

# 誤認識の修復機能を備えた音声対話システムの構築

矢野 浩利<sup>†</sup> 北岡教英<sup>†</sup> 中川 聖一<sup>†</sup>

<sup>†</sup> 豊橋技術科学大学 情報工学系

e-mail: {hyano, kitaoka, nakagawa}@slp.ics.tut.ac.jp

## 1 はじめに

近年、音声認識をベースとしたインターフェースを備えたシステムの実用化が進んでいる。しかしながらコンピュータと人間が音声を通じてコミュニケーションをはかる場合、誤認識は避けられない。また、現在はインターフェースが未熟であるために、誤認識からの回復が困難である。一般に、対話システムでは誤認識したまま対話を進めるのを防ぐため確認発話をを行う。しかし、ユーザの発話を受け取る度に確認発話をを行うとタスク達成までのターン数が増え、ユーザにとても煩わしいものである。

本稿では、確認発話を行わない対話手法を提案する。確認発話を行わないことで、ユーザの発話を誤認識したまま対話を進めてしまう恐れがある。そこで発話の理解候補を複数保持させることで、曖昧性を残したまま対話を進めていく、理解候補からユーザの正しい意図を導くようなシステム応答を行う。ユーザから効果的な情報を得ることで、システムは対話を進めていくうちに誤認識からの自然な回復を目指す。

関連研究はこれまでにも多く行われている。伊藤らは以前のユーザ発話情報として複数の理解候補を保持し、認識信頼度と対話履歴を用いた音声言語理解手法を提案しており、単純に音声認識結果の第一候補を最優先するよりも、提案手法で10%程度の理解性能の向上が得られたと報告している[3]。東中らは複数の文脈と音声認識結果候補から得られる複数理解候補を、統計的モデルを用いることで順序付けする手法を提案し、複数の対話状態を保持することの優位性を示している[4]。駒谷らは発話の内容語の音声認識結果の信頼度を用いた効果的な確認・誘導を行う方法を提案しており、信頼度によって発話の受理・確認・棄却を行い、必要な場合にのみ効率よく確認発話を行えることを示している。また内容語がうまく認識できなかった場合に、適切にユーザの誘導を行う手法も提案している[5]。堂坂らは、効率的な対話を実現するための対話制御法として、デュアルコスト法を提案した。デュアルコスト法は、対話における確認コストと情報伝達コストの和を最小化する原理に基づいた方法で、この手法において無駄な対話を回避することができたと報告している[6]

本稿では複数理解を保持した状態で、できるだけすみやかに理解を収束させつつ、ユーザに矛盾した応答として自然性を損なうことのない応答生成法を提案する。

## 2 複数理解候補を保持する 音声対話理解方法

一般に、音声対話中ではシステムによる誤認識が生じる。誤認識から回復できる対話戦略として、確認発話がある。確認発話をを行う対話は図1のようなものがある。例中で確認発話を行わなければ、対話ターン数は大幅な削減が期待できる。

しかし、「豊橋市」を「豊川市」と誤認識したまま対話が進行し、対話中にこれまでの理解と現在のユーザ発話において矛盾が起こり対話が破綻したり、最終的に誤った理解結果に至る。

これは、音声認識の結果の第一候補のみを信じて対話を進めることに起因する。各認識において、複数得られる認識候補を有効に用いれば、この危険は低減される。

そこで図2のように、認識の複数候補に基づいたユーザ発話の理解候補を複数持ったままシステムは対話をを行う。各行が1つの理解候補に対応し、それぞれの候補には認識結果より信頼度の値を持たせる。信頼度の値は、通常の連続単語認識結果における対象単語区間の対数尤度と、別に動作させる音節列認識結果の同区間の対数尤度の差(対数事後確率)をとったものを用いる。また、図中で理解候補中の()で囲まれた単語は、ユーザは発話していないがシステムの知識より候補が予想できている状態を示す。この例の場合、「静岡県清水区」という認識結果に基づいた理解では、県名は自動的に決まっていることを示す。

ユーザ発話の度に、ユーザ発話の複数認識候補と現在の各理解候補を組み合わせることで最新のシステムの理解を構成する。そのため、理解候補は対話を繰り返すことで増加していく。

