

## 誤認識の修復のための自然で効率的な音声対話戦略

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

<sup>†</sup> 豊橋技術科学大学 〒441-8580 愛知県豊橋市天伯町雲雀ヶ丘1-1

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

あらまし 音声対話におけるシステムの誤認識を対話を通じて回復する対話戦略を提案する。確認発話を用いずに回復するために、各対話ターンにおいて複数の理解候補を保持し、対話を通して最適な仮説を探索する。対話戦略として、理解候補を収束させる効率性の尺度と、理解候補と矛盾しない無矛盾性の尺度を応答選択に用いる。また、理解候補の更新には“繰り返し訂正発話”の検出も用いる。これらの手法を用いた音声対話システムを構築し、自然に誤認識から回復できた対話例を得ることができた。また、ターン数によるシミュレーション実験によって本対話戦略が効率的であることも示した。

キーワード 音声対話システム, 誤認識, 効率的な応答, 自然な応答

## A Spoken Dialog Strategy for Natural and Efficient Recovery from Misrecognition

Norihide KITAOKA<sup>†</sup>, Hirotohi YANO<sup>†</sup>, and Seiichi NAKAGAWA<sup>†</sup>

<sup>†</sup> Toyohashi University of Technology, Aichi, 440-8580 Japan

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

**Abstract** We proposed a novel dialog strategy which can recover from misrecognition through a spoken dialog. To recover from the misrecognition without confirmation, our system kept multiple understanding hypotheses at each turn and ‘searched’ a globally optimal hypothesis across user’s utterances in a whole dialog. As for a dialog strategy, we introduced a new criterion based on ‘efficiency for convergence’ and ‘consistency with understanding hypotheses’ to select an appropriate system response. We also proposed to adopt the repetition utterance detection to update the understanding hypotheses. We developed a spoken dialog system using these techniques and showed some dialog examples in which misrecognition is naturally corrected. We also showed that our strategy is efficient in terms of the number of turns.

**Key words** Spoken dialog system, misrecognition, efficient response, natural response

### 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行う。しかし、ユーザの発話を受け取る度に確認発話を行うとタスク達成までのターン数が増え、ユーザにとっ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き、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

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 毎回確認発話を行う対話例

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]。萱坂らは、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、対話履歴に矛盾せず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保持したままシステムは対話を行う。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括弧で囲まれた単語は、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、「豊橋市」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。

複数候補を保持して対話を進めていくことは連続音声

User 発話 豊橋市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	(愛知県)	豊橋市	—	コンビニ
2	富山県	—	—	コンビニ
3	(愛知県)	豊川市	—	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

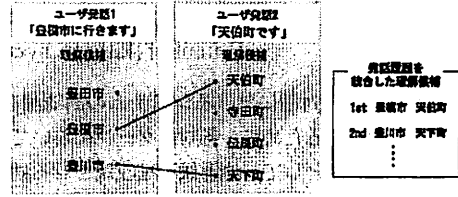


図3 複数理解候補を持った対話理解の例

認識のサーチ問題と似た考えを用いることができる。図3のように、タスク完了までの分割して発話された情報を逐次認識・理解していくことで、これまでの履歴全ての発話を通じた最適解を得ることとえられる。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とする。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。

確認発話を行わないことで、対話ターン数は大幅な削減が期待できる。しかし、システムが対話中においてうまく振舞わないと一向に誤認識・理解からの回復ができない。そこで以降では、複数理解候補を持った中でシステムの応答決定法と正しい判定を用いた方法について検討し、対話の円滑化を目指す。

## 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。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み、最終的に1つの候補を得ることが望ましい。

関連研究として、音声認識結果の信頼度に基づいて受理・確認・棄却の決定することで確認発話を行うか否かの決定を行っている手法[5]や、ある質問を行うことで見込めるエントロピーより質問を決定する方法もある[7]。

本節では、システムの情報を要求する質問(確認発話含む)からシステム応答を選択する提案手法について説明する。その方針として、現在の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(効率性)、かつユー

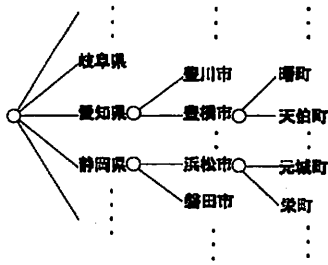


図4 地名入力におけるカテゴリの上位・下位関係

ず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(自然性)ものとする。

### 3.1 応答が理解候補を絞り込む度合いの尺度

システムが複数候補で迷っている状態から、なるべく少数の候補に絞り込むことができる、自らにとって有益な情報が期待できる質問が好ましい。

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4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」とユーザー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り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ーから回答  $a$  が得られる確率を以下のように求める。

