

# ロボット・エージェント対話行動制御部構築ツール RIME-TK を用いた 質問応答機能の実現

成松 宏美<sup>†</sup> 中野 幹生<sup>††</sup> 船越孝太郎<sup>††</sup> 長谷川雄二<sup>††</sup> 辻野 広司<sup>††</sup>

<sup>†</sup> 津田塾大学 学芸学部 情報数理科学科  
〒 187-8577 東京都小平市津田町 2-1-1

<sup>††</sup> (株) ホンダ・リサーチ・インスティテュート・ジャパン  
〒 351-0188 埼玉県和光市本町 8-1

**あらまし** 本稿では、ロボットやエージェントの対話行動制御部を構築するためのツールキット RIME-TK を用いて、質問応答機能を実現する方法について述べる。RIME-TK は、さまざまなタイプのタスクを行う対話行動制御部を構築するツールであり、特定の種類のタスクを遂行することに特化したエキスパートと呼ばれるモジュールを構築することで、新しい機能を追加することができる。本研究では、簡易な記述形式で書かれた質問応答データベースを読み込み、それに基づいて音声理解・応答を行うエキスパートの設計と実装を行った。これにより、専門知識がなくても質問応答機能を開発でき、他のタスクとともに実行させることが可能になった。また、構築したシステムを用いて収集した対話データの分析から、初期システムとしては十分な性能がでることが判明した。

**キーワード** 対話ロボット, 対話エージェント, 質問応答, 記述言語, 音声理解

## Building a Question-Answer System based on RIME-TK, a Toolkit for Dialogue and Behavior Controller of Robots and Agents

Hiroimi NARIMATSU<sup>†</sup>, Mikio NAKANO<sup>††</sup>, Kotaro FUNAKOSHI<sup>††</sup>, Yuji HASEGAWA<sup>††</sup>, and  
Hiroshi TSUJINO<sup>††</sup>

<sup>†</sup> Department of Mathematics and Computer Science, Tsuda College  
2-1-1 Tsuda-machi, Kodaira-shi, Tokyo 187-8577, Japan

<sup>††</sup> Honda Research Institute Japan Co., Ltd.  
8-1 Honcho, Wako, Saitama 351-0188, Japan

**Abstract** This manuscript reports our effort for facilitating building question-answering functions using RIME-TK, which is a toolkit for dialogue and behavior controller for robots/agents. RIME-TK employs experts, each of which is responsible for executing a task in a certain domain, and various kinds of tasks can be dealt with by incorporating new kinds of experts. We have created an expert class for question-answering based on a database written in a simple description format. This makes it possible for non-expert to create question-answering contents, resulting in a wide variety of systems. The result of a preliminary evaluation showed that this method enables creating question-answering systems with enough utterance understanding accuracy.

**Key words** conversational robot, spoken dialogue agent, question answering, description language, speech understanding

### 1. はじめに

近年のロボットやアニメーションエージェントの研究の発展は目覚ましいものがある。これらのロボットやエージェントが、音声で人間と対話しながら情報を授受する機能を持つことが

待されている。そのような音声対話技術は、カーナビゲーションシステムなどのように、限定されたタスクであれば、十分実用に耐える技術になってきている。

しかしながら、まだ、一般のユーザが音声対話機能を日常的に使う段階には至っていない。そのためには、理解率の向上や

誤りの発見と回復といった、技術的な課題の解決することもさることながら、ユーザが興味を持って対話ができるように、魅力的なコンテンツを作成することも重要である。

本研究では、さまざまなタイプの対話の中で、近年多く構築されている質問応答システムに注目する。質問応答システムは、ユーザの質問に対し、質問応答データベースを検索して応答する単純なシステムである。Frameに基づく対話管理を行うシステムや、VoiceXMLで記述されるような対話システムに比べると単純ではあるが、逆にシステムの動きが予測しやすく、音声言語処理の非専門家でも容易にデザインすることができる。西村ら [8], [9] や Cincarek ら [1] は、「たけまるくん」や「キタちゃん」などの質問応答システムを開発し、実環境で運用してデータ収集に用いている。収集されたデータは、応答選択の向上手法の研究に用いられている [12], [13]。また、これらの質問応答システムを有限状態文法ベースの音声認識を用いた音声対話システムとの統合も検討されている [4]。

