきたオブジェクトに対応し、インスタンス・スキーマには個々のオブジェクトの状態が記述される。またクラスとはインスタンスの集合を表すものであり、クラス・スキーマには、個々のインスタンスが持ちうる属性などが定義される。

① フレームによる知識の表現

先に述べたようにフレーム部には、上位-下位関係、部分-全体関係、属性関係など静的な知識が記述される。基本的な枠組は、代表的なフレーム・システムであるKRL [Bobrov77]と同様に、フレームはスロットの集合であり、スロットはファセットの集合となっている。

```

schema(cis.vtr750x.
  [(superC.録画方式,[(value,[vhsビデオ])]),
   (superC.録音方式,[(value,[hifiビデオ])]),
   (has-part,[(value,[powerボタン,電源ランプ,
                     再生スイッチ,...])]),
   (ステイタス,
    [(car-num.1),
     (enumeration-[再生,録画,
                     待機,...])]),
   (電源, [(car-num.1),
           (enumeration,[on,off])]) ],
[s-rule(dengen.dengen-on=1.V,
  (kr-schema(V,[(電源,[off])],true),
   event(押す.1,[(object,[SW])]),
   kr-schema(V,[(電源,[on])],true),
   [('$var-constraint',(
     SW#power ボタン:-part-of(SW.V) ))]),
s-rule(dengen.dengen-lamp=1.V,
  kr-schema(V,[(電源,[on] ]],true),
  kr-schema(LP,[(点灯状態,[on])],true),
  [('$var-constraint',(
    LP#電源ランプ:-part-of(LP.V) ))])
)].

```

図3 スキーマによる知識表現の例

```

schema(cis.録画する,
  [(superC, [(value,[動作])]),
   (agent, [(value-class,人)]),
   (object, [(value-class,テレビ番組)]),
   (tool, [(value-class,vtr)]),
   (obligatory-case,
    [(value,[object.goal])]) ],
  [ ]).

```

図4 スキーマによるイベントの表現例

(a) 概念体系の表現

ある概念を下位概念に分ける場合、どのような特徴に注目して分類するか、すなわち、どのような見方で分類するかによって、異なった分類が可能となる。例えば、ビデオの例では、録画方式で分類すると、ビデオは「VHSビデオ」と「ベータビデオ」に分類できる。また、録音方式で分類すると「モノラルビデオ」と「ハイファイビデオ」に分類できる。録画の可能性に着目すれば「再生専用ビデオ」と「ふつうの録画ができるビデオ」に分類することもできる。これまでのフレームシステムにおいては、このような見方による分類ができなかった。本処理系では、スロットに対してこの見方を付加情報として付けることで、上で述べたような概念体系の記述を自然に行うことができる。なお、5章で、このような概念体系を用いた照応参照の問題について考察する。

(b) 否定の表現と閉世界仮説

Prologをベースにしたシステムでは、その簡単さから、否定を「閉世界仮説」に基づいて扱っている場合がほとんどであるが、このような仮説が全ての対象分野で適用できるわけではない。単に値が不明である場合もあれば、否定型の情報のみが判っている場合もある。このような問題を扱うには、推論に使用されるデータが完全であるか不完全であるかを表現する枠組や陽に否定を記述する枠組が必要である。推論の対象となるデータは主としてスキーマのフレーム部のスロット値となるため、スロット値のレベルで完全性を表現する枠組を設け、閉世界仮説と開世界仮説とを使い分ける。

(c) イベントに関する知識

図4は「録画する」というイベントのスキーマによる表現例である。2. 3での述べたように、フレーム部のスロットをとって格を設け、格フィルアに対する制約条件が記述してある。「録画する」の例では、格であるagent.object.tool はそれぞれ「人」、「テレビ番組」、「vtr」のインスタンスでなければならないことが表現されている。

(2) スキーマルールによる知識の表現

スキーマのルール部に記述されたルールをスキーマルールと呼ぶ。ルール部には、オブジェクトの動的な性質やオブジェクト内で成り立っている論理的依存関係などを記述する。ビデオを例にとると、動的な性質として考えられるのは、ビデオに対して行なう操作に対してビデオの内部状態がどの様に変化するかといった知識であり、「電源がoffの時POWER ボタンを押すと、電源がonになる」などである。また、ビデオ内で成り立っている論理的依存関係の例としては、「ビデオの電源がonならそのビデオの電源ランプは点灯している」といった知識である。このスキーマルールは、フレーム部のスロットと同様に、上位のクラスのルールは下位に継承される。

