

## 質問応答システムにおけるユーザ発話モデルと 協調的応答の生成

山田 耕一<sup>†</sup> 溝口 理一郎<sup>††</sup> 原田 直樹<sup>†</sup>

質問応答システムにおけるユーザ発話モデルと協調的応答生成モデルを提案する。協調的応答の生成に関してはこれまでも多くのモデルやシステムが提案されているが、個々のモデルが扱う協調的応答の種類は少ない。我々のモデルはユーザの質問の意図を推論し、意図に応じた応答を生成することによって様々な協調的応答を可能にすることが大きな特徴である。本論文ではまず、ユーザの発話とその意図の関係を表わすユーザ発話モデルを導入する。次に日常会話でよく見られる協調的応答の分類を行い、応答の種類ごとにユーザの意図と応答の関係を議論する。そしてユーザ発話モデルに基づいてユーザの発話からその意図を推論する方法を述べる。ここではドメイン独立な意図推論ルールと話題となる対象物に関する知識を用いる。さらに、推論された意図を用いて先に分類された協調的応答を生成する方法を述べる。最後に、提案した協調的応答生成モデルに基づいて構築したプロトタイプシステムによる実行例を示す。

### User's Utterance Model and Cooperative Answering for Question-Answering Systems

KOICHI YAMADA,<sup>†</sup> RIICHIRO MIZOGUCHI<sup>††</sup> and NAOKI HARADA<sup>†</sup>

A user's utterance model and a cooperative answering model are proposed for question-answering systems. Many models for cooperative answering have been developed since early 1980s and have shown that they can achieve some form of cooperative answers. However, the coverage of each model has been limited. In this paper, we propose a model that covers various types of cooperative answers. The paper starts with the user's utterance model that relates an utterance with its intentions. Then, it classifies cooperative answers into several classes and discusses the relations between the user's intentions and the classified answers. An intention recognition mechanism is then developed employing domain-independent rules and knowledge about the topic object. The recognized intentions are used to generate appropriate cooperative responses. The proposed model is verified by a prototype system that we developed.

#### 1. はじめに

意思決定支援システムやデータベースの自然言語インタフェースなど、質問応答システムにおける課題の一つとして協調的応答の生成がある。対話型システムにおける協調的応答の必要性は80年代以降広く認識されるようになり、これまでも多くの研究が行われている<sup>1)~10), 13)~16)</sup>。

研究の一つの大きな流れはデータベースの分野にある<sup>1)~10)</sup>。それらのほとんどは質問の答の後に何らかの情報を加える<sup>2), 5), 6), 8)</sup>か、失敗した質問\*に対して理

由<sup>2), 3), 5), 6), 9)</sup>または代替案<sup>2), 4), 7), 10)</sup>を提示する。付加情報としては質問の答に対する根拠<sup>5), 6)</sup>や関連情報<sup>2), 8)</sup>を提示するものが多い。失敗理由には対象領域の知識を用いるもの<sup>3), 5), 6)</sup>、データベーススキーマに関するユーザの誤解を指摘するもの<sup>9)</sup>、質問内の名詞句の存在を前提条件としてその不成立を指摘するもの<sup>2), 3)</sup>などがある。また、代替案としては失敗した質問の条件を緩めるか取り除いて新たな質問を作り、その検索結果を提示するものが多い<sup>2), 4), 7)</sup>。しかし、これらの協調的応答はデータベース検索という限られた状況での対話を対象としており、また各手法が扱う協調的応答の種類も2, 3種類と少ない。

一方、発話行為論<sup>11)</sup>に基づくプランベースアプローチ(PBA)<sup>12), 13)</sup>やJoshiらの方法<sup>14), 15)</sup>では、質問者

<sup>†</sup> 技術研究組合 国際フェジィ工学研究所  
Laboratory for International Fuzzy Engineering  
Research

<sup>††</sup> 大阪大学産業科学研究所  
The Institute of Scientific and Industrial Research,  
Osaka University

\* 答がNoであるYes/No質問あるいは答の見つからないWH質問。

と応答者の双方の信念を用いてより一般的な状況での対話を扱うが、これらも対象とする協調的応答の種類は多くない。また、PBA では互いの信念と行為の関係に基づいた詳細なプランニングによって質問者の意図を推論するが、この方法は一方の信念に対する他方の信念といった信念の入れ子を扱うことになり、意図推論の各ステップの粒度が小さく、複雑で時間のかかる処理となる。さらに Carberry<sup>16)</sup> は、ユーザはドメイン知識を十分持たないため質問するにもかかわらず、PBA はユーザが正しいプランを作るのに十分な知識を持つと仮定する矛盾を指摘した。Carberry はこの指摘の後、矛盾を解消するために精緻なアプローチを提案した<sup>16)</sup>が、実装を考慮にいたより現実的な方法に目を向けると、信念に関してより簡略化されたモデルの方が望ましい。実際、我々の日常の対話においてはよほど複雑な問題の場合を除き、自分の信念についての相手の信念までは考えていないように思われる。

そこで本稿では対話者双方の信念を考慮することなくユーザの意図を推論することによって、より多くの種類の協調的応答を行う応答生成モデルを提案する。そして、このモデルをある目的のために情報収集を行うユーザとの質問応答対話に適用する。

本章以降、第2章ではまず質問者の発話モデルについて述べ、第3章で本稿で扱う協調的応答を分類し、その多くが質問者の意図と密接な関連を持つことを示す。第4章と第5章では意図推論と応答生成方法についてまとめ、第6章で実験システムでの実行例を示す。

## 2. ユーザ発話モデル

質問応答システムに対するユーザの質問はあるゴールを達成をするための情報収集行為と考えられる。そこでユーザゴールを頂点とする図1のようなゴール木を考え、各サブゴールをユーザが望む行為で表現する。各行為は質問の形で現われる情報収集行為とそれ以外の非情報収集行為に大別する。また、それらの行為で表現されるゴールをそれぞれ情報収集ゴール、非情報収集ゴールと呼ぶ。