複数理解候補を持ったまま対話を進めていくと、理解候補中には新たに得られるユーザ発話を考慮すると、理解に一貫性がなく矛盾している候補が出現することがある。このような理解候補については、これまでの対話理解が誤っているか、最新のユーザ発話の認識が誤っていると考えられる。そのため、このケースに該当する理解候補については信頼度を下げることにする。すると、対話を進めていくことで過去のユーザ発話の認識結果を統合して矛盾のない理解候補は信頼度が高くなり、過去の認識結果において正解が最尤でなくても、後に誤りからの回復が可能になると考えられる。

System	目的地を設定してください
User	豊橋市に行きます
System	豊川市でよろしいですか？
User	違います。豊橋です
System	豊橋市でよろしいですか？
User	はい
System	町名を入力してください

図1: 毎回確認発話をを行う対話例

User 発話 静岡県のコンビニに行きたいです

システムの理解候補				
ランク	県名	市名	町名	目標物
1	静岡県		-	コンビニ
2	(静岡県)	静岡市清水区	-	コンビニ
3	滋賀県	-	-	コンビニ

図 2: 1 つのユーザ発話からの複数理解候補生成の例

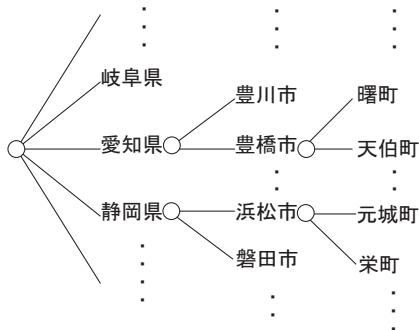


図 3: 地名入力におけるカテゴリの上位・下位関係

### 3 システム応答の決定手法

2 節のような複数理解候補からのシステム応答発話の決定方法を検討する。対話の目的は、ある正しい 1 つの理解候補を最終的に決定することである。従って、ある時点の複数候補からできるだけ理解候補を絞り込むことが望ましい。しかし理解候補を絞り込むために、ある候補を前提に応答することは別の候補が正解であった際に対話に矛盾が生じることがある。そのため、各理解候補においてなるべく矛盾のない自然な応答を行うことが望ましい。本稿では、理解候補中から正しいであろう候補を絞り込める可能性が高く、かつユーザにとって自然な応答である可能性の高い、複数候補の理解にできるだけ共通するシステムの応答を目指す。

#### 3.1 応答が理解候補を絞り込む度合いの尺度

まず、現在の理解候補を絞り込める応答(質問)かどうかの尺度を考える。本稿で扱う対象は、カーナビゲーションを想定した目的地設定タスクとする。目的地設定タスクの地名のデータは図 3 のように上位・下位の関係がある。

システムが認識結果の豊橋市と浜松市で迷っている状態を考えると、システムが「県名を教えてください」という質問を行うことで、正解が「豊橋市」である場合は「愛知県」「浜松市」が正解であった場合には「静岡県」とユーザが発話することが予想される。よってこの質問により理解候補中からの絞込みができると予想できる。このように保持する理解状態によっては、システムの質問によって現在の理解候補を絞り込むことができ、さらに新しい情報を得ることができる。そこで、行う質問によって現在の複数理解候補をどの程度絞り込むことができるかという尺度を設ける。

現在の理解候補集合  $N$  において、システムが質問  $q$  を行った場合にユーザから回答  $a$  が得られる確率を以下のようにを求める。

$$P(a|N, q) = \sum_n P(a|n, q) \cdot P(n) \quad (1)$$

ここで理解候補  $n$  が正解である確率  $P(n)$  は信頼度を考慮して計算する必要があるが、理解候補の信頼度は 2 節で述べたように連続単語認識と音節列認識の対数尤度の差を用いているため、信頼度の尺度  $Conf(n)$  を元に理解候補が正しい確率を求めるには信頼度の値と正解確率の関係について統計を取る必要がある。今回は擬似的に  $P(n) = Conf(n) / \sum_{m \in N} Conf(m)$  を用いた。そして、 $n$  において  $a$  が矛盾することを  $I(a, n) = 1$ 、矛盾しないことを  $I(a, n) = 0$  として、候補が絞りめるスコアを以下のように定義する。

$$Score_n(q) = \sum_n \{1 - P(n)\} \sum_a I(a, n) \cdot P(a|N, q) \quad (2)$$

#### 3.2 応答が理解候補に共通する度合いの尺度

システム応答を決定する方法として、応答が複数候補について共通である尺度を設ける。図 2 の例について考えると、ここでシステムが「市名を教えてください」という応答を選択すると、(実際には誤った理解であるが) 理解候補の 2 位ではすでにユーザが発話した内容について 2 度尋ねることに相当してしまう。また、「静岡県でよろしいですか?」とシステムが応答すると 3 位の理解候補にとっては滋賀県と理解しているため矛盾している。このケースのようにシステムの行う応答が、各理解候補について考えたときになるべく矛盾しているものは避けたい。そこで応答を選択する尺度として、理解候補に共通である尺度を用いる。