このルールを用いて前向き推論を行うことで、オブジェクトの動作をシミュレートすることが可能である。もちろん、このような推論は純粋なフレーム・システムでもデーモンを駆使することで可能である。しかし、ルールとして独立して記述することで、知識がより宣言的に記述可能になり、知識の要素間の関係が明確になる。

(2) ルールによる知識の表現

(1) で述べたようにオブジェクトの上位一下位関係や、個々のオブジェクトにおける動的な知識を表現するにはスキーマで表現すればよいが、イベント間の因果関係などを表現するには適当ではない。また、ビデオの操作の仕方などといった手続きに関する知識もイベント間の因果関係として捕えることが出来る。これらは2章でも述べたようにルールとして表現する。

図5は「カセットをビデオで再生する」方法をルールで表現したものである。手続きの目標となるイベントをルールの結論部に、手続きを構成する動作をルールの条件部に記述することで、手続き型の知識を表現している。ルールの付加情報には、現在ルール中の変数に対する制約条件が記述されている。制約条件としては、変数にバ

```

rule( operation, play-back,
  seq(event(入れる2 .1. [(object.[PW])]),
    event(入れる1 .1. [(object.[K]),
      (goal.[VTR])]),
    event(押す.1. [(object.[SW])]),
    event(再生する.1. [(object.[K]),
      (tool.[VTR])]) .
    [(`$var-constraint'.(
      VTR # vtr;
      K # カセット;
      PW # 電源:-attribute -of(PW.VTR);
      SW # power ボタン:-part-of(SW.VTR) )]))
) ) .

```

図5 ルールによる操作手続きの表現例

インドされるインスタンスのクラス名と他の変数にバインドされるスキーマとの部分-全体関係などが記述されている。

(3) 時間の表現とワールド

故障診断など現在の機器の状態のみを扱えばよいタスクにおいては、時間の概念を扱う必要性は少ない。しかし、操作法のコンサルテーションなどでは、操作の順序が問題になる場合が多いと考えられる。すなわち、A、B 2つの操作を行う場合でも、それらの実行の順序が異なると、機器の状態も異なる。したがって、時間、より正確には時刻の表現が必要となる。

中島らはUranusの多重世界を用い、個々の時刻ごとに世界を設けることで時間を扱っている【中島88】。例えば、時刻 t_1, t_2, \dots, t_n に対してそれぞれ世界を考え、 $t_1 \sim t_n$ まで時間が経過した状態を次のように表現している。

```

(with t1
  (with t2 ... (with tn ... ) ... )
)

```

しかし、この方法では次のような点で問題があると思われる。

- ①時刻 t_i と時刻 t_j の間に新たに時刻 t_x を設定しようとした場合、時刻 t_i の世界の元で時刻 t_x の世界を作り、さらに時刻 t_x の世界の上に時刻 t_j 以降の世界を再構築する必要がある。
- ②時刻間の相対関係が明確でない場合でも、とりあえず時間軸上の前後関係を決定しないと異なる2つの時刻の事象が表現できない。
- ③いわゆるATMSにおけるワールドのように、仮説に基づいた推論と組合せて使用したい場合、1つのワールドtree上に2種類の異なるワールドが存在することになる。

以上の点を考慮して、時間をワールドの属性の形で扱うことはせず、ワールドとは全く独立な枠組みを設ける。我々のシステムでは、時刻に依存した値を持ちうるのはスキーマのスロットであるので、スロットの値を値リストと時刻情報のペアで表現する。

3. 2 推論機能

推論機能としては、フレーム・システムが持つ継承、スロットレベルでの制約条件の検査などの他に、ルールを用いた前向き推論、後向き推論の機能を提供する。ルールを用いた推論には、スキーマルールを用いた推論と因果関係のルールを用いた推論の2つのレベルが可能である。また、2章で述べたようにワーキングメモリをワールドという形で実現してあるので、これを用いることで、必要に応じて副作用のある推論結果を破棄してワーキングメモリを回復することが出来る。

