

WOZシステムのログ情報を利用した事例ベース音声対話システムの開発

村尾 浩也^{†,††} 河口 信夫^{††,†††} 松原 茂樹^{††,†††} 山口由紀子^{†††} 稲垣 康善^{†††}

† 三洋電機(株) デジタルシステム研究所 〒 573-8534 大阪府枚方市走谷 1-18-13

†† 名古屋大学統合音響情報研究拠点 〒 464-8603 名古屋市千種区不老町

††† 名古屋大学情報連携基盤センター

†††† 名古屋大学大学院工学研究科

E-mail: †mura@hr.hm.rd.sanyo.co.jp

あらまし 筆者らはこれまで、柔軟な対話制御を行うことを目指し、対話事例を利用する音声対話制御手法を提案してきた。今回我々は、WOZ (Wizard of OZ) システムと人間との対話事例を事例データとして利用する新しい対話システム構築手法を提案するとともに、その手法に基づいて構築した情報検索向け音声対話システムの概要について述べる。WOZ システムとの対話事例を使用することによるメリットとして、1) WOZ システムのログに埋め込まれた、発話から検索式への変換あるいは検索結果に基づく応答発話生成、などの知識を対話システムの知識として利用できること、2) 実際の使用環境により近い、つまり、人間対機械の対話に近い状況下の対話事例を収集できること、3) 事例データ構築に必要な発話の書き起こしの手間を削減できること、などがあげられる。本稿では、まず、事例に基づく対話処理モデルについて述べ、WOZ システムを用いた事例データ構築手法について提案する。その後、その手法を用いて行ったデータ収集について述べ、最後に、収集したデータに基づいて動作する情報検索向け音声対話システムについて述べる。

キーワード 音声対話、音声認識、Wizard of OZ、意図理解、応答生成、自動車、対話コーパス

Example-based Spoken Dialogue System using WOZ System Log

Hiroya MURAO^{†,††}, Nobuo KAWAGUCHI^{††,†††}, Shigeki MATSUBARA^{††,†††},Yukiko YAMAGUCHI^{†††}, and Yasuyoshi INAGAKI^{††††}

† Digital Systems Development Center, SANYO Electric Co., Ltd.

Hashiridani 1-18-13, Hirakata-shi, Osaka, 573-8534 Japan

†† Center for Integrated Acoustic Information Research, Nagoya Univ.

Furo-cho, Chikusa-ku, Nagoya-shi, 464-8603 Japan

††† Information Technology Center, Nagoya Univ.

†††† Graduate School of Engineering, Nagoya Univ.

E-mail: †mura@hr.hm.rd.sanyo.co.jp

Abstract This paper proposes a new framework of spoken dialogue system that is based on the examples of dialogue between human and Wizard-of-OZ (WOZ) system. Along with modeling of information retrieval dialogue, a system for shop information retrieval in a driving car has been designed. The system refers to the dialogue examples to find an example that is suitable to generate a query or a reply. And the authors have collected a large-scale dialogue data using a WOZ system. Using the WOZ system for collecting dialogue examples, it becomes efficient to collect dialogue examples.

Key words Spoken dialogue, Speech recognition, Wizard of OZ, Speech understanding, Reply generation, Car environment, Dialogue corpus

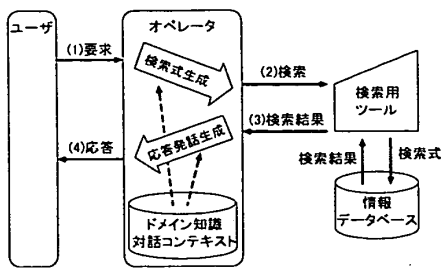


図1 情報検索対話における対話情報の流れ

Fig.1 Information flow of information retrieval dialogue

1. はじめに

近年、計算機能力の向上などを背景に、大量の音声やテキストのコーパスを利用した音声対話システム構築手法が盛んに研究されている([1]~[3])。筆者らはこれまで、情報検索対話において柔軟な対話制御を行うことや、音声認識誤りに対してロバスタな意図理解を行うことなどを目指し、対話事例を利用する音声対話制御手法を提案してきた[4],[5]。この手法では発話データとその発話に伴う検索式などの付加情報の対を事例として用いることにより、システムの行動を決定する。このシステムを有効に動作させるためには、大量の事例データを収集することが必要であるが、これまでは、人間対人間の対話データを収集し、発話に対して人手で付加情報の付与を行っており、大きな労力を伴うという問題点があった。

