

音声対話システムにおける ユーザの誤り原因推定に基づく動的ヘルプ生成

福林 雄一郎[†]

駒谷 和範[‡]

尾形 哲也[‡]

奥乃 博[‡]

[†] 京都大学 工学部情報学科

[‡] 京都大学大学院 情報学研究科 知能情報学専攻

1. はじめに

現在の音声対話インターフェースは特に初心者にとって必ずしも使いやすいものではない。なぜなら、システムが受理可能な機能や表現は限られているにも関わらず、ユーザにはそれがわからないためである。したがって、ユーザに対する適切なヘルプが必要である。しかし、ユーザ主導で対話を行うシステムでは、ユーザの自由度が高く、ある時点でユーザが行える発話の範囲が広い。システム主導に基づくシステムのように単純に適切なヘルプを生成することはできない。さらに、ディスプレイを持たない電話などのインターフェースでは、音声しか情報を伝える手段を持たないので、一度に説明できる量は限られる。このように、音声対話システムでユーザにヘルプを提供する際にはその内容とタイミングを適切に選ばなければならない。

従来にもユーザの文法外の発話に対して、統計的言語モデルによる音声認識結果を利用してユーザの発話に応じたヘルプを生成する手法 [1][2] が提案された。しかし、ヘルプの生成にそれまでの対話の内容は利用されず、同じ認識結果に対しては毎回同じヘルプが生成される。

我々是对話中に得られる情報からユーザの知識を表現し、これを利用して効率のよいヘルプを生成する。これにより事前に説明を受けていないユーザ（初心者）でも使い方を学習しながら利用できるシステムを開発した。

2. ユーザ知識の管理とヘルプ生成

2.1 ヘルプ提供戦略

ユーザにヘルプを提供する際には、抽象的な概念から順に提供の方が効率が良い。また、すでに知っていることを提供するのには冗長である。例えば、システムに対する知識を全く持たないユーザに対しては「例えば『清水寺の住所を教えてください』と言えば検索できます』のように具体的な発話パターンを教示するよりも、先にシステムの概要を表す「京都の寺社に関することが調べられるシステムです」のようなヘルプを提示の方がよい。これにより後続する対話において「京都の渋滞状況を教えてください」といったシステムの能力を越えた質問を抑え、全体の対話の効率を上げることにつながる。一方、すでにシステムの概要を知っているユーザに対してシステムの概要を伝えるようなヘルプは無駄である。

2.2 ドメイン概念木

このような戦略のために図1のようなドメインの概念の階層を表わす木（以下ドメイン概念木）を用意する。

ドメイン概念木は、抽象度が異なる4つの層から成り、より上位の層が抽象度が高い。

システム層 システム全体の概要を表す。

機能層 システムが提供する個々の機能を表す。

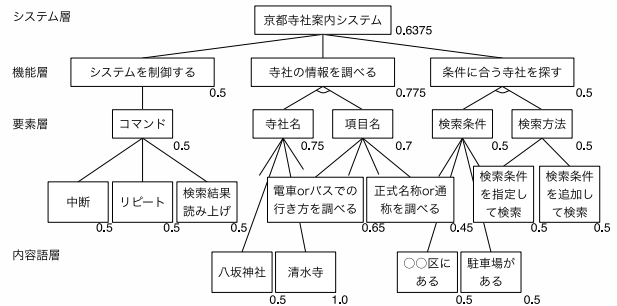


図1: 京都寺社案内システムのドメイン概念木

要素層 機能を利用するために発話に含まれるべき要素にあたる。

内容語層 各要素を構成する実際の単語やフレーズにあたる。

システム層、要素層の子ノード同士は OR の関係になる。例えば「検索方法」という機能は「検索条件を指定して検索する」か「検索条件を追加して検索」のどちらかで実現できる。