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

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、候補が絞り込めるスコア(効率性尺度; Efficiency score) を以下のように定義する。

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

### 3.2 応答が理解候補に共通する度合いの尺度

3.1節の尺度で選んだ応答は、誤認識した結果から得られた場合には、ユーザーにとっては対話上不自然で、また、システムとしては自らの誤認識をユーザーに伝えてしまうことになる。ここで、現時点の複数候補のいずれかで正

しい理解を表現されていると仮定すると、多くの理解候補と矛盾しない応答が、ユーザーにとって一貫性があり自然で、かつ誤解を悟られない応答であるとする。

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、理解候補の1位と3位ではすでにユーザー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、「愛知県でよろしいですか?」とシステムが応答すると(実際には誤った理解であるが)2位の理解候補にとっては富山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、各理解候補に対してなるべく矛盾しているものは避けたい。そこで、 $n$  において  $q$  が矛盾しているものを  $I(q, n) = 1$ 、矛盾しないことを  $I(q, n) = 0$ 、理解候補  $n$  の信頼度を  $Conf(n)$  として、理解候補に共通である尺度(=無矛盾性尺度; Consistency score) を以下のように定義して用いる。

$$S_c(q) = \sum_{n \in N} (1 - I(q, n)) \cdot Conf(n) \quad (3)$$

### 3.3 最終的なシステム応答の選択

最終的には、上で定義した2つの尺度の重みつき和が最大となる質問  $\hat{q}$  をシステムの応答として選択する。

$$\hat{q} = \underset{q}{\operatorname{argmax}} w_c \cdot S_c(q) + w_e \cdot Se(q) \quad (4)$$

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」
- 県名の確認発話(「〇〇県でよろしいですか?」)
- 市名の確認発話(「△△市でよろしいですか?」)
- 町名の確認発話(「□□町でよろしいですか?」)
- 目標物の確認発話(「××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行う。

## 4. 言い直し検出を用いた対話処理

3.節に示したように、システムはなるべく自身の理解が未熟であることを悟られないように対話を行う。しかし、確認発話がどうしても必要な場合や、選ばれた質問が結果的に矛盾したものになった場合などに、誤解が伝わってしまうことがある。そういった際に、誤解からの回復を容易にするために訂正発話をうまく扱う必要がある。

一般にシステムの誤認識に気付いた際にユーザーは言い直し発話を行うことが考えられる。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。システムが言い直しを検出すると、次のような効果が期待できる。

表 1 単語単位の DP マッチングの判定性能

再現率	適合率	F 値
0.835	0.841	0.837

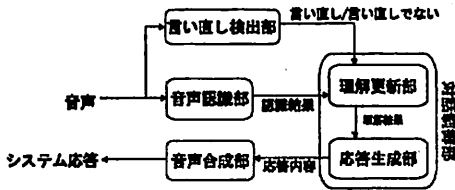


図 5 対話システムの構成図

● 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言い直した単語で同じことを発話していると考えられることができる。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

言い直しを単語単位で判定することで明確に言い直された単語を知ることができ、さらに誤って理解していた単語について知ることが出来る。ここでは、DP マッチングを用いた言い直し判定 [1] を、マッチング区間に現れる単語を言い直し単語とするように、単語単位の判定に拡張して用いる。

単語単位で行った DP マッチングの判定性能を表 1 に示す。ここで、DP マッチングの結果、対応する言い直し区間が一部でも言い直し区間として検出されたものを正解とした。高精度で発話のどの単語が言い直されているか判定できることが期待できる。

言い直し判定の効果として挙げた、判定結果による信頼度への影響は、DP マッチングの結果得られるスコアを元に行う。現在は信頼度の上下度合いは先見的に与えているが、将来的には統計的に、言い直し検出で得られるスコアを判定の正解率に変換することが望ましい。

## 5. 音声対話システムの構築

### 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 5 に示す。

ユーザの発話を得ると、音声認識器 SPOJUS [2] による認識結果と、過去のユーザ発話との DP マッチングによる言い直し判定結果を対話制御部へ送る。対話制御部中の理解更新部がこれらを用い、過去の複数の発話理解候補から最新の理解候補を生成する。応答生成部は、この理解候補に基づき、提案したシステム応答選択手法によって応答を選んで音声合成部へ送る。