このゴール木において頂点のゴールを、以下に述べる他の意図の源という意味で、一次意図と呼ぶ。ユーザの質問は末端の情報収集ゴールに対応すると考えられるが、ある一つの質問に対応する末端のゴールをその直接意図と呼ぶ。また、一次意図と直接意図の間に

あるすべてのゴールを間接意図と呼び、その中で最も下位の非情報収集ゴールを発話目的とする。さらに、発話目的の下位にある直接意図を除く末端の情報収集ゴールを関連意図と呼ぶ。また、直接意図、間接意図、関連意図、一次意図をすべて併せて単に意図と呼ぶことにする。

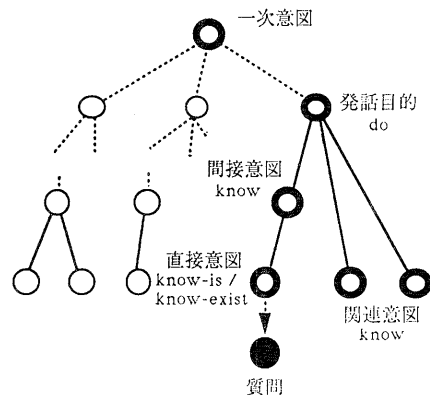
例えば、「列車 A は午前 8 時に出発しますか?」という質問に対し、直接意図は「列車 A が午前 8 時に出発するかどうか知る」という情報収集行為であり、その間接意図としては「列車 A が何時に出発するか知る」や「列車 A に乗車する」などが考えられる。このとき後者が最も下位の非情報収集行為であればこれが発話目的となり、「列車 A が何番線から出発するか知る」という情報収集行為は関連意図となる。

## 3. 協調的応答

本稿で扱う協調的応答は直接応答と間接応答の2種類から構成される。直接応答とは質問の直接意図に対する応答である。間接応答はそれ以外の応答で、間接意図や関連意図、あるいは関連意図を除く意図の前提条件に関する応答である。直接応答は省略されたり存在しない場合<sup>2)</sup>があり、間接応答は複数存在する場合もある。著者らは人間同士で行われる質問応答対話を分析してよく行われる12種類の間接応答を見だし、それらを図2のように四つに大別した。

### 3.1 前提応答

前提応答は質問の意図（関連意図を除く）を実行、あるいは満足するために成り立たなければならない前



know, know-is, know-exist は、質問が YES/NO タイプであるときの各意図の述語。do は任意の非情報収集行為を表わす。

図1 ユーザ発話モデル

Fig. 1 User's utterance model.

提条件に関する応答で、以下の3種類がある。

### (1) 前提修正

質問の意図の前提条件が誤っている場合に、それを修正する応答である。

Q1: 横浜美術館行きのバスの停留所はどこですか?

A1<sub>1</sub>: 横浜美術館行きのバスはありません。

A1<sub>2</sub>: 3番です。でも横浜美術館は開いていません。

ここで、A1<sub>1</sub>はKaplan<sup>2)</sup>がCorrective indirect responseと呼んだものと同等で、質問文中に現われる名詞句の存在をデータベースで調べ、存在しない場合にそれを否定することで生成する。しかし、A1<sub>2</sub>はそれとは異なる。これを生成するためには発話目的「横浜美術館で美術を鑑賞する」を推論し、その前提条件「横浜美術館は開いている」を導き出す必要がある。

### (2) 前提通知

質問の意図の前提条件が満たされていない場合でも前提条件を修正するのではなく、そのまま前提条件を通知する場合がある。

Q2: この施設は利用可能ですか?

A2<sub>1</sub>: はい、でも予約をする必要があります。

前提条件が満たされていないとき前提修正と前提通知のどちらを行うかは状況に依存する。例えばQ2の質問場所がその施設のフロントで行われたのなら、

A2<sub>2</sub>: いいえ、お客様の予約は入っていません。

という応答でもよい。しかし、本稿では質問者が発話目的の行為を起こす前の情報収集対話を対象としている。従って前提条件が質問者の行為であるなら、これから質問者が前提条件を満たすことが可能であり、前提修正より前提通知の方が適切な応答となる。逆に、A1<sub>1</sub>やA1<sub>2</sub>のように前提条件が質問者の行為でないときは前提通知より前提修正の方が自然である。

### (3) 前提確認

前提条件が満たされるかどうかを知らないとき、応

答者は相手にそれを尋ねることがある。

Q3: そのスポーツ倶楽部にプールはありますか?

A3: はい。あなたはこの倶楽部の会員ですか?

しかし前提条件が満たされるかどうかかわからないとき、応答者が常に前提確認を行うわけではない。満たされていないと仮定して前提通知を行ったり、「質問者が入場料を持っている」のように満たされると仮定する前提条件もある。ただし、この状況で前提修正が行われることはほとんどない。

## 3.2 情報付加応答

情報付加応答は直接応答に加えて質問者に有用と思われる情報を付加する応答である。どのような情報を付加するかによって以下の4種類に分かれる。

### (1) 希望情報付加

Allen<sup>13)</sup>はある命題についてのYes/No質問の答がNoであるとき質問者は正しい命題を知りたいと考え、間接応答としてそれを付加する方法を提案した。しかし、質問者が正しい命題を知りたいのは直接応答がNoのときばかりではない。次の例ではYes/Noのどちらの場合も正しい命題の付加が行われるが、それらは間接意図としての情報収集行為「東京行きの305便の出発時刻を知る」に対する答、つまり質問者の希望情報と考えることができる。

Q4: 東京行きの350便の出発は朝ですか?