そこで本論文では、WOZ (Wizard of OZ) 方式を利用した擬似音声対話システムと人間との間で行われた対話事例を事例データとして利用する新しい事例ベース対話システム構築手法を提案するとともに、その手法に基づいて構築した情報検索向け音声対話システムの概要について述べる。

2. 事例に基づく対話処理

まずはじめに、我々がこれまでに提案してきた事例に基づく対話処理の概要について述べる[4][5]。

2.1 情報検索対話のモデル化

人間のオペレータが情報データベースを検索し、ユーザに対して情報を提供する状況における、オペレータとユーザ間の対話は、図1のようにモデル化することができる。

(1) 要求 ユーザの要求発話を受けたオペレータは、ドメイン知識や現在の対話コンテキストを参照しながら検索式を生成する。検索式の生成は、一般にはコンピュータなどの検索ツールを操作することで間接的に行われる。

(2) 検索 生成された検索式により、検索が実行される。

(3) 検索結果 検索結果が生成される。

(4) 応答 オペレータは、検索結果と対話コンテキストに基づいて応答を行う。

このように情報の流れを整理すると、次のように考えることができる。図1で、オペレータは対話の進行のために次の2つの判断を行っている。

判断1 ユーザの発話を受け、検索式を生成する

判断2 検索結果を受け、応答発話を生成する

熟練したオペレータは、ユーザ発話の表層情報だけでなく、ドメイン知識や対話コンテキスト、検索結果などの情報を使用して、その状況におけるユーザの目的に最も合致するような検索式あるいは応答発話を生成している。つまりこの場合の対話処理とは、ユーザの発話やドメイン知識などの入力情報から検索式などの出力情報へのマッピング操作であるとみなすことができる。そこで我々は、このような対話を進行するための「判断」を、熟練した人間のオペレータが行った行動事例を参照して行うことができるのではないかと考えた。つまり、大量の対話事例すなわち前述のようなマッピング情報を蓄積しておき、ある入力情報に対する出力情報を、類似した対話事例におけるマッピング情報を基に決定するのである。

2.2 事例に基づく検索式および応答発話の生成

2.2.1 事例データの構造

ユーザとオペレータが情報検索対話を行う際に行われる2つの「判断」は、以下のような入出力情報間のマッピングとしてあらわすことができる。

- 検索式生成のための「判断」における入出力情報
 - 入力 ユーザの発話内容、対話コンテキスト
 - 出力 検索式
- 応答発話生成のための「判断」における入出力情報
 - 入力 ユーザの発話内容、対話コンテキスト、検索結果
 - 出力 応答発話

従って事例データとしては、このような入出力情報を網羅するものを保持すればよい。具体的には、以下の情報を要素とする事例データを多数収集し、事例データベースを構築する。

- (1) ユーザの発話テキスト
- (2) 検索式
- (3) 応答発話テキスト
- (4) 検索結果
- (5) 対話コンテキスト (過去の発話内容、グラウンディング情報、談話オブジェクト[6]など)

入力発話に対し、事例データベースを参照して検索式および応答発話を生成する手順は以下の通りである。

2.2.2 検索式生成の過程

入力発話およびその時点の対話コンテキストについて、事例データベースの中から最も類似した事例を抽出する。この最類似事例中の検索式を入力発話に合うように修正し、検索式を生成する。

2.2.3 応答発話生成の過程

生成した検索式により検索を行い検索結果を受ける。その後、入力発話とその時点の対話コンテキストおよび検索結果について、事例データベースの中から最も類似した事例を抽出する。この最類似事例中の応答発話を現在の状況に合うように修正することにより、応答発話を生成する。