3. 3 インプリメント

上で述べた機能を持つ知識処理モジュールを、ICOTで開発された逐次型推論マシンPSI上にESPを用いて構築した。プログラムの大きさは、コメントを含め約7000行である。ESPはオブジェクト指向言語であるため、スキーマで表現された知識は、各概念をESPのオブジェクトとして実現することも可能であるが、作成したモジュールでは、知識のデバッグの容易性とロード時間の観点などから、ファイルごとにESPのプログラムに変換し主記憶上にロードしている。なお、フレーム型データの処理は、現在インタプリタによって処理しているが、田中らが提案しているように、フレームを論理型プログラムに変換することで、高速化を図ることも可能である【田中86】。

4 試用と検討

4. 1 対話システムへの応用

(1) 概要

我々は意味談話理解の研究の一環として、ユーザとの対話によりコンサルテーションを行うコンサルテーションシステム(Information Service System by Analyzing Conversational Context - ISAC-)を開発しており、現在はタスクとしてビデオの操作法に関するコンサルテーションを題材として使用している【浮田87】。

対話システムはこれまでSHRDLUを初めとして様々な対話システムが開発されている【Vinograd72】。SHRDLUとの大きな違いは、SHRDLUでは、対話の対象となる世界が計算機内部で閉じており、システムが世界の全ての情報を把握しているのに対し、本システムの場合、ユーザの使用していると考えられるビデオの状態やユーザの行った操作は、ユーザの発話を通してしか知ることができない。これは、エキスパートが電話などを通じてコンサルトしているケースに相当する。ここで、当システムにおける談話処理について簡単に説明する。まず構文解析結果から①指示名詞、②‘その’や‘この’がついた名詞、③名詞単独、④述語の格表現における必須格の省略を照応形として検出する。そして、先行文脈中に存在するオブジェクト(主としてインスタンス・スキーマ)の中から種々の制約条件を満たす候補を抽出する。このとき②③については、部分-全体関係(再生スイッチとビデオ)や属性関係(電源とビデオ)も調べて抽出する。また、適当な候補が抽出できなかった場合には新しいインスタンス・スキーマを生成し

候補とする。次に知識ベースに記述されているビデオの操作手続きを参照して最適な候補を決定し、情報が補われた格表現を出力する。現在は最適な候補の決定に際してスキーマルールは使用していない。なお、知識ベースを用いて最適な候補を決定するアルゴリズム（最適解積判定法）の詳細については【住田87】を参照されたい。

(2) 対話例

図6に本システムによる対話例を示す。この対話では、対話に先立ち、ビデオとテレビが1台ずつあり、ビデオの電源はオフであるといったことが初期状態としてシステムに与えられている。その様な初期状態が与えられていない場合には、システムはビデオの状態やビデオに対して行った操作についてユーザに質問しコンサルテーションを進めていく。

4. 2 知識表現の評価

コンサルテーションでのユーザの入力文を理解するために、ビデオに関する知識を3章で述べた形式で表現した結果について述べる。ビデオとしてある機種（家庭用ハイファイビデオ）を、操作マニュアルをもとに表現した。コンサルテーションの範囲としては通常の録画、再生などの操作の他に、時計合せや録画予約などについての知識も表現した。

(1) スキーマによる知識の表現

スキーマにより表現した概念数は約300である。そのうち約100はイベント（ビデオの操作に関連した動詞）で、残りの約半数はビデオ本体とスイッチなどその構成要素を表現したもの、それ以外は電源、チャンネルといった属性概念やテレビ番組、画像などである。

概念体系の表現においては、見方ごとにサブクラスを分類できることを示した。実際の表現では、例で用いたビデオの他には「端子」に関する階層でしか使用していないが、概念階層が自然に表現でき、その有効性が確認できた。この概念体系は5、1で述べるように、オブジェクトの同定に有効に利用できる。

閉世界/開世界仮説の使い分けについては、ビデオに関する知識の表現では、スロットの値の個数が全て1個であったため、有効性は確認できなかった。また否定に関するものとしては、スロットに否定の値を設定しようとするとき、背理法により肯定の値を求める推論機能を設けた。例えば、図3のビデオの例では、電源スロットは値としてonかoffしか取らないことが指定されているため、電源スロットにonの否定を設定する操作が行われると、実際にはoffという値がスロット値となる。このような推論は、コンサルテーションにおいて、ユーザへの質問に対して否定の回答がなされた時に、それを処理するのに有効であることが確認された。

