

## 計算機利用支援システム Neo-ASSIST における ユーザモデルを用いた応答メカニズム

2G-3

上原 龍也 上原 邦昭 豊田 順一

大阪大学産業科学研究所

### 1. はじめに

現在、我々は計算機についてユーザの質問に答える計算機利用支援システム Neo-ASSIST [1][2][3]を開発している。本システムでは、ユーザが実際に求めている応答を行うために、システム利用者の知識状態を表したユーザモデルを用いている。本稿では、Neo-ASSISTにおけるユーザモデルを用いた応答メカニズムについて述べる。

本メカニズムでは、ユーザの質問の意図を推定するために、まずユーザのゴールを仮定する。そして、ユーザモデル上でユーザの思考過程をシミュレートすることによって、仮定したゴールからユーザがなぜ質問したかを調べる。ユーザへの応答は、ゴールとなぜ質問したかというユーザの質問意図を考慮して生成される。さらに、現在のユーザモデルの内容から、ユーザがゴールを達成するために行う次の質問を予測し、その質問をユーザへの応答に反映させることによって、ユーザの質問回数を減少させることが可能である。

### 2. ユーザモデル

#### 2.1 ユーザモデルの獲得

ユーザモデルはシステムが信じているユーザの信念やゴールの集合である。Neo-ASSISTの場合、ユーザモデルの記述は以下のものについての信念からなる。

- ・ 現在の計算機の状態  
「ファイル foo は存在する」
- ・ 計算機についての概念や概念間の関係  
「ファイルには親ディレクトリがある」
- ・ 計算機使用のプラン  
「ファイルを消すためには rm を用いる」
- ・ 質問を行うためのプラン  
「ゴールを解決する方法が分からなければ、その方法をシステムに聞く」

ユーザモデルを獲得する方法としては、対話を始める前にシステムがユーザに質問する方法と、対話で交わされた情報やシステムの持つ領域知識を用いることにより、動的にユーザモデルを獲得する方法がある。しかしながら、前者の方法では、システムからの質問が多くなるためにユーザの手間が増え、また、対話中のユーザの変化に対応することができない。したがって、本研究では後者の動的なユーザモデルの獲得について考える。

Neo-ASSIST では、対話からユーザモデルを獲得するために以下のルールを用意している[2]。

1. ユーザが使用したコマンドの動作をユーザは知っている。
2. ユーザの質問の前提となる事実をユーザは知っている。
3. ユーザの述べた内容をユーザは知っている。
4. システムが説明したことをユーザは知っている。

さらに、対話中で明示的に現れないユーザの知識を獲得するためには、ユーザモデルの内容とシステムの持つ領域知識からユーザモデルの新たな内容を獲得する必要がある。Neo-ASSISTで用いられているルールの一部を以下に示す。

- ・ ユーザモデル内に、ある行為の影響が存在することを知っているという記述があれば、ユーザはその行為が発生したと知っている。
- ・ ユーザモデル内に、ある概念の特殊化にあたる概念をいくつか知っているという記述があれば、ユーザはその一般的な概念を知っている。

#### 2.2 質問意図の推定

質問応答の対話において、個々の質問の意図は状況に依存する。たとえば、「ファイルが消えません」というユーザの発話があった場合、単にファイルの消し方を聞いているのか、もしくはファイルが消えない理由を聞いているのかなど質問意図の候補が複数存在する。したがって、ユーザの質問に答えるためには、ユーザの発話から得られる意図の候補から、ユーザモデルを参照して適切なものを決定する必要がある。Neo-ASSISTではこの問題を解決するために、まずユーザのゴールを推定する。次にユーザモデルに記載された知識を用いて、質問にいたるユーザの思考過程をシミュレートすることにより、質問の意図を推定している[3]。

質問の意図の推定過程は、以下の通りである。

1. ユーザの現在のゴールを仮定する。ゴールとしては、以前、対話に現れたゴールや発話から直接得られたゴール、またはこれらのゴールのサブゴールを設定する。
2. ユーザモデルに含まれるプランを用いてゴールを解決する。
3. ゴールを解決するために支障が生じれば、ユーザの持つ質問を行うためのプランを用いてユーザがどのような質問を行うのかを推測し、この質問とユーザの実際の質問と比較する。もし、同じ質問ならば、この支障を解決することがユーザの質問の意図となる。
4. 推測した質問がユーザの実際の質問と異なれば、新たなゴールを仮定するために、1に戻る。

Question-Answering Mechanism by Using a User Model  
in Neo-ASSIST

Tatsuya UEHARA Kuniaki UEHARA Jun'ichi TOYODA  
The Institute of Scientific and Industrial Research,  
Osaka University