$$Score_c(Q) = \sum_{n \in N} f(Q, n) \quad (3)$$

$$f(Q, n) = \begin{cases} Conf(n) & (Q \text{ が理解状態 } n \text{ で矛盾しない}) \\ 0 & (\text{それ以外}) \end{cases} \quad (4)$$

#### 3.3 最終的なシステム応答の選択

上で定義した尺度において、2 つの値の重みつき和が最大となる質問  $Q$  をシステムの応答として選択する。

$$Question = \operatorname{argmax}_Q w_n \cdot Score_n(Q) + w_c \cdot Score_c(Q) \quad (5)$$

本稿では、次のタイプの質問に対してこのスコアを計算し、最もスコアの高かったものをシステムの応答とした。

- ・「県名を教えてください」
- ・「市名を教えてください」
- ・「町名を教えてください」
- ・「目標物を教えてください」
- ・県名に対する確認発話(「 県でよろしいですか?」)
- ・市名に対する確認発話(「 市でよろしいですか?」)
- ・町名に対する確認発話(「 町でよろしいですか?」)
- ・目標物に対する確認発話(「 × × でよろしいですか?」)

上の 4 つは新たな情報を要求するものであり、複数の理解候補に共通に使える傾向がある。一方、下の 4 つは確認発話であり「愛知県でよろしいですか？」という応答である。候補を 1 つに絞ることを目的とするもので、複数候補において共通する割合は小さい。また、この 8 つとは別にタスク完了のため最終的に確認する発話をを行う。最終的な確認は、上の 8 つの質問のスコアが一定閾値以下だった場合、これ以上有効な応答はないと考える。

## 4 言い直し検出を用いた対話処理

一般にシステムの誤認識に気付いた際、ユーザは言い直し発話をを行う [1]。言い直し発話とは、図 1 のユーザの 1 発話目と 2 発話目のように、過去に発話した内容と同じ内容を繰り返すことによってシステムの誤認識を伝えるものである。ユーザの言い直し発話を検出できれば、システムは自身の誤認識に気付くことができ回復が容易になる

言い直し判定には DP マッチングによる手法を用いた。マッチした区間を用いて単語単位で言い直し判定を行うことで理解候補中のどのカテゴリが訂正されたかの判定が可能である。

言い直しを検出したときの効果は、以下の通りである。

- システムの確認発話に対してユーザの言い直しを受け取った場合、提示した理解が誤っていることになる。よってその候補の信頼度を下げる

- 言い直しされた単語と直した単語で同じことを発話していると考えることができる。そのため、2 つの単語の認識結果において共通して出現した認識結果は信用できるため信頼度を上げる

信頼度の上下度合いは現在は先見的に与えているが、将来的には言い直し判定で得られるスコアを言い直し判定の正解率に統計に基づいて変換することが望ましい

## 5 音声対話システムの構築

複数理解候補を保持し、提案手法によるシステム応答決定、言い直し発話検出を備えたシステムを構築する。

### 5.1 対話システムの構成

本音声対話システムの構成図を図 4 に示す。ユーザの発話は、本研究室で開発された音声認識器 SPOJUS[2] と、DP マッチングによる言い直し検出部に送られ、それぞれの結果が理解更新部に送られる。そこで、過去のユーザ発話を踏まえた現在の複数理解候補と、得られた複数の認識結果、言い直し検出結果を基に最新の理解候補へと更新する。最新の理解候補に基づき、提案したシステム応答の選択手法により応答が選ばれ、音声合成器を通じてユーザに発話される。この動作をシステムが要求する情報（県名、市名、町名、目標物名）すべてが保持されるまで繰り返す。

### 5.2 提案手法による対話例

構築したシステムで得られた対話の一部を図 5 に示す。例では、ユーザの最初の発話 (User1) に対する理解候補を絞り込むために、システムは県名を尋ねる質問を行った (System2)。それに対するユーザの 2 発話目 (User2) の認識結果より、1 発話目の「調布」という候補が誤っている可能性が高いと判