### 5.2 タスク

本システムで扱うタスクは、図 6 に示すようなカーナビゲーションを想定した目的地設定とする。タスク達成に必要な情報のカテゴリは県名、市名、町名、目標物である。

対話はシステム主導とするが、必ずしも全てのカテゴリについて質問を行うわけではない。これは、例えば市

System:目的地を仮定してください。

User:天伯町に行きたいです。

System:市名を教えてください

User:豊橋市です。

System:目標物を教えてください？

User:豊橋技術科学大学です

System:豊橋技術科学大学でよろしいですか？

User:はい。

図 6 システムの対話例

名“豊橋市”がわかっているれば県名“愛知県”も自動的に決定されるためである。実際に行うべきか否かの決定は、提案手法の応答選択の結果による。システムが行う応答は 3.3 節で挙げた 8 つである。

システムの想定するユーザ発話としては、1 発話において複数のカテゴリを含む発話を許す。例を以下に示す。

- 「愛知県に行きます。」
- 「豊橋市の天伯町です。」
- 「静岡県にあるガソリンスタンドに行きたいです。」

また、本システムで扱う認識辞書のサイズは、情報カテゴリに分類される単語(県名 5, 市名 22, 町名 139, 目標物 29)を含む約 300 単語である。県名, 市名, 町名には上下関係があり、また目標物についても、一部のものについては同様の関係がある。

### 5.3 提案手法による対話例

構築したシステムを用いて得られた対話を図 7, 図 8 に示す。

図 7 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方が正しいであろうと理解した。

図 8 では、ユーザの最初の発話 (User1) で得られた理解候補から、システムの知識から予想はつくがまだ埋まっていないカテゴリである市名について尋ねた (System2)。その結果得られたユーザの 2 発話目による理解結果より、台東区と文京区のどちらかと考え、信頼度の高い台東区について確認発話を行った (System3)。するとユーザの訂正発話 (User3) が得られたため文京区が正しいと判断し、2 位の候補との信頼度の差を考慮した結果、最終的な確認に至った。

この 2 つ例では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。

表 2 応答戦略の平均ターン数およびタスク達成率による比較

想定単語認識率	毎回確認		尤度により確認		提案手法	
	平均ターン数	タスク達成率	平均ターン数	タスク達成率	平均ターン数	タスク達成率
90 %	8.98	100 %	6.08	97.4 %	5.88	97.4 %
80 %	9.97	100 %	6.52	95.9 %	5.87	95.5 %
70 %	11.56	100 %	7.16	91.3 %	4.66	91.1 %
60 %	13.26	100 %	8.11	88.4 %	5.71	88.8 %

System1:目的地を設定してください

User1:豊橋市のコンビニに行きます

認識結果

- 1: 大須 コンビニ
- 2: 豊橋市 コンビニ
- 2: 岡布市 コンビニ

理解候補

1:	(愛知県)	(名古屋市中区)	大須	コンビニ
2:	(愛知県)	豊橋市	—	コンビニ
3:	(東京都)	岡布	—	コンビニ

System1:目的地を設定してください

User1:東京ドームです

認識結果

- 1: 東京 台東
- 2: 東京
- 2: 東京ドーム

理解候補

1:	東京都	(台東区)	台東	—
2:	東京都	—	—	—
3:	(東京都)	(文京区)	(後楽)	東京ドーム

System2:県名を教えてください

User2:愛知県です

認識結果

- 1: 愛知県
- 2: 見付

理解候補

1:	愛知県	(名古屋市中区)	大須	コンビニ
2:	愛知県	豊橋市	—	コンビニ
6:	(東京都)	岡布	—	コンビニ

System2:市名を教えてください

User2:文京区だよ

認識結果

- 1: 文京区
- 2: 台東区

理解候補

1:	東京都	台東区	台東	—
2:	(東京都)	文京区	(後楽)	東京ドーム
3:	東京都	台東区	—	—
4:	東京都	文京区	—	—

System3:市名を教えてください

User3:豊橋市だよ

認識結果

- 1: 豊橋市
- 2: 愛知県

理解候補

1:	愛知県	豊橋市	—	コンビニ
2:	愛知県	(名古屋市中区)	大須	コンビニ

System3:台東区でよろしいですか？

User3:文京区です

認識結果

- 1: 文京区

理解候補

1:	(東京都)	文京区	(後楽)	東京ドーム
2:	東京都	文京区	—	—

System3:東京都 文京区 後楽 東京ドームでよろしいですか？

図 7 構築したシステムの対話例 1

## 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地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。評価のためには種々の条件で大量の発話データが必