操作に対するビデオの動作（内部状態の変化）と各状態に対応するランプなどの点灯状態は約100のスキーマルールで記述しており、これにより操作に対するビデオの状態をシミュレートすることが可能となっている。

ユーザ：ビデオでカセットを再生したい
システム：電源を入れて、カセットをビデオに入れて、再生スイッチを押して下さい。
ユーザ：カセットがビデオに入らない
システム：ビデオの電源がオフだからです。
ユーザ：電源スイッチを押した
システム：はい、わかりました。
ユーザ：カセットを入れて再生スイッチを押したが動かない
システム：ビデオの再生ランプが点灯していますか。
ユーザ：はい
システム：VHF出力端子がテレビにつながっていますか。
ユーザ：いいえ
システム：VHF出力端子がテレビにつながっていないからです。

図6 対話システムによる会話例

ただし次のように、継続して起こるイベントは表現していない。

IF ビデオが再生状態 かつ カセットが最後まで巻き取られていない

THEN カセットは回転する

また、次のような定性的な知識も表現していない。

IF カセットが回転している

THEN 将来カセットは停止する

先に述べたように、現在は照応参照処理においてスキーマルールを使用していないが、スキーマルールで表現した知識もイベント間の関係として見ることができ、それを利用していく必要があると考えられる。

(2) ルールによる知識の表現

ビデオの操作手続きはルールによって表現した。表現に際してはマニュアルの操作案内を可能な限りそのままルールの形で表現したが、次のような付帯条件がついた記述はそのまま表現できないため、操作の部分のみを記述した。

「～○○ランプがつくように**ボタンを押し～」

↓

「～**ボタンを押し～」

また、ルール内の変数に対する制約条件であるが、図4に示したルールの例からも分るように、それぞれの変数には、変数に束縛されるオブジェクトが満足すべき性質が、制約条件として記述されている。現在この制約条件は、スロット値が単に未定義で矛盾するといえない場合にも制約違反と判断されるため、照応参照処理でルールを使用する上での問題点となっている。

5 考察

ここでは、3で述べた知識表現と推論の枠組に記述された知識を利用していく上で今後問題となると思われる点について考察する。

5. 1 概念体系と同一性

2. 1でも述べたように、対話の中では、同一のオブジェクトを示すのに異なる語が使用される場合が多く、しかもそれらの語が表す概念が異なる場合もある。このような場合にオブジェクトの同一性を判断するには、概念体系上の位置関係が判断基準の1つとなる。先行文脈中のオブジェクトの概念名をA、処理中のオブジェクトの概念名をBとすると、AとBの関係として以下のような場合が考えられる。

① AがBのサブクラスの場合

(例 A : VHSビデオ, B : ビデオ)

② AとBがある共通の概念Cの同じ見方のサブクラス

(例 A : テレビ, B : ビデオ, C : 電気製品)

③ AとBがある共通の概念Cの異なる見方のサブクラス

(例 A : VHSビデオ, B : ハイファイビデオ, C : ビデオ)

④ BがAのサブクラスの場合

(例 A : ビデオ, B : VHSビデオ)

図7は上の分類を図式化したもので、概念間の矢印はsuperCリンク、リンクにまたがった点線はそれぞれ見方を表している。①は照応参照の最も一般的な場合であり、2つのオブジェクトが矛盾した性質を持っていない限り2つのオブジェクトは同一であるということが出来る。概念体系を設計する際、同じ見方のサブクラス同士はそれらのインスタンスの集合が排反となるように概念体系を考えている。したがって、②の場合には、それら2つのオブジェクトは本質的に異なる性質を持っているため同一化することは出来ないが、③の場合には同一化することが可能である。但し実際には、上の例のように概念Cが直接の上位概念になっているとは限らないため、そのような場合には概念体系上での探索量が増え処理コストが増加する。④は①の反対の場合で、①では単に同一化するだけなのに対し、④では先行するオブジェクトについての情報が增加することを意味する。