一方、我々は、タスク指向音声対話やロボットの物理動作によるタスク遂行を出発点として、マルチドメイン対話行動制御のモデル RMIE およびツールキット RIME-TK を構築し [7]、さらに、雑談やプレゼンテーションなどさまざまな対話を対象にしてきた。音声言語処理の専門家でなくてもプレゼンテーションなどのコンテンツを作成できるように、簡易な記述言語を開発し、RIME-TK で用いることができるようにしてきた [2], [10]。これにより、デザインの専門家などが魅力的なコンテンツを作れるようになり、音声対話技術が普及することを期待している。このツールに、単純だが有用な質問応答機能を導入することで、さらにタスクの幅を広げることを試みる。

本研究の目的は、質問応答機能を非専門家でも容易に実現できるようにすること、および、その機能を他のタイプの対話と統合できるようにすることにある。これによって、様々なタイプの対話を行うロボットやエージェントの開発において、質問応答機能の部分をアウトソーシングすることが可能になる。今までに構築された質問応答システム「たけまるくん」や「キタちゃん」は大量の訓練データを用いて音声認識用言語モデル等の向上を行っているが、我々のアプローチは、なるべく容易に対話行動制御部を構築することであり、方向性が異なる。

本研究がベースとしている RIME-TK は、エキスパートと呼ぶ特定のドメインのタスクを遂行するのに特化したモジュールを複数用いる。エキスパートが実装すべきインタフェースが決まられており、その要件を満たしていれば、どのようなタイプの対話も実装することができる。我々は、簡易な質問応答機能の記述から、質問応答を行うエキスパートを構築できるようにした。これによって、さまざまな分野の人がその専門的知識を生かして質問応答システムが作成できるようになる。自動構築したシステムは、従来のシステムのように大量のデータで言語モデルなどを訓練しているわけではないので、性能はそれほど高いわけではないが、訓練用のデータ収集には十分な性能を持つ。

本稿では、質問応答機能の記述言語および、それを用いたエキスパートの作成の詳細を述べる。さらに、構築した質問応答

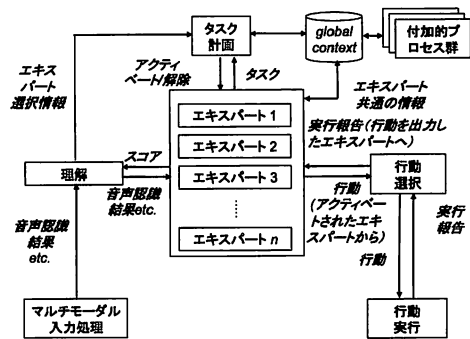


図1 RIMEのモジュール構成

システムの性能に関して述べる。

## 2. RIME-TKに基づく質問応答機能の構築

### 2.1 RIME: 対話行動制御のマルチエキスパートモデル

RIMEは、ロボット/エージェントの知能システムの中で、記号レベルでの状況理解と行動選択を司るモジュール（対話行動制御部と呼ぶ）のモデルである。対話行動制御部は、音声・画像認識や他のセンサ解釈モジュールによってシンボル化された入力を受け取り、それを基に人の意図や状況を推測し、適切な行動を選択し、ロボットハードウェア制御や音声合成などの行動実行モジュールに送る。

RIMEでは、特定の種類のサブタスクに特化した知識と内部状態を持つエキスパートと呼ぶモジュールを用いる（これは、マルチドメイン音声対話システムで用いられているドメインエキスパート [11] の概念を拡張したものである [7]）。たとえば、天気予報に関する質問に答えられるロボットであれば、「天気予報に関する質問を理解する」というサブタスクのためのエキスパートや「天気予報を人に伝える」というサブタスクのためのエキスパートを持つ。また、「特定の場所に移動する」という物理行動を行うサブタスクのためのエキスパートや、「人の発話に単純に反応する」ためのエキスパートなども用いることができる。これらのエキスパートを順次利用することにより、複雑なタスクを遂行することができる。たとえば、ある物を説明するタスクは、その物のところに人を案内して、言葉で説明するという二つのサブタスクを順次遂行することによって行うことができる。