A4<sub>1</sub>: はい。305便は午前8時の出発です。

A4<sub>2</sub>: いいえ。305便は午後2時の出発です。

しかし、これらの応答の解釈はただ一つではない。Webberら<sup>5)</sup>のようにYesあるいはNoの根拠と解釈することも、直接応答がNoの場合は後述の代替情報付加と見られることもできる。本稿では、後述する応答生成の仕組みの観点から、Yesの場合は希望情報付加、Noの場合は代替情報付加として扱う。

### (2) 関連情報付加

発話目的を達成するために必要な関連情報収集行為、つまり関連意図に対する答を付加する応答である。例えば、質問Q4の発話目的は「朝出発する東京行きの305便に乗る」であるが、それを達成するには出発時刻のほかに出発ゲートも知る必要がある。従って関連意図として「東京行きの305便の出発ゲートを知る」を推論し、次のような応答を生成する。

A4<sub>3</sub>: はい。305便は午前8時に5番ゲートから出発します。

これはAllen<sup>13)</sup>のProviding more information than requestedやKaplan<sup>2)</sup>のSupportive indirect

1) 前提応答	3) 理由付加応答
・前提修正	・反期待理由付加
・前提通知	・非標準理由付加
・前提確認	・標準理由付加
2) 情報付加応答	4) 質問による応答
・希望情報付加	・協力質問
・関連情報付加	・意図確認質問
・間接情報付加	
・代替情報付加	

図2 協調的応答の分類

Fig. 2 Classification of cooperative responses.

response と同様の応答になる。

### (3) 間接情報付加

質問に対する答を知らないが、その答を推論するのに必要な知識と情報を不完全ながら持っている場合に行われる。

Q5: そのアパートの日あたりはよいですか?

A5: わかりません。でも、窓は南向きです。

この場合、応答者は窓が南向きだという事実、および「窓が南向きでしかも前に日を遮るものがなければ日あたりはよい」という知識を持っている。ただ、二つある条件のうちの一つしか知らないため、知っている事実のみを伝えたと解釈できる。この応答の生成にはドメインに関する推論知識が必要となる。

### (4) 代替情報付加

質問者は発話目的を達成するための情報収集行為として質問を行う。しかし、何らかの理由によってその情報収集や発話目的の遂行が不可能だとわかったとき、質問者は一段上の意図を満たす代替情報を望むだろう。質問者の発話目的が実現不可能と判定されるのは次の2通りである。

- 1) 前提修正によって前提条件が否定された場合。
  - 2) 応答が発話目的の前提条件を否定している場合。
- 前提通知の場合も発話目的の前提条件は満たされていないが、質問者がその前提条件を満たす行為をする可能性があるため実現不可能とは見なさない。2)は質問内容が発話目的の前提条件の場合で先の A4<sub>2</sub>の応答を含む。この例では「朝出発する東京行きの305便に乗る」という発話目的が、305便が午後出発するために実行不可能である。また、2)は以下のQ7のように直接応答が Yes の場合も扱い、Q8のような Kaplan<sup>2)</sup> の Suggestive Indirect Response をも含む。本稿ではこれらを一括して代替情報付加として扱う。

Q6: 東京行きの305便は何時に出ますか?

A6: 10時です。でも305便には空席がありません。午後2時の306便なら利用できます。

Q7: 東京行きの305便はキャンセルされましたか?

A7: はい。でも306便なら利用できます。

Q8: その居間にはテレビはありますか?

A8: いいえ。でも寝室にはあります。

### 3.3 理由付加応答

日常の対話では直接応答を行った後にその理由を述べる場合がよくある。従来の理由付加の研究は直接応

答が No か存在しない場合<sup>2),3),5),6),9)</sup>がほとんどであるが、理由を付加する状況はそれだけではない。なお、質問 Q1 の応答 A1<sub>1</sub>は「停留所がない」という直接応答に対する理由と解釈することができ、Q4 に対する A4<sub>1</sub>, A4<sub>2</sub>も Yes/No の直接応答の理由(根拠)と見なせるが、ここではそれらを除いたものを扱う。

#### (1) 反期待理由付加

質問者の質問に対し、応答者がその期待に応えられない場合に行う理由付加である。

Q9: 東京行きの臨時便はどこから出ますか?

A9: わかりません。臨時便の情報を持っていません。

情報収集のための質問応答システムでは質問者の期待は「システムが自分の質問に答える」ことに限ってよいが、範囲を一般の対話に広げると様々な期待が当然ある。次の例では、質問者の期待は質問に答えるだけでなく「相手が自分と一緒に帰る」ことも含まれる。

Q10: 一緒に帰りましょうか?

A10: いいえ、本屋に寄って帰ります。

#### (2) 非標準理由付加

直接応答が標準あるいは普通と異なる場合に、なぜ標準どおりでないかを伝える応答である。

Q11: 大阪行きの便はキャンセルされましたか?

A11<sub>1</sub>: はい。濃い霧のためキャンセルされました。

#### (3) 標準理由付加

非標準の答を想定する理由があるにもかかわらず標準的な答を返す場合に行われる。Q11の質問に対する次の間接応答が標準理由付加である。

A11<sub>2</sub>: いいえ、霧がでていますが出航します。

### 3.4 質問による応答

日常会話では質問に対して質問を返すことがよくある。前述の前提確認もその一つであるが、以下のような場合にも質問による応答が行われる。

#### (1) 協力質問

質問者の質問に答えたりその発話目的に協力しようとして、応答者が質問者に不足する情報を尋ねる場合がある。これを協力質問と呼ぶ。

Q12: この店にボールペンがありますか?

A12: はい。何色がよろしいですか?

#### (2) 意図確認質問

質問者の質問の発話目的が明確でなく、その真意を尋ねる質問である。意図推論に失敗したときに行われることがある。

Q13: 箸を持っていますか? (例えば, バス  
の中で)

A13: いいえ, 箸で何ををするのですか?