図 8 構築したシステムの対話例 2

要となるため、今回は擬似的にユーザ発話と認識結果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手法と、ユーザの発話ごとに確認発話を行う手法、尤度により確認を行ったり行わなかったりする手法を比較した。

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) ユーザ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。シミュレータは以下のプロセスでこの目的地の設定を完了しようとする。

(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応答(テキスト)を生成

(4) 疑似認識器がユーザ発話(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

上記のステップ(4)では、1-best 単語認識率  $R$  を60%から90%の範囲で事前に設定した。認識器の振舞いを正確にシミュレートするため、語彙数に従って、町名と県名はそれぞれ認識率  $1.1 \times R$  と  $0.9 \times R$  に設定した。2位以下の候補に正解が現れる確率は設定した認識率に関係なくそれまでに正解が現れなかった確率の50%とした。これは、出力される  $N$ -best が1つ増えるごとに正解候補が  $N$ -best 中にならぬ確率が半分になることに相当する。また、確認発話に対する肯定発話は認識率100%とした。

提案システムでは、すべての応答(質問)のスコアが閾値より低かった場合、最終の確認発話をする。そしてユーザが肯定的な応答をすれば対話は終了する。ステップ6では3.節で説明した方法を用いる。毎回確認する方法ではユーザの新しい情報の入力度に確認を行い、ユーザの肯定の応答の次には新しい情報の入力を求める。認識尤度により確認発話を行う方法では、認識尤度がある閾値より低い場合には確認発話を行う。この場合、確認されずに対話が終了するカテゴリがありうるが、これらのうち1つでも誤りがあればタスク失敗とみなす。

いずれに手法においても、シミュレータは誤認識を繰り返し発話によって訂正することとした。訂正発話の認識率も上記に従った。

対話のターン数が16を越えた場合、失敗とみなした。1000回の対話シミュレーションを行ってその平均ターン数により評価を行った。

## 6.2 実験結果

実験結果を図9および表2に示す。我々のシステムの実際の認識率が70%程度であったことから、図9は単語認識率70%を想定した場合の結果とした。また、ある程度のタスク達成率が得られるように、提案法では  $w_e$  と  $w_c$  の、尤度による方法では尤度閾値を調整した。提案手法によりターン数が減少したことが分かる。しかし、タスク達成率は認識率100%になっていない。原因としては、理解候補更新のルールが未熟なため、たまたま低い信頼度を得た候補が上位にランクされ、ユーザからの新たな情報を受け付けても一向に順位が変わらないなどが挙げられる。その結果、理解候補から考えられるシステム応答も有効なものを選択されず、対話をうまく制御できなかったことが挙げられる。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ように適宜、確認発話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。

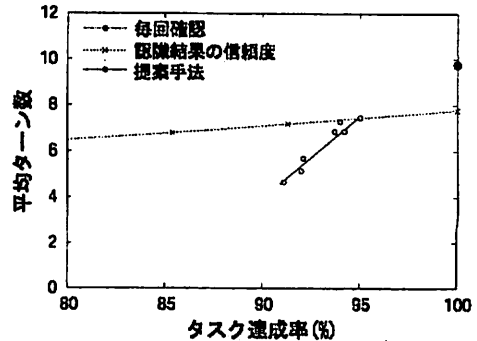


図9 ターン数による評価実験結果(単語認識率=70%)

## 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。

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、現在の理解候補を効率良く絞り込む尺度と、理解候補にできるだけ共通するユーザにとって自然な応答の尺度を求め、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できた。また、対話のターン数も少なくできることをシミュレーション対話によって示した。

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

### 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(D-II), Vol. J87-D-II No. 7, 1441-1450, 2004.
- [2] A. Kai and S. Nakagawa. "A frame-synchronous continuous speech recognition algorithm using a top-down parsing of context-free grammar," ICSLP92, pp.257-260, 1992.
- [3] 水野智士, 高木浩吉, 小暮悟, 甲斐彦彦, 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.
- [4] R. Higashinaka, N. Miyazaki, M. Nakano, K. Aikawa. "Evaluating discourse understanding in spoken dialogue systems," EUROSPEECH-2003, pp. 1941-1944, 2003.
- [5] 駒谷和隆, 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.
- [6] 盤坂 浩二, 安田 直仁, 相川 啓明. "システム知識制限下での効率的音声対話制御法," 自然言語処理, Vol. 9, No. 1, pp. 43-63, 2002.
- [7] T. Mitsu and T. Kawahara. "Speech-based Information Retrieval System with Clarification Dialogue Strategy," Proc. HLT/EMNLP, pp. 1003-1010, 2005.