RIMEでは、このようなエキスパートを利用して、全体のシステムを動作させるためのプロセス群（調整プロセス群と呼ぶ）が走っている。RIMEのモジュール構成を図1に示す。調整プロセスは3つあり、並行動作する。理解プロセスは音声認識結果をエキスパートに送信し、最適なエキスパートを選択し、タスク計画プロセスにその情報を送る。行動選択モジュールは選択されたエキスパートに対し、次の動作の決定を要求する。タスク計画プロセスは、タスクを遂行したり、音声認識結果に反応したりするために、どのエキスパートをアクティベートし、どのエキスパートをディアクティベートするかを決定する。これら3つのプロセスは発話割り込みを扱うために並列で動作

する。

それぞれのエキスパートは内部状態にアクセスするためのメソッドを持っていなければならない。initialize メソッドはエキスパートが作られたときに呼ばれ、内部状態を初期化する。understand メソッドは音声認識結果を受け取った際に理解プロセスから呼び出され、音声認識結果に基づいて情報を更新する。select-action メソッドは、行動選択プロセスから継続的に呼び出され、発話待ちの状態であれば、内部状態に基づき、MADL で表現されたマルチモーダルアクションを 1 つ出力する。detect-interruption メソッドは、直前に understand で理解した発話が、現在実行中の行動に対する割り込みであるかどうかを判断した結果をブール値で返す。handle-interruption メソッドは、割り込みが発見された場合にシステムが実行すべき行動を返す。たとえば、実行を中止する命令などが返されることがある。

understand メソッドの返り値は、その音声認識結果がどのくらいそのエキスパートで処理されるべきかを表す 0 以上 1 以下のスコアである。理解プロセスは、音声認識結果を、現在アクティブされているエキスパートおよび、新規にアクティブされる可能性のあるエキスパートに、この understand メソッドを用いて送り、最も高いスコアを返したエキスパートを選択して、その情報をタスク計画部に送る。これは、マルチドメイン音声対話システムにおけるドメイン選択の機能にあたる。

これらのインタフェースを実装しさえすれば、内部で知識や状態をどのような形で保持しているか、また、どのようなアルゴリズムで理解や行動選択を行うかに関わらず、どのようなエキスパートでも導入することができる。

各エキスパートは、グローバルコンテキストと呼ばれるデータ格納部を介して、共通に使える情報（例えば、話題になった事物、ユーザの興味、周囲の状況など）を授受できる。

## 2.2 RIME-TK

前節で説明したモデルに基づき、対話行動制御部を構築するためのツールキット RIME-TK(RIME-ToolKit) を作成した。このツールキットは Java で実装されている。RIME-TK には、理解、行動選択の各プロセスおよびその他のモジュールを作成するのに必要な抽象クラスがあらかじめ実装してある [6]。

RIME-TK を用いて対話行動制御部を構築するには、次の作業を行う。まず、ロボット・エージェントが達成するタスクに必要なエキスパートを用意し、インタフェースを実装する。さらに、タスク計画プロセス、タスク計画プロセス、マルチモーダル入力処理部、行動実行部の具体クラスを作成する。マルチモーダル入力処理部は音声認識結果などを RIME-TK 内で用いられるデータ構造に変換し、理解プロセスに送る。行動選択部から行動実行部に送られる行動命令は、MADL(Multimodal Action Description Language) と呼ぶ XML 形式で記述される。MADL は、音声合成用のテキストやロボットのジェスチャーや移動のコマンドなどを含む。発話や動作の停止命令を含むこともできる。また、動作の開始を遅らせるような命令も含まれる。ひとつの MADL に含まれる命令は同時に実行が開始され

要素	効果
utterance	音声合成を用いて発話する
emotion	音声合成の感情パラメータを変化させてid属性で指定された感情を含んだ音声を出力する
cancel-utterance	音声合成を中断する
play	name属性で指定されたジェスチャーを行う
go-to	属性でしてされた位置に移動する
stop	移動を中断する
show-slide	スクリーンにスライドまたはweb画面を表示する。uri属性でスライドやweb上のアドレスを指定する。
show-text	スライド上にテキストを表示する
suspend	コマンドの実行をdurationで指定した時間遅らせる

図 2 現在扱える MADL の要素

る。以下に発話とジェスチャーを同時に行わせる際の例を示す。

```
<madl>
<utterance>こんにちは</utterance>
<play name="wave-both-hands">
</madl>
```