一方、機能層の子ノード同士は AND の関係になる。これは機能層の機能を利用する際には、要素としてのキーワードがすべて発話に含まれていないと利用できないことに対応する。例えば、「寺社の情報を調べる」場合は、「寺社名（清水寺）」と「項目名（電車で行けますか?）」の両方の要素が含まれないと検索できない。ただし、機能層の子ノードがひとつしかない場合は「機能」と「要素」をまとめてもかまわない。例えば、「システムを制御する」には「中断」「リピート」「検索結果読み上げ」のコマンドがあるように扱える。

それぞれの概念ノードには、ユーザがどれだけその概念を知っているかを表す既知度と、そのノードの概念をユーザが理解していない時に提示するのが適当なヘルプメッセージを保持させる。

例えば、図1の「電車 or バスでの行き方を調べる」ノードは「例えば『<寺社名>は電車で行けますか?』』と言えば<寺社名>に電車やバスで行く方法を調べられます」といったヘルプメッセージを保持している。

既知度は、0 から 1 の値をとり、0.5 より大きければその概念を「知っている」、小さければ「知らない (= ヘルプが必要)」と考える。初期値は 0.5 とし、これは「どちらかわからない」に相当する。

2.3 ユーザ知識の更新

本手法では、対話中に得られる情報から各ノードの既知度を更新する。これにより、ユーザのシステムに対する知識（以下ユーザ知識）を推定し適切なヘルプを生成する。

既知度の更新は以下の renew に従う。ただし、 n をノードとし、その既知度を $k(n)$ 、親ノードを p_n 、子ノード集合を C_n とする。また、 $p(-0.5 \leq p \leq 0.5)$ は更新の度

Dynamic Help Generation for Spoken Dialogue System by Estimating cause of user's invalid utterance: Yuichiro Fukubayashi, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno (Kyoto Univ.)

合いで更新する場合に応じて $p_u^+ (> 0)$, $p_h^+ (> 0)$, $p_u^- (< 0)$ を利用する. p の添字 u はユーザ発話からの更新を表し, h はシステムのヘルプによる更新を表す. $\lambda (0 < \lambda \leq 1)$ は更新の影響をどれだけ上位 (下位) に伝えるかを表す係数である.

```
sub renew (n, p) {
  k(n)=k(n)+p;
  if (p ≥ 0 && p_n != 0) {renew(p_n, λp);}
  if (p < 0 && C_n != 0) {
    foreach l ∈ C_n {renew(l, λp);}
  }
}
```

renew では, $p \geq 0$ の時は上位に, $p < 0$ の時は下位に向かって既知度を更新する. これは, ある概念を「知っている (知らない)」ときにはより上位 (下位) の概念も「知っている (知らない)」ととの仮定に基づく.

2.4 ドメイン概念木に基づく動的ヘルプ生成

システムは次の2つの場合にヘルプを提供する.

1. ユーザの発話が不完全でシステムの機能を利用するに至らない場合 (AND ノードの要素が足りない)
2. ユーザが沈黙した場合

まず, 2 の場合は, 無言が t 秒続いたときに生成する. その内容は, AND ノードの要素が全く得られない発話があったとみなすことで得られる.

次に 1 の場合について, 以下にユーザ発話からヘルプ生成までの流れを示す.

ユーザ知識更新 ユーザの発話から取り出したキーワードをもとにユーザ知識の更新をする. 「清水寺」「電車で行けますか?」が一発話に対するキーワードとして得られれば, 対応するノードとして $\text{renew}(\text{清水寺}, p_u^+)$, $\text{renew}(\text{電車 or バスでの行き方を調べる}, p_u^+)$ を実行する. この場合のように AND ノードの要素がすべてそろっているときはヘルプを生成する必要はない.

しかし, 「電車で行けますか?」しか得られなかったときは, $\text{renew}(\text{電車 or バスでの行き方を調べる}, p_u^+)$ だけを実行する. 「電車 or バスでの行き方を調べる」は AND ノードの要素の1つだが, もう1つの要素として「寺社名」が足りない. したがって, 寺社名を指定しなければならぬことを知らないと考えられるので $\text{renew}(\text{寺社名}, p^-)$ を実行する.