2.3 問題点

対話事例に基づく対話システムを動作させるためには、前述のような事例データベースを構築することが必要である。大規模な事例データベースを構築するためには、発話に対して検索式や検索結果などの付加情報が付与された大量の対話テキスト

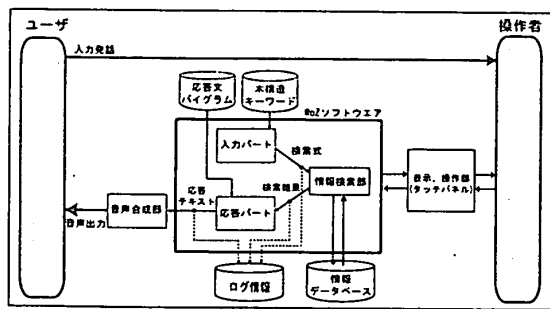


図 2 WOZ システムの構成
Fig. 2 Configuration of Wizard of Oz System

が必要である。

これまで我々は、人間対人間の対話を収集し、発話内容をテキスト化した後、各発話に対して検索式や検索結果などを付与することで事例データベースを作成していた。しかし、これには大きな労力が必要であった。また、人間対人間の対話データに基づいて構築した事例データは、人間対話システムの対話データとは異なる特徴を持つ可能性もあり、最適な事例データ構築法とはいえなかった。

3. WOZ システムを用いた事例データベースの構築

発話に対して付加情報を付与した対話データを効率的に収集する手段として、我々は WOZ (Wizard of Oz) 方式を使用することを提案する。WOZ 方式を用いて対話を行い、事例データベース構築に必要な情報を発話データと同時に収集する。

3.1 WOZ システム

WOZ 方式による対話とは、対話システムの構成要素の一部を人間が代行するが、使用者にはあたかも機械が全てを行っているかのように見せかけたものである。WOZ 方式を使用して対話データを収集することにより、より人間対機械の対話に近いデータの収集が期待できる。

これまでも WOZ 方式を用いた音声対話データの収集は多く行われている ([7]~[10] など) が、これらはいずれも人間と人工物の間の対話データの収集、分析、評価などが目的であり、また人間が代行する機能は音声認識の機能のみであることが多かった。

我々はそれにとどまらず、ユーザ発話の意味内容を解釈する機能、検索式を生成し検索を実行する機能、応答発話を生成する機能、を人間が代行できるような専用ソフトウェアを構築し (WOZ ソフトウェアと呼ぶ)、その WOZ ソフトウェアを操作者 (ウィザード) が操作しながらユーザと対話を行い、得られたデータを直接対話システムの制御に用いる枠組みを提案する。つまり、WOZ ソフトウェアと操作者が擬似音声対話システム (WOZ システム) を構成し、人間との間で情報検索対話を行い、その際に生成された検索式や検索結果をログ情報として保存する。そしてそのログ情報と、テキスト化した発話内容を用いて事例データベースを構築し、対話制御に利用する。

3.2 システム構成

今回構築した WOZ システム全体の構成を図 2 に示す。WOZ

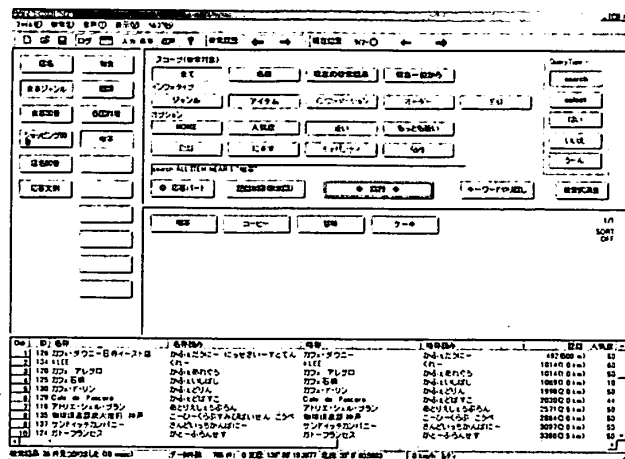


図 3 WOZ システムの画面の例 (1) 検索式生成画面
Fig. 3 An Example of Display of Wizard of Oz System (1): Query Generation Part