上記の具体クラスを一から作成するのは容易ではない。そのため、RIME-TK は次のようなライブラリを備えている。

まず、エキスパートの構築を容易にするために、頻繁に用いられる機能を実装したテンプレートを用意している。これらはエキスパートのサブクラスとして用意されている。

さらに、マルチモーダル入力処理部のサンプルとして、音声認識システム Julius [3] の入力を受け付けるものが用意されている。また、デバッグやシミュレーションのために、ユーザ発話をキーボード入力やファイルから入力できるようにするサンプルも用意されている。Julius を使う場合、ひとつの音声入力ストリームに対して複数の音声認識器を用いることが可能である。各エキスパートがどの音声認識器の出力を用いるかを指定する必要がある [2]。

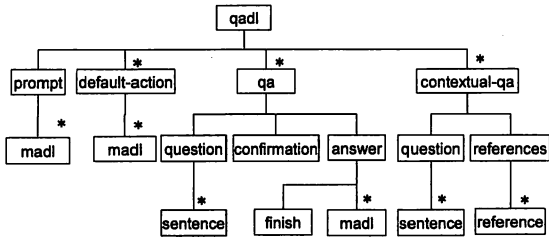
また、行動実行部のサンプルとして、ロボットの発話・行動を制御するモジュールが用意されている。このモジュールは、NTT-IT 製の FineVoice を用いた音声合成サーバ、Honda ヒューマノイドロボット ASIMO や、全方向移動台車ロボット、ロボットのシミュレータなどへの行動命令を授受するサーバ [14]、ロボットやエージェントがディスプレイを用いてプレゼンテーションするためのスクリーン制御モジュールとの通信機能を持つ。扱える MADL の要素の一部を表 2 に示す。

## 2.3 質問応答機能記述言語

質問応答機能を簡単に記述できるように、XML に基づく言語 QADL を策定した。図 3 に QADL の構造木を、図 4 に QADL の例を示す。

prompt 要素は質問応答が始まったときに実行する MADL のリストを持つ。default-action 要素は、ユーザの質問がどの質問応答カテゴリにもマッチしなかった場合に何を実行するかを示す。

qa 要素は質問応答カテゴリを表す。question 要素は質問文例を表す sentence 要素を持ち、ユーザの質問文がこの sentence 要素のどれかにマッチしたときには answer 要素の中の madl 要素を順に実行する。この際、answer 要素の属性が文脈情報



\*は複数の要素を子要素として持ち得ることを示す。

図 3 QADL の構造木

として保存される。sentence 要素の中の文では、{} でキーフレーズを指定する。質問文のマッチングの仕方は実装に依存するが、現在の実装ではキーフレーズを用いてマッチングを行う。answer 要素の中に finish 要素がある場合、この質問応答を終える。confirmation 属性は、ユーザの発話のカテゴリの信頼度が低いときの確認発話である。

contextual-qa 要素は、文脈に依存して理解すべき発話とその応答へのポイントを持つ。reference 要素の属性は文脈情報であり、これが保持している文脈情報とマッチしたとき、この reference 要素の値を id 要素を持つ qa 要素の応答が実行される。

## 2.4 質問応答エキスパートの自動構築

### 2.4.1 質問応答エキスパートクラス

QADL で記述された質問応答を実行するため、RIME-TK に QAExpert クラスを用意した。QAExpert クラスのインスタンスは、QADL を一つ読み込んで、質問応答データベースを構築する。qa 要素、contextual-qa 要素の質問文例のリストを保持し、各質問文例から qa 要素、contextual-qa 要素が同定できるようにする。

### 2.4.2 発話の理解

QAExpert クラスの understand メソッドは、ユーザ発話の音声認識結果に最も近い質問文例を選ぶことによって発話理解を行う。質問文例の選択はキーフレーズマッチングによって行う。従来システムではキーフレーズマッチング [8] を行うものと、単語セットのマッチング [1], [9], [12] を行うものがあるが、予備実験の結果から、統計言語モデルの訓練データが大量にない場合には、キーフレーズマッチングの方が良いことが判明したため、キーフレーズマッチングを用いる。

キーフレーズマッチングは次のように行う。各質問文例について、その中のキーフレーズが、最尤の音声認識結果の中にすべて現れるかどうかを調べる。すべて現れた質問文例のうち、最もキーフレーズの数が多いものを選ぶ。それでも一つに絞れない場合は、キーフレーズが何個の単語（ここで言う単語は言語モデルの辞書に登録されている単語である）からなるかを調べ、単語数の合計が多いものを選ぶ。