#### 4. 意図推論

協調的応答の生成ではその多くで質問者の意図が重要な役割を果たしている。そこで、ここでは直接意図からその発話目的と関連意図を推論する方法およびそのときに用いる知識について述べる。なお、これ以降システムの入出力は直接意図の形で行われるものとする。一般の対話でよく見られる省略や代名詞、指示語については前処理によって解消されていると仮定する。

##### 4.1 ドメイン知識

システムが参照するデータはすべてフレームベース内に対象物知識として蓄えられているものとする。各フレームはクラスとインスタンスの区別を持つ。ユーザからの質問に対しシステムは質問の対象となるフレームを参照し応答を生成するが、この対象物知識は質問対象だけでなく、質問やその意図に現われるすべての概念を含む。参照したスロットの値が存在しない場合は、デーモンとしてその値を推論するルールベースが後ろ向きに起動される。また、質問対象となる事象については後述する意図推論用の知識(スロット)を持つものとする。

##### 4.2 意図表現

質問の意図は次のように表現する。

〈意図〉 ::= 〈述語〉〈行為者〉〈対象項〉〈付属項〉。

ここで、{|} は 0 個以上の繰り返しを意味する。〈述語〉は〈意図〉が WH タイプの情報収集行為なら know, 存在に関する Yes/No タイプの情報収集行為なら know-exist, その他の Yes/No タイプの情報収集行為であれば know-is, 非情報収集行為の場合はその行為を表す任意の動詞が入る。また、〈行為者〉には〈述語〉の行為者、〈対象項〉には行為の対象となるもの、〈付属項〉にはその他の記述が入る。

〈対象項〉 ::= (〈上位〉〈値〉〈修飾項〉)。

〈付属項〉 ::= (〈付属性〉〈上位〉〈値〉〈修飾項〉)。

〈修飾項〉 ::= (〈属性〉〈上位〉〈値〉)。

ここで、〈値〉には対象物知識のフレーム名、〈上位〉にはその上位フレーム名が入る。〈修飾項〉は直前の〈値〉を修飾するもので、〈属性〉はその〈値〉が示すフレームのスロット名に対応する。また、〈付属

```
R1: if do (S (C I)) then know-exist(S (C I))
R2: if do (S (C I (A P V))) then know-is(S (C I (A P V)))
R3: if do (S (C I)) then know (S (C ?))
R4: if do (S (C I (A P V))) then know (S (C I (A P ?)))

R5: if do (S (P V)) then know-exist (S (C I (A P V)))
R6: if do (S (P V)) then know-is (S (C I (A P V)))
R7: if do (S (P V)) then know (S (C ? (A P V)))
R8: if do (S (P V)) then know (S (C I (A P ?)))
```

後件部においてSは行為者、その後にくリストは対象項である。対象項内のC, I, A, P, Vについては4.2節の対象項と修飾項の定義を参照。また、前件部におけるdoは任意の非情報収集行為を意味するが、これは4.2節の定義の点から意図ではない。行為者Sの直後のリストはそれ(R1-R4ではCまたはI, R5-R8ではPまたはV)が話題であることを示し、後件部と直接意図とのマッチングによる後向き推論時のルール間の競合解消に利用される。また、このリストが対象項、付属項のどちらになるかは話題となる対象物の用途知識によって決まる。

図 3 意図推論ルール

Fig. 3 Intention inference rules.

性〉には意図の表現に必要な付属項名が入る。〈値〉には変数\*や特種変数としての“?”が許される。例えば「大きなスーパーで安いテレビを買う」というユーザの意図は次のように表現される。

```
buy(user(TV-set T(price price-ranges cheap))
(place supermarket S(size size-ranges
big))).
```

なお、意図の前提条件や応答の伝達内容も同様に表現されるが、物の存在と属性記述のためにそれぞれ exist, is 述語を用意する。また、これらの述語の時は〈行為者〉の項は使われない。

##### 4.3 意図推論ルール

意図推論は意図推論ルール(図 3)を用いて行われる。これらは前件部に非情報収集行為、後件部に情報収集行為に対応する表現を持ち、人が発話目的を持つときどのような質問の意図が生まれるかを記述する。例えば、R1は「行為者SがクラスCのインスタンスIに関して何かをするとき、Sはそれが存在するかどうかを知ろうとする」という意味を持つ。ルール内の非情報収集行為が具体化されていなく、行為内の各項の名前や値がすべて変数であるという意味で、ドメイン独立な知識となっている。

意図推論の概略手順を以下に示す。

step 1: 直接意図内から話題を取り出す。話題の決定はそれ自身が研究対象となる問題<sup>7), 17), 18)</sup>であるが、本稿では話題候補の語を優先順位つきのリストに記

\* 変数は1文字の大文字アルファベット、定数は2文字以上のアルファベットで記述する。

述しておき、それを利用して決定する。

step 2: 話題が直接意図の対象項の〈上位〉または〈値〉にあれば R1 から R4 のルールのいずれかの後件部に、話題が修飾項の〈上位〉または〈値〉にあれば R5 から R8 のいずれかのルールの後件部に直接意図をマッチさせる。マッチしたルール内の変数は直接意図の変数や定数にバインドされる。

step 3: マッチしたルールの前件部を具体化前の発話目的とし、次節に示す方法で発話目的を具体化する。

step 4: 発話目的の具体化後、今度はこれを図 3 の R4 の前件部にマッチさせて前向きに推論し、関連意図を推論する。この時どの属性を関連情報とするかは図 4 に示す対象物の意図推論用知識 (reference) を参照して決める。なお、質問の直接意図に現われる属性が reference 内にあるときはそれを無視する。

#### 4.4 発話目的の同定

発話目的の具体化は、対象物知識内に記述された話題に関する用途知識を用いて行う。

まず、直接意図が次のように得られているとする。

know-is (user (flight 123 (departure morning T))).