電総研の石崎らはアクティブマッチング法と呼ばれる方法で異なる概念のオブジェクト間の同一性をチェックしている【石崎88】。この方法では、全ての概念をある1つの概念をルートとして複数のユニット(フレーム)の組み合わせによって表現しておき、同一性のチェックは、それら概念の構造の比較により行っている。したがって、チェックの際には上で述べたような概念体系上での位置関係を基にした比較は行う必要はないが、シソーラスが大きくなったときに個々の概念構造も大きくなるため、同一性のチェックに要する時間も増加すると考えられる。

なお、4. 1で述べた対話システムにおける照応参照処理では、同一化できる場合として①の場合しか対象としていないため、それ以外の場合についても扱えるように拡張する必要がある。

5. 2 構成要素の生成

一般に幾つかの構成要素をもつ有形物の下位概念Aを

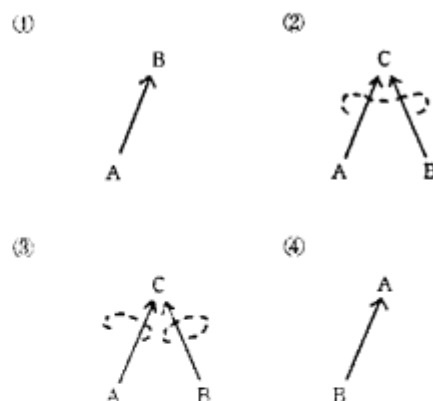


図7 概念体系における概念間関係

考えた場合、Aのインスタンスは基本的に、Aのhas-partに記述されている各構成要素のインスタンスを構成要素として持つと考えることができる。例えば4で述べたビデオの知識表現においては、ビデオの構成要素として約80の構成要素がある。このような知識があるとき、ビデオのインスタンスを生成する度に、一度にすべての構成要素のインスタンスを生成し、has-part階層を作るのは、資源や処理時間の点から満足できるものではない。そこで、談話解析やタスク処理における必要に応じて、インスタンスを生成しhas-part階層を作っていく機構が必要となると考えられる。

5. 3 タスク処理におけるワールドの利用と管理

ユーザとの対話によって操作法のコンサルテーションを行うというタスクを考えた場合、ユーザに対して最適な回答を行なうためには、システムは、ユーザが使用していると考えられる機器の現状を知る事が重要なのは明らかである。しかしながら、ユーザは、非現実世界についての陳述も可能である。これの顕著な例は条件文である。

条件文の処理は基本的に、まず仮定の部分を評価する際に新しいワールドを生成し、そのワールド上で推論することで処理し、その後、ワールドを条件文の処理前の元のワールドに戻すことで処理できると考えられる。

ここで重要となるのは、仮定を含む文を処理した後の文の処理である。例として下の1から3を考えてみよう。

- (1) もし、録画スイッチを押したらどうなるか
- (2) もし、代わりに再生スイッチを押したらどうなるか
- (3) そのあと、巻き戻しボタンを押したらどうなるか

図8に(1)～(3)を処理する際のワールドの構成例を示す。文(2)の仮定「再生スイッチを押す」は文(1)の仮定「録画スイッチを押す」の代わりに評価しなければならない。したがって文(2)を処理する際には、(1)の処理を行う際と同様にワールドW0を親とするワールドW2を生成して処理しなければならない。ところが、文(3)の仮定は文(2)を処理した後の状況が対象となっているため、ワールドW2を親とするワールドを生成して

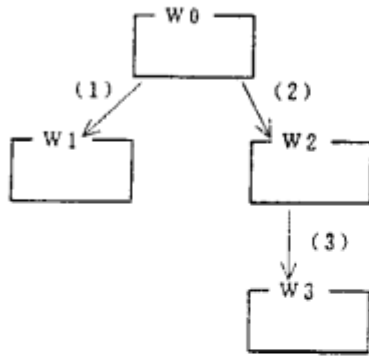


図8 ワールドを用いたタスク処理

処理する必要がある。このように、仮定を含む文を処理した後では、次の文がどのワールドをベースにして述べられているかを解析することが重要である。

6 おわりに

本論文では、文脈処理のための一般的な知識の分類と表現・推論の枠組について述べた。まず文脈理解に必要な知識として、対象とする世界における知識を、オブジェクトに関する知識、イベントに関する知識、イベントとイベントの関係、概念体系の4種類に分類し、それらがフレームとルールという2つの知識表現のパラダイムを用いて表現できることを述べた。また、文脈処理をする上で生じる解釈の曖昧さを扱うという観点からワールドを用いた推論の必要性について述べた。