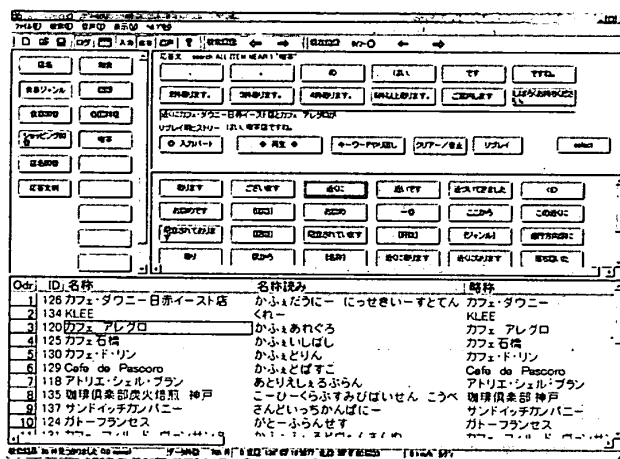


図 4 WOZ システムの画面の例 (2) 応答文生成画面
Fig. 4 An Example of Display of Wizard of Oz System (2): Reply Generation Part

ソフトウェアは PC 上のソフトウェアとして実現されており、Windows2000 上で動作する。使用プログラム言語は C++ である。WOZ ソフトウェアは検索式を生成する画面 (入力パート) と、応答文を生成する画面 (応答パート) から成る。各パートの画面例を図 3, 4 に示す。操作者が簡単に操作を行うことができるようにタッチパネルを採用し、GUI 表示された画面中のボタンをタッチすることにより、検索式の生成、情報データベースの検索、応答発話の生成と合成音声の出力を行うことができる。

WOZ システムと人間の間で自然な対話を行うためには、WOZ ソフトウェアの高い操作性が求められる。我々は、事前に収集した人間対人間の対話コーパスの情報を用いて WOZ ソフトウェアの設計を行い、ユーザとの対話中に実時間で操作を行うことができるように以下のような種々の工夫を施している。

入力パートでは、内容別に木構造に整理されたキーワードの中から適切なキーワードをタッチパネル操作によって選択することで検索式を生成し、素早く店情報の検索が実行できるよ

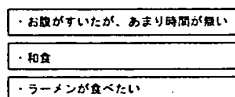


図 5 状況プレートの場合

Fig. 5 An Example of Prompting Panel

にした^(注1)。検索結果は画面下部にリスト形式で表示される。

応答パートには、応答発話生成用のテキスト入力ボタンと店情報の検索結果リストが表示される。テキスト入力ボタンは、応答発話生成に用いられる単語や文節、定型的な短文などに対応しており、そのボタンを順次押すことにより応答発話テキストが生成される。テキスト入力ボタンの配置は、人間対人間の対話コーパスから事前に求めた、各テキスト要素間の接続頻度(応答文バイグラム)を基に決定した。すなわち、直前に入力されたテキストに後続する頻度の高いものから順にテキスト入力ボタンが表示されるようにすることにより、応答発話テキストを素早く生成することができる。さらに、固有名詞である店情報を素早く入力できるようにするため、検索結果リストの店名の部分を直接タッチすることにより応答発話テキストに店名を含めることができるようにした。生成された応答発話テキストは、音声合成部で音声に変換され出力される。

2つのパート間は切り替えボタンにより自由に移動することができる。また、あいづちなど、被験者発話中に即座に提示すべき音声については入力パートにも発話生成ボタンを設け、自然な対話を行うことができるよう配慮している。

3.3 WOZ システムによる対話データの収集

音声対話による情報検索アプリケーションとして、自動車内での店情報検索をターゲットとし、WOZ システムと人間との間の対話データを収集した[11]。本データは被験者を情報検索を行うユーザとし、被験者が運転する自動車内で収集された。WOZ ソフトウェアが動作する PC を自動車内に配置し、ウィザードは後部座席で操作を行う。対話の内容は別の PC を用いて全て録音している。