[123 便は朝出発するかどうか知る.]

話題がフライトとすると R2 が逆向きに適用され、具体化前の発話目的は次のように得られる。

do (user (flight 123 (departure morning T))).

[朝出発の 123 便に関して何かする.]

ここで、flight 123 で始まる部分は対象項あるいは付属項の〈上位〉、〈値〉、〈修飾項〉である。そして、flight の用途知識が take (S (flight F)) で与えられているとするとその発話目的は次のように具体化される。

take (user (flight 123 (departure morning T))).

[朝に出发する 123 便に乗る.]

しかし、話題には状況によって異なる用途を持つものもある。そこで、そうした場合にも対処できるように対象物知識内に図 4 のような意図推論用知識を用意する。図においてテレビは日用品と商品の 2 つのクラスのサブクラスであるが、どちらの視点をとるかは状況によって異なり、それによって用途も異なる。用途を決めるためにはまず状況を判定しなければならない。

状況は過去の状況と場所の情報を用いて判定する。

つまり、その対話セッションの過去に今候補となっているどちらかの状況があればそれを現在の状況とする。過去に同じ状況がなければ、場所の情報と図 4 における normal-place の知識を利用する。すると、

know-exist (user (TV-set T (place room LR-1))),

[居間 1 にテレビがあるかどうか知る.]

という直接意図に対しては watch (user (program P) (with TV-set T (place room LR-1))) という発話目的が得られ、

know-exist (TV-set T (place store store-1)),

[店 1 にテレビがあるかどうか知る.]

に対しては buy (user (TV-set T) (place store store-1)) のように得られる。

## 5. 協調的応答の生成

前章で推論した意図を用いて協調的応答を生成する方法について述べる。

### 5.1 前提応答

前提応答の生成にはまず、直接意図と発話目的の前提条件を導かねばならない。そこで、次の二つの方法で前提条件を導くことにする。

- 1) 意図表現内の〈値〉が対象物知識に存在する。
- 2) 行為に関する知識から導く。

A<sub>1</sub> の応答は 1) によって求めた前提条件「横浜美術館行きのバスの存在」が満たされなかった場合である。

ここで、Q1 の質問の発話目的が美術館の用途知識を用いて次のように得られるとする。

enjoy (user (fine-arts X) (place museum YM)).

[横浜美術館で美術を鑑賞する.]

また、関連する行為の知識が図 5 のように与えられるとする。この行為知識を繰り返し適用し、対象物知識を参照すると以下の三つの前提条件が得られる。

Frame : TV-set		
subclass-of:	daily-necessaries	goods
normal-use:	watch(S (program P) (with TV-set T (place room R)))	buy(S (TV-set T) (place store S))
situation:	home-life	shopping
normal-place:	room	store
normal-area:	the-house	the-neighborhood
reference:		price, maker

図 4 対象物の意図推論用知識

Fig. 4 Object knowledge for intention recognition.

```
is(museum YM(status h-status open)), DE=ask,
PR=2.
pay(user(money ¥500)), DE=satisfied, PR=3.
is(bus bus-34(for museum YM
(status t-status operated))), DE=satisfied,
PR=4.
```

ここで、前提条件の優先度は新たな行為知識が適用される度にプラスされて上記に示す値になる。これらの前提条件を優先度によってソートした後、対象物知識を用いて順にチェックし、満たされないものがあれば次のように応答を生成する。

- 1) 満たされない前提条件がユーザの行為であれば must を付けて前提通知を生成し、そうでなければ not を付けることによって前提修正を生成する。

例：not(is(museum YM(status h-status open))).  
must(pay(user(money ¥500))).

- 2) 前提条件が満たされるかどうかかわからないときは、
  - a) 前提条件のデフォルトが satisfied であれば満たされていると想定する。
  - b) デフォルトが ask であれば前提確認をする。
  - c) デフォルトが notify であれば前提通知を行う。

## 5.2 情報付加応答

### (1) 希望情報付加

Yes/No 質問に対する直接応答は直接意図内の対象がデータとして存在すれば Yes, 存在しなければ No となる。A4<sub>1</sub>の例では305便がクラス「朝」のインスタンスとなる時刻に出るため Yes となる。また、間接応答としての希望情報付加は Yes を確認したインスタンス時刻を用いることで生成する。

### (2) 関連情報付加

関連情報付加は前章に示した方法で関連意図を求め、それに対する答を直接応答と同様に生成する。

### (3) 間接情報付加

間接情報付加を行うためには、直接応答を得るデータが存在しないとき推論によってその値を求める機能を持つ必要がある。これはフレームのデーモンを利用して実装する。値がないとき

はデーモンによってドメインのルールベースを起動し、後ろ向き推論によって必要な値を求める。そして、その値を結論するルールの前件部に複数の条件があつて一つの条件だけが満たされず推論に失敗したとき、成功した条件をユーザに伝えることによって間接情報付加が行われる。

### (4) 代替情報付加

発話目的の前提条件が否定されその実現が不可能になったとき、代替の発話目的を生成しそれを提案する。一般的には代替目的の生成はその上位の意図を推論し、その意図を実現する別のプランを立てる処理になるが、ここでより簡便な方法で行う。

行為の前提条件はその多くが、その行為に含まれる情報や概念を参照している。例えば「横浜美術館行きのバスに乗る」行為の前提条件は横浜美術館かバスかその両方に関するものが普通である。従つてある前提条件が満たされないなら、その参照情報を前提条件が満たされる同種の情報で置き換えた発話目的を生成すればよい。先の例では他の美術館や交通機関で置き換えることになる。ただし、話題を置き換えるときは発話目的全体が異なるものにならないよう、その話題と同じ用途を持つ情報に限定する。整理すると、次の手順で代替情報を探す。