### 2.3 質問の予測

質問意図の推定のためにユーザのゴールを仮定するが、さらにこのゴールを用いることにより、ユーザが次に行う質問を予測することができる。ユーザはその時点で未解決のゴールについて質問を行う。したがって、ユーザの質問に答えたと仮定してユーザモデルを変化させ、質問意図の推定の時と同様にユーザの思考をシミュレートすることによって次の質問を予測する。この予測した質問に対する答えを求め、実際のユーザへの応答に付け加えることにより、ユーザの質問回数を減少させることが可能である。

### 3. 対話例

図1にNeo-ASSISTの対話例を示す。図の左端がuserであればユーザの入力(コマンド入力・質問)、unixであればOSからのメッセージ、assistであればシステムの応答を表す。

まず、(3)の質問に対しては、ユーザのゴールはファイルを消すことであり、ファイルを消す手段がユーザモデルに記述されていないので、ユーザは(3)の質問を行ったと判断する。したがって、ユーザの質問の意図は、「ファイルを消す手段を知りたい」となる。しかしながら、同じような質問(8)の場合、すでにファイルを消す手段をユーザは知っているのので、この時のプランの木は図2のようになる。この時、ユーザの知識ではファイルが消えるはずである。しかしながら、実際にはファイルは消えないので、矛盾を生じる。このことにより、ユーザの質問の意図は「rmでファイルが消えない理由を知りたい」となる。

また、質問(10)では質問の予測が用いられている。この時点でのユーザのプランの木は図3のようになる。ユーザにライトパーミッションについて教えた場合、「ライトパーミッションとは?」というゴールは満たされる。したがって、ユーザの次のゴールは「ライトパーミッションを親ディレクトリに設定する」となる。このゴールに対して、質問を行うためのプランの一つ「ゴールを満たす方法を知らなければ、その方法をシステムに問う」というルールを適用すれば、ユーザの次の質問は、「ライトパーミッションを親ディレクトリに設定する方法を教えてください」と予測できる。したがって、(10)のシステムの応答の下線部であらかじめこの質問に答えている。

### 4. おわりに

本稿では、Neo-ASSISTにおけるユーザモデルを用いて応答を行うメカニズムについて述べた。本メカニズムでは、ユーザのゴールを仮定してそのゴールをユーザが解決する過程をユーザモデルを用いてシミュレートすることにより、質問意図の推定、および質問の予測ができ、ユーザが必要としている応答を生成することが可能となっている。

なお、今後の課題としては、ユーザの誤解に対処することが挙げられる。この問題に対してもユーザモ

デル上のシミュレーションは有効であると考えている。

#### 参考文献

- [1] 上原他: 日本語 Tree Adjoining Grammar を用いた文章生成メカニズムについて, 信学技報, Vol.88, No.460, NLC88-24 (1989).
- [2] 松元他: ユーザモデルの動的な構築と発話プランニングへの利用, 人工知能学会研究会資料, SIG-HICG-8804-2 (1989).
- [3] 上原他: ユーザモデル上のシミュレーションによる質問意図の推定, 人工知能学会ヒューマンインタフェースと認知モデル研究会 (1989年10月発表予定).

```

user      : % ls (1)
unix      : abc foo temp zzz (2)
user      : %? ファイル abc を消したい (3)
assist    : コマンド rm を使いなさい (4)
user      : % rm abc (5)
user      : % rm foo (6)
unix      : rm : permisson denied (7)
user      : %? ファイル foo を消したい (8)
assist    : ファイル foo の親ディレクトリにライトパーミッ
            ションが設定されていない。ファイル
            を消すためには親ディレクトリにライトパーミッ
            ションが設定されている必要がある。 (9)
user      : %? ライトパーミッションとは何ですか (9)
assist    : ライトパーミッションはアクセス権の一種である。ア
            クセス権はファイルの属性の一種であり、そのファイル
            に対して読み出し・書き込み・実行の権利を表したも
            のである。ライトパーミッションは書き込みの権利を
            表す。親ディレクトリにライトパーミッションを設定
            するには chmod u+w を用いる。 (10)
user      : % chmod u+w (11)
user      : % rm foo (12)

```

図1 Neo-ASSIST 対話例

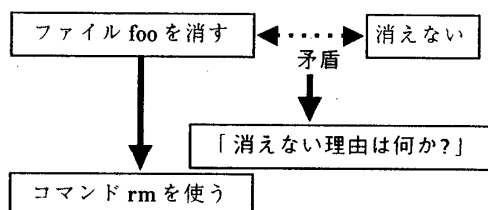


図2 対話例(8)の時点でのユーザモデル

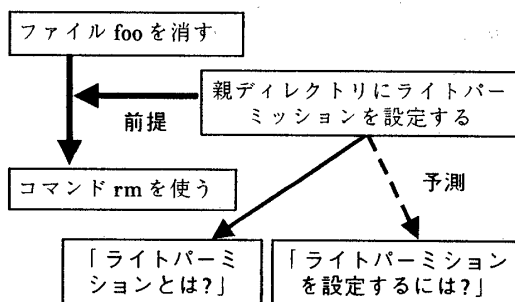


図3 対話例(10)の時点でのユーザモデル