

制限知識下における効率的対話制御

堂坂浩二 中野幹生 宮崎昇 安田宜仁 相川清明

NTT コミュニケーション科学基礎研究所

1. はじめに

音声対話によるヒューマンマシンインタフェースには、人にとって親しみやすい、手や目が使えない状況でも使うことができるという利点がある。人とコンピュータが、人と人が対話する場合と同じように自然な音声対話でコミュニケーションできるようになれば、音声対話は理想的なヒューマンマシンインタフェースとなる。こういった見地から音声対話システムの研究が盛んに行われている (Zue *et al.*, 94; Nakano *et al.*, 99)。

本稿では、ユーザがシステムに情報を要求し、システムがユーザ要求に応じて情報を提供する音声対話システムに注目する。ユーザ音声の認識には誤りが含まれる可能性があるため、システムは、情報を提供する前に、確認対話を実施することによってユーザ発話内容を確定する必要がある。このとき考慮しなければならないことは、システムの知識には制限があるということである。システム知識の制限のため、ユーザ要求の一部はシステムが応えることができない内容に関わるものである可能性があるし、ユーザ発話内容を懸命な努力で確認したとしても、ユーザに情報を提供する際に、その努力に見合っただけの利益をユーザに与えることができないとは限らない。すなわち、ユーザ発話内容のすべてを確認しようとする、確認対話が必要以上に煩雑で非効率なものになる場合がある。確認対話中に音声の認識誤りが生じる場合もあることを考慮すると、確認はできるだけ簡潔に済ませたほうが安全である。

本稿では、音声対話システムが制限知識下において効率的な確認対話を実施するために、確認発話の適切な詳細度を決定する方法を述べる。

2. 確認発話の詳細度の決定

確認発話の詳細度を決定する方法について考察するために、ユーザ要求に応じて気象情報を提供する音声対話システムを例にとる。今、システムは、ユーザが「神奈川県注意報警報を教えてください」という内容を発話したと認識したとする。システムは、現在どこにも注意報警報は発表されていないことを知っているとする。このとき、システムが、ユーザ発話内容を確定するために、「注意報警報ですか?」という確認発話を行ったとする。ここで、シ

ステムの確認発話に対してユーザが肯定の応答を行い、システムがその肯定の応答を正しく認識できたとき、その確認発話による確認は成功したとみなす。「注意報警報ですか?」という確認発話による確認が成功した場合には、システムはユーザの発話内容は注意報警報に関するものであることを確定する。結果として、システムは、「どこにも注意報警報は発表されていません」という情報提供を行うことができる。

システムの確認発話として、「神奈川県注意報警報ですか?」というさらに詳細な確認発話を行うことも想定できる。この確認が成功して、ユーザ発話内容が神奈川県注意報警報に関するものであることが確定した場合には、システムは、「神奈川県には注意報警報は発表されていません」と応答することになる。

「注意報警報ですか?」という確認発話でも、「神奈川県注意報警報ですか?」という確認発話でも、確認が成功した場合のシステム応答内容はほとんど同様のものとなる。しかし、確認発話が詳細なものとなるほど、確認発話をもたらし確認対話が長くなるという意味で確認コストが大きくなる。もし、ユーザが実際には「香川県の注意報警報を教えてください」と発話していたとすると、システムが、より詳細な確認発話「神奈川県注意報警報ですか?」を行ったなら、ユーザは「香川県です」と訂正するであろう。システムがユーザの訂正発話を正しく認識できない場合も想定でき、確認対話が無意味に長くなってしまふ可能性がある。大きな確認コストを払ったとしても、得られる応答内容がほとんど変わらないのなら、確認コストの小さい簡潔な確認発話を選ぶほうがよい。

以上にみたように、詳細な確認発話は確認コストが大きい。しかし、確認が成功した後、確認済みのユーザ発話内容に基づいてシステムが情報提供を行うとき、確認発話の詳細度が大きいほど、提供される情報は簡潔なものとなる。この意味で、詳細な確認発話は情報提供コストを下げるができる。たとえば、システムが、日本の多くの地域に注意報警報が発表されていることを知っているとする。このとき、「注意報警報ですか?」という簡潔な確認発話を使ったとすると、確認成功後のシステムの提供