- 1) 満たされない前提条件と発話目的内で共通に用いられている情報 (4.2 節における〈値〉、それが変数の場合は〈上位〉) を一つ選ぶ。
- 2) 選ばれた情報と同じ上位クラスを持つ代替情報を探す。ただし、それが話題のときは同じ用途を持つものだけに限り、場所の情報ときは許される地域内で代替情報を探すという意味で、図4に与えられた nomal-area の範囲内で探す。

```
enjoy (S (pastime pastime P) (place hall H))
PC: go-to (S (hall H)), DE=satisfied, PR=2
enter(S (hall H)), DE=satisfied, PR=1

enter (S (hall H (fee money M)))
PC: pay (S (money M)), DE=satisfied, PR=2
is (hall H (status h-status open)), DE=ask, PR=1

go-to (S (place P (distance far F)))
PC: take (S (transportation T (for place P))), DE=satisfied, PR=1

take (S (transportation T))
PC: is (transportation T (status t-status operated)),
DE=satisfied, PR=1
```

PC : 前提条件, DE : デフォルト, PR : 優先度

図5 行為知識

Fig. 5 Knowledge of actions.

3) 代替情報が見つからない場合、1) を満たすものがほかであればその代替情報を探す。

ここで、1) を満たす情報が複数ある場合は次の優先順位で選択する。

- 1) 時間の情報があればそれを最優先。
- 2) クラスとインスタンスがあればクラスを優先。
- 3) クラスが複数あって、そのうち一つが話題である場合は、話題を優先。
- 4) クラスが複数あって、話題がインスタンスである場合は任意のクラスを選ぶ。
- 5) クラスが存在しないとき、話題となるインスタンスを選択する。

ここで時間の制約を最優先するのは、情報収集のための対話を行うユーザは一次意図の計画段階である、時間変更の余地が大きいと考えられるからである。また、インスタンスよりクラスを優先するのはクラスによる条件指定はインスタンスに比べると重要度が低いと考えられるからであり、さらに話題を優先するのは話題が検索対象となっている場合が多いからである。

例えば次に示す発話目的の前提条件 `exist(train T (for seashore S) (depar. time morning))` が満たされないとする。

```
take(user(train T(for seashore S) (depar. time
                                     morning)))
```

[朝、海岸へ行く列車に乗る.]

前提条件と発話目的に共通に現われるのは、`train`, `seashore`, `morning` の三つですべてクラスである。この場合は時間情報があるため、まず朝以外の時間帯に前提条件を満たすものがあるかどうかを調べる。あればそれが代替情報となるが、もしなければ次は残った二つのうち話題である `train` の代替情報を探す。そして朝海岸へ行くバスがあればそれを代替情報としてユーザに示す。ここでもし変数 `T` の代わりに特定の列車のインスタンスが入っていたら、インスタンスよりクラスが優先されるため `seashore` の代替情報が探され、例えば朝に出るその列車が山へ行くのであればそのことが示される。

### 5.3 理由付加応答

#### (1) 反期待理由付加

反期待理由付加を実現するためには、システムがユーザの期待とその実行の障害を認識する能力を持たねばならない。しかし情報収集を目的とする対話では、システムに対するユーザの期待は「与えられた質

問に答える」ことだけと考え、その反期待理由付加は機械的に生成する。

#### (2) 非標準理由付加

この応答を生成するには標準の知識と標準と異なる結果を答える仕組みが必要であるが、これはフレームのデフォルト設定機能とドメインルールを用いて実現する。対象物知識内のある対象物の属性値について質問され、前述のデーモンによってドメインルールが起動されて値が得られたとする。ここで、その値が設定されているデフォルトと異なる場合に非標準であると判断し、その値を結論したルールの前件部の条件を非標準の理由とする。

#### (3) 標準理由付加

標準理由付加は、非標準の結果を想定する理由があるにもかかわらず標準どおりであるときに行われる。これは、非標準の結果を結論するドメインルールが起動され、しかし前件部の複数の条件の一つが否定されて標準どおりの答が得られたとき、成立した条件を示すことによって生成する。間接情報付加とは標準値としてのデフォルトが必要なこと、失敗したルールが非標準値を結論部に持っていることが異なる。

## 5.4 質問による応答

質問による応答のうち協力質問に関しては、フレームのデーモンとしてユーザへの質問を許すことで実現する。意図確認質問は発話目的の推論に失敗したときに行われるが、適切なインプリメントを行うためには、単なる意図推論の失敗をきっかけとするのではなく、文脈や状況の理解が必要と考えられる。

## 6. 実装および実行例

### 6.1 実装の概要

12種類の協調的応答のうち特に質問の意図と強く関連するものについて実装を行い、協調的応答の実験を行った。実験システムでは質問は直接意図の形で入力し、応答は伝達すべき情報を生の形で出力する。実装は Sun workstation 上に Kyoto Common Lisp を用いて行った。実装した協調的応答は、前提応答の3種類すべてと情報付加応答のうち間接情報付加を除く3種類である。残りの6種類のうち間接情報付加、標準理由付加、非標準理由付加、質問応答はフレームのデーモンとドメインルールを用いて実現できるものであり、残りの反期待理由付加と意図確認質問はさらに深い検討が必要と思われる。以下に応答生成の概略アルゴリズムをまとめる。



- 1) 直接意図に対して直接応答を生成する。
- 2) 発話目的を推論する。
- 3) 直接意図と発話目的の前提条件を求め、5.1節の方法で前提応答を試みる。ここで前提確認、前提通知が行われたら終了。前提修正が行われたら5)、前提応答がなければ4)へ。
- 4) 直接応答が Yes/No 質問に対する Yes であれば希望情報付加を試みる。
- 5) これ以前の応答で前提条件が否定され、発話目的の遂行が不可能なときは代替情報付加を試みる。
- 6) 関連意図を推論し、関連情報付加を試みる。代替情報付加によって発話目的が修正されたときは、修正された発話目的から関連意図を推論する。
- 7) 終了

## 6.2 実行例

- (1) 希望情報付加・関連情報付加  
次の質問の直接意図が入力される。

```
know-is(user(flight F(departure morning M)
           (destination airport
            Tokyo))).
```

[東京行きの便の出発は朝ですか?]