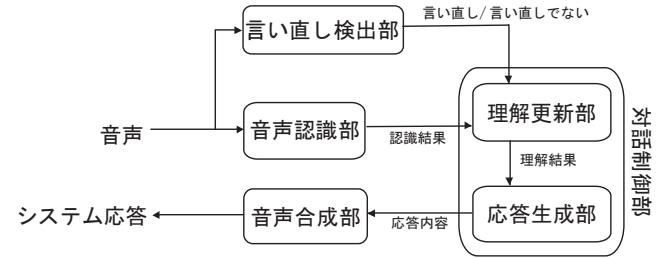


図 4: 対話システムの構成図

断した。ここで認識結果にシステムの尋ねた市名でない候補（「見付」）が出現しているが、これはユーザの返答として考えられないものとし、理解候補から除外する。まだ、「大須」と「豊橋市」で迷っているためシステムは市名を尋ねる応答を行った (System3)。このときのユーザの 3 発話目は 1 発話目の言い直しと判定され、それぞれの認識結果の N-best 結果を統合すると豊橋市のほうが正しいであろうと理解した。この例では結果としてユーザと対話を繰り返すことで、誤認識からの回復ができた。

## 6 評価実験

### 6.1 実験条件

提案した対話戦略の評価を目的に設定タスクを想定して行った。評価基準としては、ユーザがシステムを使用した際のタスク完了までのターン数（システムの応答に対してユーザが発話することに 1 ターンとする）を計測する。評価のために大量の発話データが必要となるため、今回は擬似的にユーザ発話を作りシミュレーションを行った。本稿で提案した複数理解候補を保持し、提案尺度により応答を決定する対話システムと、ユーザの発話ごとに確認発話をを行う対話システムについて比較を行った。

シミュレータのタスク完了までの対話の流れとしては次のようになる。

1. シミュレータがタスク完了条件となるユーザの目的地をランダムで決定
2. システムが最初の応答「目的地を設定してください」を発話
3. シミュレータはシステムの応答に忠実な返答を行うとし、タスク完了条件に基づいて発話（テキスト）を生成
4. システムがユーザ発話（シミュレータの発話）の誤認識を含む認識結果とその信頼度を擬似的に作成
5. システムが認識結果とこれまでの理解から、最新の理解候補を作成
6. タスク完了条件を満たしていないければ、システムが最新の理解候補からシステム応答を生成して 3 へ

発話の信頼度については正解の認識候補のほうが誤った候補より高い信頼度が割り当てられるようにした。提案手法における 5 のシステム応答の生成には、3 節の方法を用いる。毎回確認発話をを行う対話システムについては、ユーザの発話