例えば、「{ 万里の長城 } って { 何 } ですか」と「それは { 何 } ですか」の二つの質問文例があつとすると、「万里の長城は何」という音声認識結果は両方とマッチするが、キーフレーズの数が多い前者が選ばれる。

```
<?xml version="1.0" encoding="utf-8"?>
<qadl>
  <prompt>
    <madl>
      <play name="Great"/>
      <emotion id="happy_for"/>
      <utterance>こんにちは。世界遺産についての質問にお答えします。何でも聞いてみてください。</utterance>
    </madl>
  </prompt>
  <default-action>
    <madl>
      <emotion id="shame"/>
      <utterance>すみません。わかりませんでした。もう一度、他の言い方で質問してみてくださいか</utterance>
    </madl>
  </default-action>
  <qa id="great-wall">
    <question>
      <sentence>[万里の長城]について教えてください</sentence>
      <sentence>[万里の長城]って何ですか</sentence>
    </question>
    <confirmation>万里の長城についてですね？</confirmation>
    <answer topic="great-wall">
      <madl>
        <show-slide uri="file://$[WORLD_HERITAGE_PPT_DIR]/index.ppt#1"/>
        <utterance>すみません。万里の長城は、中国にある有名な世界遺産です。</utterance>
      </madl>
      <madl>
        <utterance>場所や見所など知りたいことがありましたら、聞いてください。</utterance>
      </madl>
    </answer>
  </qa>
  <qa id="great-wall-location">
    <question>
      <sentence>[万里の長城] は[どこ]にありますか</sentence>
      <sentence>[万里の長城] の[場所]を教えてください</sentence>
    </question>
    <confirmation>万里の長城の位置ですね？</confirmation>
    <answer topic="great-wall">
      <madl>
        <show-slide uri="file://$[WORLD_HERITAGE_PPT_DIR]/wh.ppt#18"/>
        <utterance>万里の長城の場所を表示します。</utterance>
      </madl>
    </answer>
  </qa>
  <contextual-qa id="location">
    <question>
      <sentence>[場所]を教えてください</sentence>
      <sentence>[どこ]にある</sentence>
    </question>
    <confirmation>場所についてですね？</confirmation>
    <references>
      <reference topic="great-wall">great-wall-location</reference>
    </references>
  </contextual-qa>
</qadl>
```

図 4 QADL の例

選択された質問文例を持つ qa 要素または contextual-qa 要素が質問の理解結果として保持される。キーフレーズがマッチする質問文例がない場合は、理解結果は「DB 外（データベースにないカテゴリ）」とする。

understand メソッドの返り値はドメイン選択のためのスコアである。エキスパートがアクティブされているかどうか (0 または 1)、音声認識の信頼度 (単語信頼度の平均値)、質問応答カテゴリが見つかったかどうか (0 または 1) の重み付け和である。重みの自動決定は行っていない。これは今後の課題である。

### 2.4.3 応答生成

QAExpert の decide-action メソッド、すなわち行動選択は次のように動作する。まず、このエキスパートがアクティブとされたばかりの場合、すなわち、質問応答が開始された直後

システム	応答内容	応答の種類	文脈依存的な質問の種類	質問文例数	言語モデルの語彙数	言語モデル作成時にChasenに追加した語彙数
A	太陽系の惑星に関する科学的知識	196	18	455	203	58
B	世界遺産に関する知識	328	13	1794	591	352
C	京王線仙川駅周辺に関する情報	232	10	890	460	65

表 1 作成したアプリケーション

には, prompt 要素の内容を実行する。

一旦質問応答が開始された後は, 直前の understand メソッドにおける発話理解の結果を参照して応答を行う。qa 要素が選ばれた場合, answer 要素の中の madl を順に実行する。contextual-qa 要素が選ばれた場合, 文脈情報がマッチする reference 要素を調べ, その値を id にもつ qa 要素の応答を実行する。

図 4 の場合, 「万里の長城について教えて」という質問の次に「場所はどこですか」と聞いた場合, 文脈情報 topic が great-wall に設定されていることから, great-wall-location を id に持つ qa 要素を探してその answer 要素の madl を実行する。

answer 要素が finish 要素を持つ場合, 最後の madl を実行し終わると, タスクの終了を, タスク計画部に報告する。これにより, 音声応答が終わる。