情報は膨大なものになるであろう。一方、「神奈川県
の注意報警報ですか?」という詳細な確認発話を使
ったとすると、提供情報はより簡潔なものなる。
こういった場合には、簡潔な確認発話を使って確認
コストを減少させると、情報提供コストが極端に増
大してしまうので、簡潔な確認発話を使うことは望
ましくない。以上の観点から、本稿で提案する手法
は、確認コストと情報提供コストのバランスに基づ
いて、確認発話の詳細度を決定する。すなわち、最
も詳細な確認発話を使った際の実確認コスト、情報
提供コストと比べて、確認コストの減少に見合う範
囲内で情報提供コストが増大するような確認発話中
で、最も詳細度の小さい簡潔な確認発話を選択す
る。

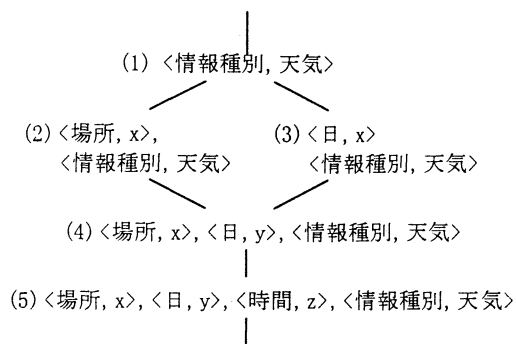


図1 確認階層

3. 関連研究

知識制限を考慮した確認発話の詳細度の決定とよく似た問題として、ユーザ発話内容が曖昧なときに、その曖昧さをどこまで解消すべきかということ、システム応答の同一性という観点から決定する従来法がある (van Beek, Cohen, 91; Rasukutti, Zukerman, 94; Ardissono, 96)。これらの従来法は、ユーザ発話内容の曖昧さを解消してもシステム応答が変化しないなら、ユーザ発話内容の曖昧さを解消する必要はないという方法である。

たとえば、注意報警報がどこにも発表されていないという状況で、ユーザ発話が「神奈川県
の注意報警報を教えてください」という内容なのか、「香
川県の注意報警報なのか」という内容なのか、曖昧
であるとき、ユーザ発話内容がいずれであっても、シ
ステムの応答は、「注意報警報は発表されていません
」と変わらない。したがって、曖昧さを解消する
必要はないという方法である。

しかし、本稿で提案する方法は、システム応答の
同一性が保証されない場合であっても、情報提供コ
ストの増大が確認コスト減少に見合う範囲内である
なら、簡潔な確認発話を使うという方法である。確
認発話の詳細度を下げても、確認成功後にシステム
が提供する情報が同一で変わらないというのは、本
手法が扱う場合の特別な場合である。この意味で、
本手法は、これらの従来法を一般化したものである。

4. 確認発話決定手法

4.1. 発話内容のフレーム表現

ユーザ発話とシステム確認発話の内容は、スロッ
ト名とスロット値の対のリストで表される。これを
フレーム表現とよぶ。たとえば、「横浜市
の明日午後の天気を教えてください」というユーザ
発話内容は、次のフレーム表現で表される。

(フレーム表現1)

<場所,横浜市>, <日,明日>,
<時間,午後>, <情報種別,天気>

このフレーム表現では、場所、日、時間、情報種
別の各スロットに、横浜市、明日、午後、天気とい
う値が代入されている。

また、「横浜市
の明日の天気ですか」というシ
ステムの確認発話の内容は、次の(フレーム表現2)と
して表される。

(フレーム表現2)

<場所,横浜市>, <日,明日>, <情報種別,天気>

フレーム表現の間には、フレーム表現の情報内容
の包含関係に関する順序関係 \sqsubset が存在する。たと
えば、(フレーム表現2) \sqsubset (フレーム表現1)である。
(フレーム表現2)の情報内容は、(フレーム表現1)
の情報内容を真に含まれる。このとき、(フレーム表
現2)の内容の詳細度は、(フレーム表現1)の内容の
詳細度よりも小さいという。