データ収集は以下のような手順で、一人あたり約5分間行った。

- 被験者に対し、図5のような状況プレートを提示する
 - 被験者は状況プレートの内容を基に、自由に対話を行う
- ウィザードは、被験者の発話を聞きながら WOZ システムを操作^(注2)し、適切な検索および音声合成による応答を行う。

収集したデータの規模を表1に示す。

3.4 WOZ のログ情報を利用した事例データベースの構築

WOZ ソフトウェアは、図6に示すような詳しいログ情報を出力するように設計されている。出力される情報としては主に以下のようなものがある。全てのログ情報は時間情報とともに記録されている。

- 話者 ID (対話開始時にウィザードが入力する)

表 1 収集した WOZ 対話データ

Table 1 Collected WOZ Data

		男性	女性	計
ユーザ	人数	326	161	487
	総発話時間 (min.)	321	178	499
	総発話文数	8,849	4,979	13,828
WOZ 音声合成	総発話時間 (min.)	527	264	791
	総発話文数	8,317	4,170	12,487

2001/10/09 14:00:11.444 [button] push 食事ジャンル
2001/10/09 14:00:11.955 [button] push 喫茶
2001/10/09 14:00:13.057 [button] push はい、喫茶店ですね。
2001/10/09 14:00:13.077 [voice] start "はい、喫茶店ですね。"
2001/10/09 14:00:13.517 [button] push 入力パート
2001/10/09 14:00:14.018 [button] push 喫茶
2001/10/09 14:00:14.018 [doc] Keyword add 喫茶
2001/10/09 14:00:14.489 [button] push 実行
2001/10/09 14:00:14.489 [Search] Starting search ALL ITEM NEAR 1 "喫茶"
2001/10/09 14:00:14.489 [search] Result 26,126:0,134:0,120:0,125:0,118:0,124:0,.....
2001/10/09 14:00:14.539 [Frame] created 応答パート
2001/10/09 14:00:16.321 [button] push 近くに
2001/10/09 14:00:16.321 [doc] BigramView set 近くに
2001/10/09 14:00:16.341 [voice] start "近くに"
2001/10/09 14:00:17.002 [ResultView] Click field 名称
2001/10/09 14:00:17.523 [button] push カフェ・ダウニー日赤イースト店
2001/10/09 14:00:17.593 [voice] start "カフェ・ダウニー日赤イースト店"
2001/10/09 14:00:17.914 [button] push と
2001/10/09 14:00:17.914 [doc] BigramView set と
2001/10/09 14:00:18.975 [ResultView] Click field 名称
2001/10/09 14:00:19.536 [button] push カフェ アレグロ
2001/10/09 14:00:19.856 [button] push が
2001/10/09 14:00:19.856 [doc] BigramView set が
2001/10/09 14:00:20.207 [button] push ございます
2001/10/09 14:00:20.207 [doc] BigramView set ございます

図 6 WOZ システムのログ情報

Fig. 6 Log of WOZ System

表 2 事例データベース作成に使用したデータ

Table 2 Configuration of Constructed Example Database

		男性	女性	計
ユーザ	人数	163	80	243
	総発話時間 (min.)	164	90	254
	総発話文数	4,539	2,489	7,028
WOZ 音声合成	総発話時間 (min.)	264	131	395
	総発話文数	4,171	2,104	6,275
有効対話数 (*)		817	389	1,206

(*) 有効対話: 検索式と検索結果を対応付けることができたユーザ発話と、それに対する WOZ 音声合成発話の組

- 入力発話に対して生成された検索式
- 生成した検索式による検索結果 (件数と店 ID)
- オペレータが生成した応答発話のテキスト

保存された WOZ ログ情報を利用することにより、以下の手順により効率よく事例データベースを構築することができる。まず、録音された対話音声をもとにユーザの発話内容の書き起こしを行い、時間情報を付けて保存する。その後、WOZ ソフトウェアのログ情報の時間情報を元に、ユーザ発話と検索式、および検索結果と応答発話の対応付けを行う。