発話理解の結果が DB 外であった場合, default-action 要素をランダムに一つ選んで実行する。

なお, 現在は信頼度に応じた確認発話の生成は実装していない。

#### 2.4.4 割り込み発話への対処

detect-interruption メソッドは, システムの行動中に発話理解が行われた場合, その発話が割り込み発話かどうかを調べ, 割り込み発話であれば, 行動選択部が handle-interruption を実行する。QAExpert では, 発話理解の結果が DB 外でなければ, 割り込み発話とした。そして handle-interruption は, その時点でのシステムの発話を中断し, 新しい応答を始めるようにした。

#### 2.4.5 音声認識用統計言語モデルの自動作成

RIME-TK に, QADL から音声認識用統計言語モデルを自動的に作成する機能を実装した。これは QADL の質問文例をリストアップし, Chasen [5] を用いて形態素解析を行い, その結果から Palmkit<sup>(注1)</sup>を用いて単語 trigram を作成するものである。形態素解析の際に, キーフレーズが未知語になったり, 多数の単語に分割されるような場合, キーフレーズの認識がうまくいかないことがあるため, 表記と発音のペアのリストを用意すれば, Chasen の辞書を自動的に拡張するようにした。これにより, ドメインに応じた辞書を用いての音声認識が可能になる。

収録場所	防音室
対話の形態	画面上のキャラクターエージェントに向かって座っての対話
マイク	指向性接話マイク
システム音声	スピーカー
敬示	<ul style="list-style-type: none"> <li>・対象としている惑星</li> <li>・答えられるトピック(大きさ 密度 明るさなど)</li> <li>・発話例3例( <ul style="list-style-type: none"> <li>・単語発話ではなく依頼や命令の口調で話すこと</li> <li>・システムの発話中にわりこめること</li> </ul> </li> </ul>
ユーザ数	11
ユーザ毎の対話時間	15分
発話数(音声区間で自動的に切り出された)	1056
区間検出が正しく行われた数	817

表 2 分析対象としたデータ

### 3. アプリケーションの構築と評価

上記の QAML を用いていくつかの質問応答システムを構築した。表 1 に各システムの諸元を示す。音声認識には Julius3.5 を用い, 音響モデルはディクテーションキット付属のもの [3] を用いた。

これらのシステムを, システム構築に携わった者以外に人に使ってもらい, データを収集した。本稿では, システム A を用いて収集したデータ(表 2 参照)の予備的な分析結果を述べる。

先行研究 [9] では, 応答が適切であったかどうかでシステムを評価しているが, その場合, ユーザ発話と質問文例とのマッチングと, 質問文例のカバレッジの両方をまとめて評価してことになる。しかしながら, 本評価では, 質問文例のカバレッジは評価対象にしない。これは, 本研究が質問応答機能を作成するツールの研究であるため, コンテンツである質問文例集合は評価の対象外であるからである。したがって, 質問応答記述が与えられたときに, どのくらい正しくユーザ発話と質問文例がマッチでき, 結果として質問応答カテゴリをうまく判定できたかを評価する。すなわち, 発話の理解の精度で評価を行う。各発話の理解の正解は, 書き起こしの理解結果を用いた。すなわち, キーフレーズがすべてユーザの発話に含まれているような質問文例の中で, 最もキーフレーズを多く持つものを選び, その質問文例を持つ質問応答カテゴリを正解とした。キーフレーズレベルでマッチする質問文例がない場合は, 正解は「DB 外」とした。このデータでは最も多く発話されたカテゴリの頻度は 21 であった。

2.4.2 節で記述した方法で理解実験を行った。音声認識の単語誤り率は 26.3% であった。また, 理解率は表 3 に示した通りである。DB 内の発話に対しては, 80.9% の正解率となり, 初期システムとしては十分な精度が得られていると考えられる。しかし, DB 外の発話(ノイズなども含まれている)が誤って

(注1) : <http://palmkit.sourceforge.net/>

正解	理解結果			
	DB内のカテゴリに正しく理解	DB内のカテゴリに誤って理解	DB外	正解率
DB内	675	131	28	80.9%
DB外	—	105	117	—