4.2. 確認階層

ユーザ発話内容を詳細度の異なる様々な内容をも
つ確認発話で確認することを可能とするために、
順序関係 \sqsubset に基づいて、確認階層とよぶフレーム表
現の半順序集合をつくる。確認階層に含まれるフレ
ーム表現を確認内容と呼ぶ。たとえば、図1は、天
気情報案内における確認階層の一部である。変数の
値域に関する制約は別途書かれている。たとえば、
図1の(5)では、変数 x は、時間ごとの天気予報が
発表されているような場所であるという制約が書か
れている。

ユーザ発話内容を確認階層中のある確認内容で
確認するとは、ユーザ発話内容のうち、その確認
内容と適合する部分だけを内容としてもつ確認発
話を生成し、その確認発話を使ってユーザ発話内
容を確認することをいう。

本稿では、ある確認内容で確認する際の実確認文
は、その確認内容に含まれるスロットをすべて含む
ような Y/N-疑問文であるとする。たとえば、(フレ
ーム表現1)は、図1に示す確認階層のいずれの
確認内容にも適合する。(フレーム表現1)を確認
内容(4)で確認する場合は、「横浜市
の明日の天気ですか?」という

確認発話が生成され、確認内容(5)で確認する場合は、「横浜市の明日午後の天気ですか?」という確認発話が生成される。

確認階層に含まれる確認内容は、その確認内容での確認が成功した後、確認済みの内容に基づいて情報提供を行うことができるような内容であるとする。しかし、このことは、システムが知識としてもっていないようなスロットが確認階層中に現れてはいけないということを意味しない。たとえば、システムが地震の予知に関する知識はもっていないとしても、地震に関するスロットが確認階層の中に現れてもかまわない。そのようなスロットを含む発話内容に対しては、システムは、知識をもっていないので答えることができないという情報提供を行うと考える。

4.3. 確認コストと情報提供コストの算出

確認階層中のいずれの確認内容で確認発話を実施するかは、確認コストと情報提供コストのバランスにより決定される。

まず、確認発話を実施するときの確認コストは、確認発話に対する訂正発話が生起する確率が高いほど大きくなるコストであり、ここでは、確認内容に含まれるスロットの数であるとする。たとえば、図1の確認階層では、(1)の確認コストは1、(2)、(3)の確認コストは2、(4)の確認コストは3、(5)の確認コストは4である。本稿では、ユーザ発話内容の各スロットの値は、同程度に誤っていると考えるので、確認発話の内容のスロット数が多いほど、確認発話に対するユーザの訂正発話が生起する確率は高くなり、結果として確認対話が長くなると考えている。

情報提供コストとは、ある確認内容の下で確認を行ったとき、確認成功後に確認済み内容に基づいて情報提供を行う際の出力量が大きいほど大きくなるコストである。情報提供コストを計算するためには、対話時点でのシステム知識を検索する必要がある。このため、ユーザ発話内容のうち確認内容と適合する部分に基づいて、提供される情報を検索する。提供情報はフレーム表現として表される。提供情報を表す一つのフレーム表現は、自然言語の一つの節に対応するようなフレーム表現であるとする。このとき、一つの節として実現できる情報はできるだけ一つのフレーム表現の中にまとめられ、可能な限り小さなフレーム表現集合が生成される。ここでは、情報提供のコストは、提供情報をフレーム表現集合で表したときのフレーム表現の数であるとする。

たとえば、(フレーム表現1)の内容をもつユーザ発話を考える。このとき、システムは、横浜市の明日の天気は、午前0時から午後3時まで雨、午後3時から明後日の午前0時まで晴れであるという知識をもっているとする。(フレーム表現1)を確認内容(4)で確認する場合には、確認成功後に提供される情報は次のフレーム表現の集合として表現される。