ヘルプ探索 ユーザの発話が不完全なときはヘルプを探索する. まず, 探索開始ノードを決定する. 探索開始ノードは発話に足りない AND ノードの要素に応じて決定する.

- AND ノードの要素が一部足りないときは足りない要素に対応するノード
- AND ノードの要素が全く得られないときはルートノード

先の例の場合は「寺社名」が探索開始ノードになる.

探索すべきノード (ヘルプを提供すべき概念) は, 探索開始ノードの下位 (自身を含む) でかつ既知度 0.5 未満のノードのうち一番上位にあるノードである (複数ある場合は既知度が最小のノード). システムは探索したノードが保持するヘルプメッセージを提供する.

U1: 清水寺

S1-1: 「清水寺」しか分かりませんでした

S1-2: 寺社名と一緒に聞きたい項目も言ってください

U2: 清水寺で何かおもしろいことないの?

S2-1: 「清水寺」しか分かりませんでした

S2-2: 例えば, 「清水寺の概要を教えてください。」と言えば清水寺の概要を調べられます

図 2: 対話例

ユーザ知識再更新 ヘルプによりユーザはシステムについて知識を得たことになるので, $\text{renew}(\text{提供したヘルプに対応するノード}, p_h^+)$ を実行する.

3. 実装と動作例

3.1 京都寺社案内システム

本手法を京都寺社案内システムに実装した. システムはユーザの質問に対して結果を音声で提供する. 音声認識エンジンとして Julian* を用いた. 語彙サイズは 659 である. また, 認識結果の各単語に対して信頼度を計算 [3] し, 信頼度が 0.8 以上の単語だけを用いる. データベースは, 寺社 1 件に対し正式名称や休館日など 16 のキーからなるもので 279 件分を手で作成した.

3.2 動作例

図 2 に対話例を示す. ただし, すべてのノードの既知度が 0.5 の状態から始めた.

U1 からは「清水寺」だけが得られ, AND ノードのもう1つの「項目名」が足りないので, $\text{renew}(\text{清水寺}, p_u^+)$, $\text{renew}(\text{項目名}, p^-)$ が実行される. 探索開始ノードは「項目名」になるが, 既知度が 0.5 より小さいため, 「項目名」に対応するヘルプ S1-2 を提供した後, $\text{renew}(\text{項目名}, p_h^+)$ を実行する.

U2 からも U1 と同様に $\text{renew}(\text{清水寺}, p_u^+)$, $\text{renew}(\text{項目名}, p^-)$ が実行される. 探索開始ノードは同様に「項目名」になるが, 今回は「項目名」の既知度は 0.5 を越えているので, 下位のノードから「電車 or バスでの行き方を調べる」が選ばれ S2-2 を提供し, $\text{renew}(\text{電車 or バスでの行き方を調べる}, p_h^+)$ を実行して次の発話を待つ.

このように発話内容が同じような場合でも, 対話の内容によってヘルプの内容を変化させることが可能になる.

4. まとめ

音声対話システムが扱うドメインの概念を抽象度の観点から構成しユーザのシステムに対する知識を表現した. これを利用してユーザ知識に応じたヘルプを探索し生成する方法を提案した.

謝辞 本研究の一部は, 科研費, 21 世紀 COE の支援を受けた.

参考文献

- [1] G. Gorrell, et al: Adding Intelligent Help to Mixed-Initiative Spoken Dialogue Systems. Proc. ICSLP, 2002.
- [2] B. A. Hockey, et al: Targeted Help for Spoken Dialogue Systems: intelligent feedback improves naive users' performance. Proc. EAACL, 2003.
- [3] 駒谷 和範, 河原 達也: 音声認識結果の信頼度を用いた効率的な確認, 誘導を行う対話管理. 情報処理学会論文誌, vol.43, No.10, pp.3078-3086, 2002.

*<http://julius.sourceforge.jp/>