表3 発話の理解率

DB内のカテゴリと理解される確率が高いので、これを抑制することは今後の課題である。

#### 4. おわりに

ロボットやエージェントの対話行動制御部を構築するためのツールキット RIME-TK 上に、質問応答機能を実現した。質問応答を記述する XML フォーマットを提案し、その記述から質問応答を行うモジュールを自動構築する方法について述べた。予備的な実験の結果、大量のデータを用いず簡単に構築できる初期システムとしては十分な性能を持つことが示唆された。今後は他のシステムのデータの分析を進める。

また、ユーザ発話の書き起こしや音声認識結果を用いて理解モデルを訓練する方法との、必要となるデータの量や労力の観点からの比較も必要である。

現在の実装では、質問応答全体のパフォーマンスは、質問文例内でのキーフレーズの指定に大きく依存している。質問応答コンテンツの設計者が質問文例の作成と効果的なキーフレーズの指定が行えるような改良をしていく必要がある。

また、本研究で評価に用いたデータは、実験室で収集したものであり、先行研究 [1], [9] のように実環境で運用した場合は異なる可能性がある。今後、実環境での運用によるデータ収集を検討していく。

#### 文 献

- [1] T. Cincarek, H. Kawanami, R. Nisimura, A. Lee, H. Saruwatari, and K. Shikano. Development, long-term operation and portability of a real-environment speech-oriented guidance system. *IEICE Transactions*, 91-D(3):576–587, 2008.
- [2] 星野, 中野, 竹内, 長谷川, 加藤, 辻野. タスク指向対話とチャットを統合した対話ロボットインターフェース. *情報科学技術レターズ*, 5:127–130, 2006.
- [3] T. Kawahara, A. Lee, K. Takeda, K. Itou, and K. Shikano. Recent progress of open-source LVCSR engine Julius and Japanese model repository. In *Proc. Interspeech-2004 (IC-SLP)*, pp. 3069–3072, 2004.
- [4] 北村, 戸田, 川波, 李, 猿渡, 鹿野. 公共音声対話システムにおける n-gram と grammar の融合によるタスク拡張. *電子情報通信学会技術研究報告 SP2005-107*, pp. 13–18, 2005.
- [5] 松本, 北内, 山下, 平野, 松田, 高岡, 浅原. 形態素解析システム『茶釜』version 2.3.3 使用説明書. 奈良先端科学技術大学院大学, 2003.
- [6] 中野, 船越, 長谷川, 辻野. オブジェクト指向に基づくロボット・エージェントのマルチドメイン対話行動制御モジュール構築ツール RIME-TK. *人工知能学会研究会資料 SIG-SLUD-54*, 2008.
- [7] M. Nakano, K. Funakoshi, Y. Hasegawa, and H. Tsujino. A framework for building conversational agents based on a multi-expert model. In *Proc. 9th SIGdial Workshop*, 2008.
- [8] 西村, 内田, 李, 猿渡, 鹿野. Julius を用いた学内案内ロボット用音声対話システムの作成. *電子情報通信学会技術研究報告 SP2001-99/NLC2001-64*, pp. 93–98, 2001.

- [9] 西村, 西原, 鶴身, 李, 猿渡, 鹿野. 実環境研究プラットフォームとしての音声情報案内システムの運用. *電子情報通信学会論文誌, J87-D-II(3):789–798*, 2004.
- [10] Y. Nishimura, S. Minotsu, H. Dohi, M. Ishizuka, M. Nakano, K. Funakoshi, J. Takeuchi, Y. Hasegawa, and H. Tsujino. A markup language for describing interactive humanoid robot presentations. In *Proc. IUI'07*, 2007.
- [11] I. M. O'Neill and M. F. McTear. Object-oriented modelling of spoken language dialogue systems. *Natural Language Engineering*, 6(3&4):341–362, 2001.
- [12] 竹内, ツインツアレク, 川波, 猿渡, 鹿野. 音声案内システムにおける質問応答データベースの最適化-音声認識結果を用いた質問応答データベースによる応答生成と最適化-. *電子情報通信学会技術研究報告 SP2007-146*, pp. 295–300, 2007.
- [13] 吉見, 南角, 李, 徳田. 音声対話システムにおける条件付確率場に基づく自動応答文選択. *日本音響学会 2008 年春季研究発表会論文集*, 2008.
- [14] 鳥井, 長谷川, 中野, 中臺, 辻野. 人・ロボットインタラクションシステムの為のミドルウェアの開発. *計測自動制御学会第7回システムインテグレーション部門 講演会 (SI2006)*, pp. 2D2-1, 2006.