(提供情報1)

```
{ [<場所,横浜市>, <開始時間, 0 時>,  
  <終了時間, 15 時>, <天気, 雨>],  
  [<場所,横浜市>, <開始時間, 15 時>,  
  <終了時間, 24 時>, <天気, 晴れ>] }
```

また、確認内容(5)で確認が成功した場合、すなわち(フレーム表現1)の全てを確認した後の提供情報は、次のフレーム表現の集合で表現される。

(提供情報2)

```
{ [<場所,横浜市>, <開始時間, 12 時>,  
  <終了時間, 15 時>, <天気, 雨>],  
  [<場所,横浜市>, <開始時間, 15 時>,  
  <終了時間, 24 時>, <天気, 晴れ>] }
```

(提供情報1)、(提供情報2)では、開始時間、終了時間のスロットの値を工夫して、多くの情報ができるだけ一つのフレーム表現の中に表すように努められている。(提供情報1)の場合も(提供情報2)の場合も、情報提供コストは2となる。

図1の他の確認内容で確認した場合の情報提供コストも同様に計算される。たとえば、確認内容(1)、(2)、(3)は、日スロットや場所スロットの一方あるいは両方の値が指定されていないため、確認成功後の情報提供コストはかなり大きくなると考えられる。

4.4. 確認発話の決定

ユーザ発話内容 UF が与えられたとき、確認階層の中で、いずれの確認内容で確認するかを決定する手続き Decide_Confirmation_Utterance を図2に示す。この手続きの戻り値は確認内容の集合 Result である。

まず、ユーザ発話内容 UF 自体を確認内容とみなして、その確認コスト Max_C と情報提供コスト Min_I を計算する。たとえば、UF が (フレーム表現1)である場合、確認コスト Max_C は4となり、情報提供コスト Min_I は2となる。図2において、確認コスト(c)は確認内容cの確認コストを表し、情報提供コスト(c,UF)は、発話内容 UF を確認内容cで確認するときの情報提供コストを表す。

次に、確認内容の集合 Result を UF のみから成る集合として初期化する。また、確認コスト Min_C の値を Max_C - 1 と初期化する。この手続きの骨子は、確認コスト Min_C を1ずつ下げていきながら、情報提供コストの上がり幅が確認コストの下がり幅に見合う範囲内であるという条件を満たす確認内容で、最も詳細度の小さい確認内容を探索することである。

すなわち、確認コスト Min_C をもつ確認内容 c3 を確認階層の中から探し、UF を確認内容 c3 で確認する際の情報提供コストを計算する。このとき、UF の情報提供コスト Min_I から確認内容 c3 の情報提供コストまでの上がり幅に正の係数 α をかけたものが、UF の確認コスト Max_C から確認内容 c3 の確認コスト Min_C までの下がり幅よりも等しいか小さいならば、確認コスト Min_C + 1 の確認内容の代わりに、確

```

Decide_Confirmation_Utterance(UF)
/*変数と定数の宣言*/
  確認コスト Max_C, Min_C;
  情報提供コスト Min_I;
  フレーム表現の集合 Result, Next1, Next2, Next3;
  ブール変数 End_of_Loop := False;
/*初期設定*/
  Max_C := 確認コスト(UF);
  Min_I := 情報提供コスト(UF,UF);
  Result := UF のみから成る集合;
  Min_C := Max_C - I;
/*確認内容の探索*/
  Repeat
     $\exists r (r \in \text{Result} \ \& \ c1 \in \text{確認階層} \ \& \ c1 \sqsubset r)$ 
    を満たす c1 から成る集合 Next1 をつくる;
    If Next1 が空集合である Then End_of_Loop := True
  Else
     $c2 \in \text{Next1} \ \& \ \text{確認コスト}(c2) = \text{Min}_C$ 
    を満たす c2 から成る集合 Next2 をつくる;
    If Next2 が空集合でない Then {
       $c3 \in \text{Next2}$  であるような c3 に関して,
       $\text{Max}_C - \text{Min}_C \geq$ 
       $\alpha (\text{情報提供コスト}(c3, \text{UF}) - \text{Min}_I)$ 
      を満たす c3 から成る集合 Next3 をつくる;
      If Next3 が空集合である
      Then End_of_Loop := True
      Else { Result := Next3;
            Min_C := Min_C - I }
    }
  Until (End_of_Loop = True)
/*終了*/
  Return Result;