システムは対象物知識ベースの中からこの条件に合う便を探す。departure が morning のインスタンス時刻で、destination が Tokyo の便が対象物知識ベースには用意されている。そこでまず直接応答の Yes を出力し、マッチした便とその時刻を希望情報付加として出力する。

```
is(flight 305 (departure morning 8:00)
   (destination airport Tokyo)).
```

[東京行き 305 便は午前 8 時の出発です。]

このとき、直接意図の変数 F, M にもマッチした値が代入される。ここで話題が flight であるとする。4.4 節の例と同様にして図 3 の R2 が適用され、発話目的は次のように得られる。

```
take(user(flight 305 (departure morning 8:00)
             (destination airport Tokyo))).
```

[東京行の午前 8 時発の 305 便に乗る。]

次にこの発話目的に R4 を前向きに適用して関連意図を得る。このとき話題である flight の reference 知識を参照するが、ここには gate-no, departure, destination の三つの属性が記述されている。しかし後者の二つは直接意図内に現われているため無視さ

れ、gate-no についてののみ関連意図が生成される。

```
know(user(flight 305(gate gate-no ?))).
```

[305 便の出発ゲートを知る。]

そしてこれに対する答が関連情報付加としての次のように出力される。

```
is(flight 305(gate gate-no 3)).
```

[出発ゲートは 3 番になります。]

(2) 前提修正・代替情報付加・関連情報付加

Q1 の場合、次の直接意図が入力される。

```
know(user(bus B(stop place ?) (for museum
               YM))).
```

これに対する直接応答は、bus フレームの下位フレームを調べることによって次のように得られる。

```
is(bus bus-34(stop place No. 3) (for museum
                                   YM)).
```

[そのバスの停留所は 3 番です。]

話題が横浜美術館であるとする。発話目的とその前提条件は 5.1 節に示したように得られる。ここで横浜美術館の状態が closed であるため前提条件が満たされず、次のように前提修正が行われる。

```
not(is(museum YM(status h-status open))).
```

[でも、横浜美術館は開いていません。]

さらに発話目的の前提条件が否定されたため、代替情報付加を行おうとする。ここで、発話目的と満たされない前提条件の双方に現われる情報は YM である。そこでこれと同じ上位フレーム、つまり museum を持つ代替情報を対象物知識ベースの中から探す。この場合、museum の normal-area に記述された範囲 (the-neighborhood) 内に神奈川美術館 (KM) が存在し、次の代替情報付加が行われ、出力される。

```
enjoy(user(fine-arts X) (place museum KM)).
```

[神奈川美術館なら大丈夫です。]

さらに、museum の reference 知識に special 属性があり、KM の special 属性には美術観賞に関する関連情報があるため次の関連情報付加応答が加わる。

```
is(museum KM(special exhibition
              impressionism-ex)).
```

[神奈川美術館では印象派展をやっています。]

(3) 代替情報付加

Q8 の直接意図と発話目的は 4.4 節で示したようにそれぞれ次のように得られる。

```
know-exist(user(TV-set T(place room the-LR))).
watch(user(program P)
```

```
(with TV-set T(place room the-LR))).
```

この例では place 属性に the-LR の値をもつ TV-set のインスタンスがないため、まず直接応答として No を出力する。これは発話目的の前提条件 exist (TV-set T(place room the-LR)) をも否定しているため、発話目的が実行不可能となる。代替情報の候補は TV-set と the-LR であるが、TV-set がクラスであるためまずこの代替情報を対象物知識ベースの中から探す。しかし place 属性が the-LR で TV-set と同じ normal-use を持つ代替情報がない (TV-set が話題である) ため、次は the-LR の代替情報を探す。これは場所の情報であるため図 4 の normal-area, normal-place 知識を用いてその家の中の別の部屋を探す。そして place 属性の値が the-bed-room である TV-set のインスタンスを見つけ、次の代替情報付加を出力する。

```
watch(user(program P)
      (with TV-set TV-set-1(place room
                             the-bed-room))).
```

この場合は reference の値がないため関連意図は生成されず、関連情報付加は行われない。

## 7. おわりに

質問応答システムにおける協調的応答の分析と分類を行い、それらを生成する方法を提案した。本手法ではユーザの発話の意図をゴール木で表すユーザ発話モデルを導入し、そのゴール木の中の下位レベルの発話の意図を推論し、それらの意図に関する応答を生成することで協調的応答を実現する。

意図推論は意図推論ルールと対象物知識を用いて行われる。「知る」という行為以外ルール内に記述された行為がすべて具体化されていない、行為のパラメータもすべて変数で与えられているという意味で、意図推論ルールは全くドメイン独立な知識である。また、対象物知識の内容は応用に合わせて作られるが、その枠組みは汎用的である。意図推論は始めにルールを適用して具体化前の行為を得た後、話題となる対象の用途知識を用いて行為の具体化を行う。状況によって異なる用途を持つ対象の場合は、以前の状況かその行為が通常行われる場所の知識を用いて決定する。

協調的応答はこのようにして推論した意図を用いて生成する。ただし、前提応答は意図を遂行するための前提条件を知る必要があるため、行為に関する知識を用いてそれらを導く。また、理由付加応答や情報付加応答の中の間接応答はドメインの問題解決に関する知

識を必要とする。本稿ではそれらを含めて分類したすべての協調的応答の生成方法を提案し、意図推論が大きな役割を果たすものについて実装を行い、本手法の有効性を実証した。