収集した対話データの一部を用いて事例データベースを構築した。事例データベース構築に使用したデータを表2に示す。

タスク外発話や、検索式の仕様外の発話を除き、総発話数の約2割に対して検索式や検索結果が対応付けられた。

4. 対話事例を利用した音声対話システム

ここでは、構築した事例データベースを利用して動作する対話システムについて述べる (詳細は[5])。タスクは自動車内

(注1): 文脈情報を扱うような発話に対する検索式は、WOZ ソフトウェアの操作が複雑になるのを避けるため、現在は定義していない。

(注2): 定義されている検索式が適用可能な発話に対しては検索式をできるだけ生成して検索を行い、応答発話を生成した。検索式が生成できない場合には、無理に検索式を生成せず、適当な応答発話のみを生成した。

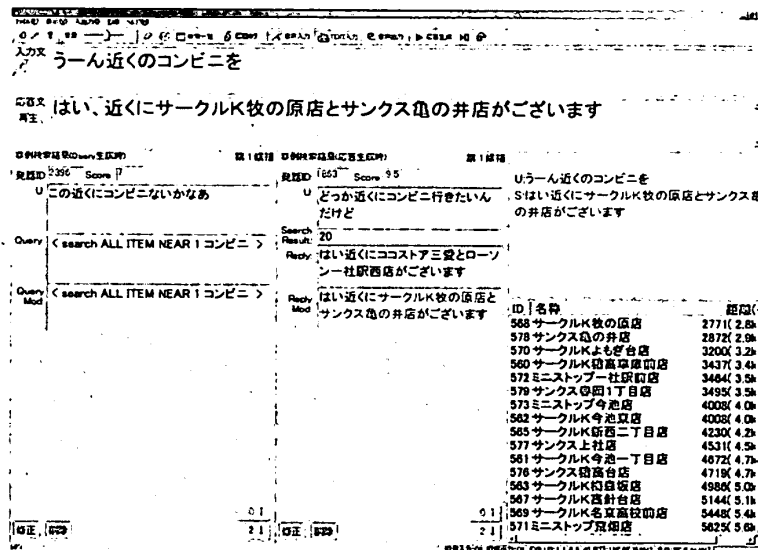


図7 事例ベース対話システムの画面

Fig. 7 A View of Example-based Dialogue System

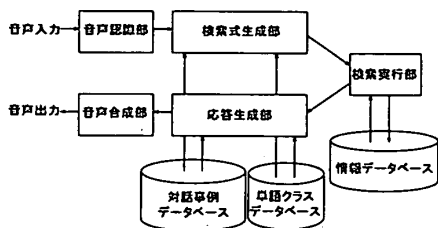


図8 事例ベース対話システムの構成

Fig. 8 Configuration of Example-based Dialogue System

# ID [0001]	U:何かスナックが食べたいな	# ユーザ発話
	< search ALL ITEM KEY=スナック >	# 検索式
S:はいスナックのお店ですね。		# システム発話
イルソーレと英園店があります		# 検索結果
< ? ID=2010 102 >		
# ID [0002]	U:うどんが食べたいのでうどん屋さんの場所を	# ユーザ発話
教えてください		# 検索式
< search ALL ITEM NEAR 1 うどん >		# システム発話
S:かしこまりました。近くに豊松と四防園酒造店		# 検索結果
がございます		
< ? ID=50 54 52 49 45 >		

図9 事例データベースの例

Fig. 9 Dialogue Example Database(a part)

での店舗情報検索である。本システムは、Windows2000上で、C++言語を用いて実装された。システム画面の一例を図7に示す。

4.1 システム構成

システムの構成について図8を参照しながら述べる。

対話事例データベース WOZソフトウェアのログ情報と対話テキストを基に作られる。発話テキストは形態素解析^(注3)されており、対話の遂行に重要な語(店名、施設名、食品名など)はあらかじめ意味別にクラス化され、単語クラスタグが付与されている。事例データベースの例を図9に示す。