```

図 2 確認内容を決定する手続き

認コスト Min_C の確認内容 $c3$ が選ばれ、Result は、そのような $c3$ から成る集合とされる。

正の係数 α は、確認コストと情報提供コストのいずれを重大視するかによって変わる。係数 α が小さいほど、情報提供コストを軽視していることになる。係数 α が小さいほど、情報提供コストの大きい詳細度の小さな確認内容が選ばれる。以下では α は 1 とする。

たとえば、UF が (フレーム表現 1) の場合、確認内容(4)での確認コストは 3、情報提供コストは 2 である。確認コストの下がり幅は 1、情報提供コストの上がり幅は 0 となり、UF の代わりに (4)が選ばれる。次に、確認内容(2)、(3)の確認コストは 2 である。確認内容(2)、(3)の情報提供コストが 4 を超えるようなら、確認内容(4)のみを含む集合 Result を戻り値として手続きは終了する。結局、「横浜市の明日の天気ですか?」といった確認発話が生成される。(2)、(3)の情報提供コストが 4 以下なら、すなわち、日や場所が確定できなくても、提供情報の出力量がそれほど増えないなら、(4)の代わりに、(2)、(3)が選択され

て、さらに、(1)が条件を満たすかどうかの判断に移る。

提供情報の出力量を削減するための一つの技法として、「おおむね晴れ」といった粒度の粗い表現を使うことがある。この場合には、情報提供コストを提供情報の出力量の大きさだけでなく、提供情報を表す際の正確さでも測る必要がある。

本稿で提案した確認発話の詳細度を決定する方法を用いて、確認対話は次の手順で行う。まず、ユーザ発話内容が与えられたとき、本手法で確認内容を決定し確認発話を生成する。確認が成功したら確認対話を終了し情報提供を行う。確認が失敗したら、ユーザの訂正発話の認識結果を取り入れてユーザ発話内容を計算しなおし、手順を最初から繰り返す。

5. おわりに

本稿では、効率的な確認対話を行うことを狙いとして、システムの知識制限を考慮して、確認発話を行うコストと情報提供を行うコストとの間のバランスに基づいて、確認発話内容の詳細度を決定する方法を示した。この方法は、余計な確認対話を避け、確認対話の効率性を高める効果をもつ。

謝辞

日頃よりご指導いただく NTT コミュニケーション科学基礎研究所 東倉洋一所長、石井健一郎部長、萩田紀博部長、熱心に討論して下さった、知識流通研究グループ 加藤恒昭 GL、松下光範氏、マルチモーダル対話研究グループの諸氏に感謝致します。

参考文献

- [Ardissono, 96] L.Ardissono (1996). Dynamic User Modeling and Plan Recognition in Dialogue. PhD Thesis, Università di Torino.
- [Nakano *et al.*, 99] M.Nakano, K.Dohsaka, N.Miyazaki, J.Hirasawa,, M.Tamoto, M.Kawamori, A.Sugiyama, T.Kawabata (1999). Handling rich turn-taking in spoken dialogue systems. Proc. of Eurospeech-99: 1167-1170.
- [Rasukutti, Zukerman, 94] B.Rasukutti, I.Zukerman (1994). Query and response generation during information-seeking interactions. In: Proc. of the 4th Inter. User Modelling Conf.: 25-30.
- [van Beek, Cohen, 91] P.van Beek, R.Cohen (1991). Resolving plan ambiguity for cooperative response generation. In: Proc. of the 12th IJCAI: 938-944.
- [Zue *et al.*, 94] V.Zue, S.Seneff, J.Polifronti, M.Phillips, C.Pao, D.Goodine, D.Goddeau, J.Glass (1994). PEGASUS: A spoken dialogue interface for on-line air travel planning. Speech Communication 15:331-340.