次に、現在P S I上で開発を行っている対話システムにおける知識処理として、ビデオの操作コンサルテーションのための知識の表現を行い、表現の枠組について評価した。オブジェクトに関する知識、イベントに関する知識、概念体系はフレーム部とルール部を持つスキーマと呼ぶ枠組で記述した。オブジェクトに関する知識については、オブジェクトの属性や他のオブジェクトとの関係はフレームで、また、オブジェクトの持つ動的な性質はスキーマルールで記述することができた。また、イベント間の関係の1つとしてビデオの操作手順をルールで記述した。基本的には、操作の目標をルールの結論部とし、実際の操作にあたる動作をルールの条件部で記述できるが、操作に対する付帯条件があるようなものは、そのまま記述することはできない。

表現したビデオの知識は、ユーザの入力文の意味談話解析に使用されるほか、コンサルテーションのための知識としても利用されている。

ここで述べた知識表現の枠組は、他のタスクにおける自然言語理解にも適用可能であると考えられる。今後、現在のシステムにおいてさらに実験を行い、文脈理解における知識の利用について考えていきたいと思っている。

(参考文献)

- [Bobrov77] Bobrov,D.G. and Winograd,T. : "An Overview of KRL,a Knowledge Representation Language." Cognitive Science, Vol.1,1977.
- [deKleer86] de Kleer,J. : "An Assumption-based TMS." Artificial Intelligence 28,1986.
- [Doyle79] Doyle,J. : "A Truth Maintenance System." Artificial Intelligence 12,1979.
- [Hobbs79] Hobbs,J. : "Cohereence and Coreference." Cognitive Science, Vol.3,1979.
- [Minsky75] Minsky,H. : "A framework for representing knowledge." in Winston,P. (ed.), The psychology of computer vision, McGraw-Hill,1975.
- [Rieger75] Welger,C.J. : "CONCEPTUAL MEMORY AND INFERENCE." In Schank,R. (ed.), Conceptual Information Processing1, NORTH-HOLLAND, 1975.
- [Shank77] Schank,R. and Abelson,R. : "Scripts,Plans,Goals and Understanding."Lawrence Erlbaum Associates,1977.
- [Wilensky83] Wilensky,R. : "Planning and Understanding."Addison-Wesley,1983.
- [Winograd72] Winograd,T. : "Understanding Natural Language." Academic Press,1972.
- [石崎86] 石崎俊, 井佐原均: "文脈処理技術",情報処理 Vol.27, No.8,1986.
- [伊東87] 伊東幸宏, 島田洋, 柴田昇吾, 橋博文, 高木朗, 小原啓義: "自然語の文章の理解と知識の利用について,"電子情報通信学会論文誌D Vol. J70-D No.4,1987.
- [浮田87] 浮田輝彦, 住田一男, 木下聡, 佐野洋, 天野真家: "談話理解機能を持つ機器操作案内システムー概要ー",情報処理学会第35回全国大会予稿集,1987.
- [島津86] 島津秀雄, 友納正裕, 高島洋典: "ストーリー駆動型アニメーションシステムー物語理解部ー",情報処理学会第33回全国大会予稿集,1986.
- [住田87] 住田一男, 浮田輝彦, 木下聡, 佐野洋, 天野真家: "最適解釈判定法による会話文の理解",電子情報通信学会NLC 研究会 NLC87-16,1987.
- [田中86] 田中穂積, 小山晴生, 奥村学: "知識表現形式D C K Rとその応用",コンピュータソフトウェア, Vol.3, No.4,1986.
- [辻井87] 辻井潤一: "知識の表現と利用",昭晃堂,1987.
- [中島86] 中島秀之, 眞訪基: "多重世界機構による時間の表現と問題解決",情報処理学会第33回全国大会予稿集,1986.
- [向井85] 向井国昭, 田中裕一, 安川秀樹, 三吉秀夫, 平川秀樹, 滝塚孝志, 横井俊夫: "談話理解システムD U A L S第2版の構想",情報処理学会第31回全国大会予稿集,1985.
- [劉87] 劉学敏, 西田豊明, 堂下修司: "統合的自然言語理解のための論理的整合性と確からしきの管理",電子情報通信学会NLC 研究会 NLC87 -7,1987.