単語クラスデータベース 本タスクにおける重要語を抽出し、意味別にクラス化したデータベース。単語クラスは、対話コーパス中の対話を元に経験的に求めた。

(注3)：形態素解析は茶釜[12]を使用している

情報データベース 名古屋市内のレストラン、店舗など約800件が収録された店舗情報データベース。WOZシステム内で使用されているものと同一である。

音声認識部 日本語ディクテーション基本ソフトウェア[13]を使用。言語モデルは事前に収集した人間対人間の対話コーパスから作成した。

検索式生成部 入力発話に対し、その時点の対話の状況に最も近い事例を対話事例データベースから抽出し、その事例で用いられた検索式を現在の状況に合うように修正して出力する。

検索実行部 検索式を用いて情報データベースにアクセスし、検索結果を得る。

応答生成部 入力発話と検索結果に対し、最も近い事例を対話事例データベースから抽出し、その事例で行われた応答発話を現在の状況に合うように修正して出力する

音声合成部 応答発話テキストを音声として出力する。三洋電機製音声合成エンジン、EleganTalk Ver.2.1[14]を使用している。

4.2 動作

システムの動作について述べる。(具体的な動作例は、図10を参照。)

【ステップ1：類似事例抽出】 音声認識結果テキストに対し、対話事例データベース中から最類似事例を抽出する。走行雑音や運転中発話に対する音声認識・理解のロバスト性を考慮し、2発話間の類似スコアは単語クラスを考慮したキーワードマッチングにより求めている。キーワード抽出は、形態素解析結果から自立語を抽出して行った。

【ステップ2：検索式修正】 抽出した最類似事例に対応する検索式を、入力発話に合うように修正する。具体的には、単語クラスデータベース中の類似語情報を用い、類似した語の入れ替えを行っている。

【ステップ3：検索】 得られた検索式で検索を実行。検索結果を得る。

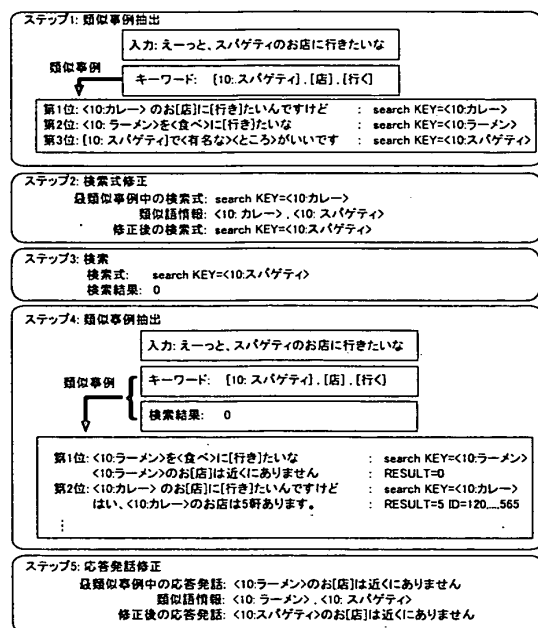


図 10 検索式生成および応答文生成の例

Fig. 10 Example of Query and Reply Generation

【ステップ 4: 応答発話生成】入力された音声認識結果テキストと検索結果件数に対し、対話事例データベース中から最類似事例を抽出する。発話テキストについての発話間類似スコアと、ステップ 3 で得られた検索結果件数と事例中の検索結果件数の差から得られる検索結果間類似スコアの 2 つの値を重み付き加算して、総合的な類似スコアを求めている。検索結果間類似スコアは次のようにして求める。

検索結果数が 0 のとき

事例データベース中の検索結果数が 0 である事例に対して最高のスコアを与え、0 以外の事例に対しては最低のスコアを与える。

検索結果数が 1 以上のとき

事例データベース中の事例における検索結果数と一致するとき最高スコアを与え、差が大きくなるほど低いスコアを与えるようなヒューリスティクスを用いている。

つまり例えば、検索結果が 1 件も得られなかった場合は、事例データベース中の 1 件も検索結果の得られていない事例にのみマッチするようになる。

【ステップ 5: 応答発話修正】求められた最類似事例中の応答発話を修正し、音声合成部へ送る。修正は、単語クラスデータベース中の類似語情報に基づいて行われる。

4.3 事例データの追加, 修正, 削除

このシステムは、事例データの追加, 修正, 削除を行うことも可能である。対話を行い、動作に失敗した場合、画面下部の修正や削除などのボタンを操作することにより、現在の事例データの修正や新しい事例の追加, 不要な事例の削除などを行うことができる。