今後は残された協調的応答の実装を行うと共に、我々が現在開発している協調的意志決定支援システム<sup>19),20)</sup>の一部として本モデルを組み込む予定である。

## 参考文献

- 1) Gaasterland, T. et al.: An Overview of Cooperative Answering, *J. Intell. Inf. Syst.*, Vol. 1, pp. 123-157 (1992).
- 2) Kaplan, S. J.: Cooperative Responses from a Portable Natural Language Query System, *Artif. Intell.*, Vol. 19, pp. 165-187 (1982).
- 3) Motro, A.: SEAVE: A Mechanism for Verifying User Presuppositions in Query Systems, *ACM Trans. Office Inf. Syst.*, Vol. 4, No. 4, pp. 312-330 (1986).
- 4) Motro, A.: FLEX: A Tolerant and Cooperative User Interface to Databases, *IEEE Trans. Knowledge and Data Engineering*, Vol. 2, No. 2, pp. 231-246 (1990).
- 5) Webber, B. and Joshi, A.: Taking the Initiative in Natural Language Data Base Interactions: Justifying Why, *Proc. COLING 82*, Horecky, J. (ed.), pp. 413-418, North-Holland (1982).
- 6) Gal, A. and Minker, J.: A Natural Language Database Interface That Provides Cooperative Answers, *Proc. 2nd Conf. Artif. Intell. Appl.*, pp. 352-357 (1985).
- 7) Cuppens, F. and Demolombe, R.: How to Recognize Interesting Topics to Provide Cooperative Answering, *Inf. Syst.*, Vol. 14, No. 2, pp. 163-173 (1989).
- 8) Cuppens, F. and Demolombe, R.: Extending Answers to Neighbour Entities in a Cooperative Answering Context, *Decision Support Syst.*, Vol. 7, pp. 1-11 (1991).
- 9) Trabelsi, Z., Kotani, Y. and Nishimaru, H.: Heuristics for Generating Informative Responses to Failing User's Queries in Natural Language Database Interface, *Proc. HCI International 93*, Vol. 2, pp. 362-367 (1993).
- 10) Ozawa, J. and Yamada, K.: Generating a Fuzzy Model from a Database and Using It to Find Alternative Data, *1st Australian and New Zealand Conf. Intell. Inf. Syst.*, pp. 560-564 (1993).
- 11) Searle, J. R.: *Speech Acts*, Cambridge Univ. Press (1969); 坂本, 土屋(訳): 言語行為, 勁草書房 (1986).

- 12) Cohen, P. R. and Perrault, C. R.: Elements of a Plan-Based Theory of Speech Acts, *Cognitive Science*, Vol. 3, pp. 177-212 (1979).
- 13) Allen, J. F. and Perrault, C. R.: Analyzing Intention in Utterances, *Artif. Intell.*, Vol. 15, pp. 143-178 (1980).
- 14) Joshi, A., Webber, B. and Weischedel, R. M.: Preventing False Inferences, *Proc. COLING 84*, pp. 134-138 (1984).
- 15) Joshi, A., Webber, B. and Weischedel, R. M.: Living Up to Expectations: Computing Expert Responses, *Proc. AAAI 84*, pp. 169-175 (1984).
- 16) Carberry, S.: *Plan Recognition in Natural Language Dialogue*, MIT Press (1990).
- 17) 吉田, 平松ほか: 機械への音声入力のための汎用対話管理システム, 人工知能学会研究会資料 SIG-SLUD-9202-9, pp. 77-85 (1992).
- 18) 中川, 加藤: 質問応答における話題管理方式について, 情報処理学会論文誌, Vol. 28, No. 9, pp. 998-1001 (1987).
- 19) 奴久妻ほか: 協調的対話システムにおける意志決定支援, 第9回ファジィシステムシンポジウム, pp. 173-176 (1993).
- 20) 原田ほか: 協調的対話システムにおけるユーザモデルについて, 第9回ファジィシステムシンポジウム, pp. 69-72 (1993).

(平成5年12月10日受付)

(平成6年7月14日採録)



山田 耕一 (正会員)

1956年生。1978年東京工業大学工学部制御工学科卒業。1980年同大学院総合理工学研究科修士課程修了(システム科学専攻)。1989年より山武ハネウエル(株)アドバンステクノロジーセンター勤務。主に知識工学, ファジィ理論の応用研究・開発に従事。1991年より2年間, 技術研究組合国際ファジィ工学研究所にて対話型システムのインタフェース研究に従事。計測自動制御学会, 電気学会, 人工知能学会, 日本ファジィ学会各会員。



溝口理一郎 (正会員)

1972年大阪大学基礎工学部電気工学科卒業。1977年同大学院基礎工学研究科博士課程修了。同年大阪電気通信大学工学部講師, 1978年大阪大学産業科学研究所助手, 1987年同研究所助教授, 1990年同教授。現在に至る。工学博士。パターン認識関数の学習, クラスタ解析, 音声の認識・理解, エキスパートシステム, 知的CAIシステムの研究に従事。1985年 Pattern Recognition Society 論文賞, 1988年電子情報通信学会論文賞, 1990, 1991年人工知能学会研究奨励賞, 1989, 1990, 1991年人工知能学会全国大会優秀論文賞受賞。電子情報通信学会, 人工知能学会, 日本音響学会, CAI学会, 日本認知科学会, IEEE, AAAI, AACE 各会員。



原田 直樹 (正会員)

1956年生。1978年神戸大学工学部計測工学科卒業。1980年同大学院修士課程修了。同年三菱化成工業(株)[現三菱化学(株)]入社。同社四日市工場にて化学プラントの計装制御および生産技術開発に従事。1992年技術研究組合国際ファジィ工学研究所出向。現在同研究所にて対話型システムのインタフェース開発研究に従事。計測自動制御学会, システム制御情報学会, 日本ファジィ学会各会員。