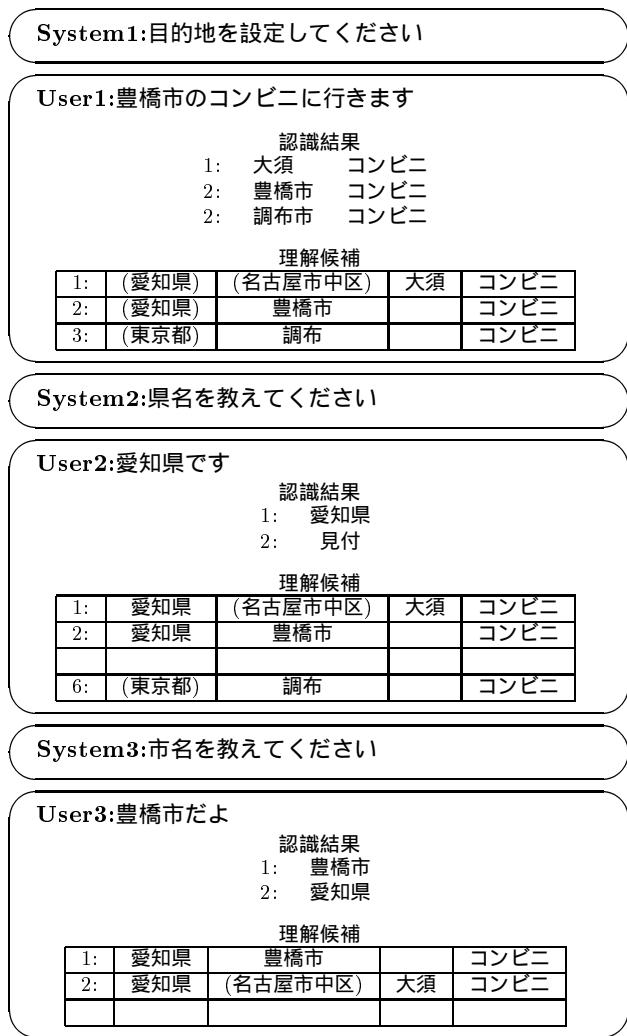


図 5: 構築したシステムの対話例

に対して確認発話をを行う。確認発話に対して肯定が得られた場合は、次の応答は空いているカテゴリについて尋ねる。

どちらの手法においても、確認発話で誤認識を確認した際には、ユーザは必ず言い直しによる訂正を行うこととする。タスク完了条件は全てのカテゴリに対して、ユーザの肯定発話が得られた時とする。また、本実験ではタスク失敗と見なすターン数を 16 とし、これを上限とした。

## 6.2 実験結果

実験結果を図 6 に示す。結果から、提案手法においてタスク完了までにかかる平均ターン数の減少が見られ効果が確認できた。しかし、提案手法におけるタスク達成率は認識率 80%において 72% であり、低いものであった（毎回確認の方法では 100%）。原因としては、理解候補更新のルールが未熟なため効果的な応答をシステムが出来ず、対話が終わらない例が存在したことが挙げられる。

今後は理解の高精度化が必要である。また、対話が終わらなくなるないように適宜、確認発話をを行う対話戦略への切り替えなどの対応を取る必要がある。

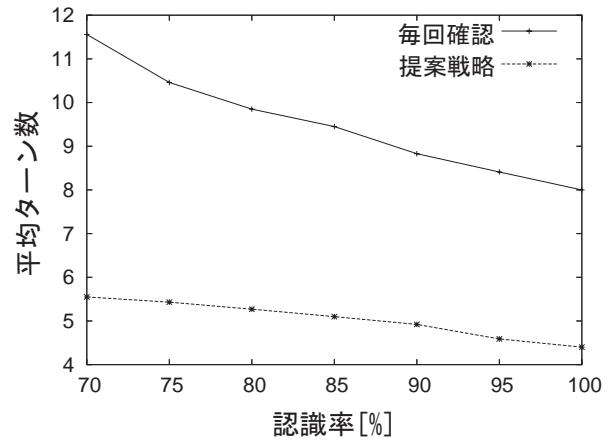


図 6: ターン数による評価

## 7 まとめ

本稿では、音声対話システムにおける、効率的で自然な誤認識回復機能を有する対話戦略を提案した。

ユーザ発話を得た際に確認発話を行わないことで、ユーザの発話を完全に理解できない問題があるが、システムが発話の理解候補を複数保持することで理解を一意に決めることなく対話を進めることで、対話中での誤認識から回復を目指した。システムが理解候補から正しい候補を導くための有効なシステム応答を決定するために、現在の理解候補を絞り込む尺度と、理解候補に共通する応答の尺度を求める、その和により最終的なシステム応答を決定した。

また、言い直し検出を対話システムで用いることでシステムの誤認識からの回復を容易にする手法を示した。

提案手法を備えた音声対話システムを構築し、実際の対話例において本手法が有効性が見れる対話例を得ることが出来た。

理解候補更新のルールをうまく作ることが出来れば、提案手法を用いることで効率的な対話が期待できる。

## 参考文献

- [1] 北岡 教英, 角谷直子, 中川 聖一:「音声対話システムの誤認識に対するユーザの繰り返し訂正発話の検出と認識」, 電子情報通信学会論文誌, Vol.J87-D-II No.7(2004.7)
- [2] 日本語連続音声認識システム SPOJUS-SYNO, <http://www.slp.ics.tut.ac.jp/SPOJUS/>
- [3] 水野智士, 高木浩吉, 小暮悟, 甲斐充彦, 伊藤敏彦, 小西達裕, 伊藤幸宏:「頑健な意味理解のための音声認識信頼度と対話履歴を利用した発話意図推定手法」, 情報処理学会研究報告, 2005-slp-55, pp77-82(2005.2)
- [4] 東中竜一郎, 中野幹生, 相川清明:「複数文脈を用いる音声対話システムにおける統計モデルに基づく談話理解法」, 情報処理学会研究報告 (SLP-45-17), pp.101-106(2003.2)
- [5] 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, pp3078-3086(2002.10)
- [6] 堂坂浩二, 安田宜仁, 相川清明:「対話コスト最小化に基づく効率的対話制御」, 電子情報通信学会技術研究報告, NCL2001-34, pp93-100(2001.7)