5. おわりに

本稿では、対話事例を利用する音声対話制御手法における効

率的事例データ収集方法として、WOZ システムを用いる手法を提案した。この手法による効果としては、以下のようなものが挙げられる。

- WOZ システムのログに埋め込まれた知識（発話→検索式変換, 応答発話生成など）を対話システムの知識として利用できる

- WOZ システムで対話の収集を行うため、実際の対話システムの使用環境により近い、つまり人間対機械の対話に近い状況下の対話事例が収集できる

- 事例データベース構築の際に必要な発話の書き起こしの手間が削減できる

今後の課題としては、WOZ システムを用いて構築した事例データベースによる対話処理性能の評価, 文脈処理の実現, などが挙げられる。

文 献

- [1] E. Levin, R. Pieraccini, and W. Eckert. Using markov decision processes for learning dialogue strategies. Proc. ICASSP98 Vol. 1, pp. 201-204, 1998.
- [2] S. Young. Talking to machines (statistically speaking). Proc. of ICSLP-2002 pp. 9-16, 2002.
- [3] 松原茂樹, 河口信夫, 外山勝彦, 武田一哉. 音声対話コーパスの収集と利用. 人工知能学会誌, Vol. 17, No. 3, pp. 279-284, May 2002.
- [4] 村尾浩也, 河口信夫, 松原茂樹, 稲垣康善. 対話事例を利用した音声対話システム. 情報処理学会研究報告, 2000-SLP-34-34, 2000.
- [5] H. Murao, N. Kawaguchi, S. Matsubara, and Y. Inagaki. Example-based query generation for spontaneous speech. Proc. of 2001 IEEE Workshop on Automatic Speech Recognition and Understanding 2001.
- [6] 中野幹生, 堂坂浩二. 音声対話システムの言語・対話処理. 人工知能学会誌, Vol. 17, No. 3, pp. 271-278, May 2002.
- [7] S. Lemmela and P. Poda. Efficient combination of type-in and wizard-of-oz tests in speech interface development process. Proc. of ICSLP-2002 pp. 1477-1480, 2002.
- [8] N. B. Yoma, A. Cortes, M. Hormazabal, and E. Lopez. Wizard of oz evaluation of a dialogue with communicator system in chile. Proc. of ICSLP-2002 pp. 2701-2704, 2002.
- [9] 伊藤克互, 秋葉友良, 速水悟, 田中和世. Woz システムで収録した自由発話音声の対話データベース. 日本音響学会秋季研究発表会講演論文集, pp. 37-38, 1998.
- [10] 岡登洋平, 加藤佳司, 山本幹雄, 板橋秀一. 相槌を打つ音声対話システムの評価. 人工知能学会研究会資料, SIG-SLUD-9803-2, 1998.
- [11] N. Kawaguchi, S. Matsubara, K. Takeda, and F. Itakura. Multi-dimensional data acquisition for integrated acoustic information research. Proc. of 3rd International Language Resources and Evaluation Conference (LREC-2002), pp. 2043-2046, 2002.
- [12] 松本裕治, 北内啓, 山下達雄, 平野善隆. 日本語形態素解析システム『茶釜』version 2.0 使用説明書. Information Science Technical Report NAIST-IS-TR99008 1999.
- [13] T. Kawahara, A. Lee, T. Kobayashi, K. Takeda, N. Mine-matsu, S. Sagayama, K. Itou, A. Ito, M. Yamamoto, A. Yamada, T. Utsuro, and K. Shikano. Free software toolkit for japanese large vocabulary continuous speech recognition. Proc. of ICSLP-2000 Vol. 4, pp. 476-479, 2000.
- [14] 余田直之, 平井裕之, 橋本誠, 大西宏樹. 自然音声合成ソフトウェア. Sanyo Technical Review Vol. 33, No. 3, pp. 55-62, Dec. 2001.