「マルチメディア，分散，協調とモバイル
(DICOMO2022)シンポジウム」 令和4年7月

複数シナリオに対応したタスク指向型対話システムの開発と介護施設向け見守りロボットへの応用

須崎 孝嗣[1] 河村 優周[1] 沼尾 雅之[1]

概要：タスク指向型対話システムは旅行案内や検索などの課題の解決に有用である．近年では，対話システムに深層学習を適用して End-to-End のシステムを開発する研究が盛んに行われている．しかし，深層学習法では大量のコーパスが必要であり，ルールベースでは対話システム分野の専門家でないとシナリオの追加や編集を行うことが難しいという問題がある．現在のタスク指向型対話システムの課題として，複数のシナリオへの遷移，各シナリオでの対話が柔軟に記述できること，さらに，シナリオ内での Redo, Undo, Skip や，シナリオ間の移動や復帰などといった，メタ制御が必要である．さらに，実際の対話では，話声のテキスト情報だけではなく，顔の表情や声のトーンなどのマルチモーダルな情報も利用して，自然な会話を成り立たせている．そこで本研究では，複数のシナリオにおいて容易にシナリオの作成・変更ができ，マルチモーダル情報も，話声テキスト情報と同様に扱えるような対話システムの開発を行った．対話制御は状態遷移マシンで行っており，遷移ルールはルールベースで行っており，シナリオは独自のルールを用いた XML ファイルによって定義した．さらに，シナリオにタイムアウトや繰り返しキャンセル等のメタなコマンドを用意することで柔軟に対話を進行できるようにした．また，このシステムを介護施設に設置することによって高齢者の見守りへ応用する．

Development of Task-Oriented Dialog System for Multiple Scenarios and Application to a Robot for Nursing Home

TAKATSUGU SUZAKI[1] MASAHIRO KAWAMURA[1] MASAYUKI NUMAO[1]

1. はじめに

1.1 対話システム

対話システムは人間とロボットとの容易で柔軟なインタラクションを可能にする．対話システムには大きく分けてタスク指向型対話システムと非タスク指向型対話システムがある [1]．タスク指向型対話システムは，あるドメインにおいてユーザのタスクを解決するという目的があり，非タスク指向型対話システムはオープンドメインにおける対話システムで雑談等のユーザを満足させるように適切な対話を行うことを目的とする．本稿では，高齢者見守りシステムへの応用を目指した．

タスク指向型対話システムのアプローチはパイプライン型と End-to-End のシステムがある．一般的な音声対話システムのアーキテクチャを図 1 に示す．パイプライン型は一般に音声認識，自然言語理解，対話管理，自然言語生成，音声合成のモジュールから構成される．音声認識では，ユーザの発話をテキストに変換する．自然言語理解では，ユーザの発話テキストからユーザ意図の理解とフレーム表現と呼ばれるスロットとその値の意味構造へと変換を行う．対話管理は，フレーム表現が入力された時，対話状態を更新し，次のロボットの行動を選択する．自然言語生成では，選択された行動からロボットの発話内容 (テキスト) を生成する．音声合成では，ロボットの発話内容を音声に変換する．パイプライン型では人間が解釈・制御しやすいというメリットがある反面，ドメインを変更する時，各モジュールのパラメータを調整しなければならないというデメリットがある．End-to-End システムは，一つのモデルでユーザの発話を入力にロボットの発話を出力するシステムであり，大量の対話コーパスデータによる事前学習が必要とされる．各モジュールの調整・学習は必要はないが，ブラックボッ

[1] 電気通信大学院 情報・ネットワーク工学専攻

クス化してしまうので対話の制御が難しくなり, 専門家でない人がシナリオを調整するのには適していない.

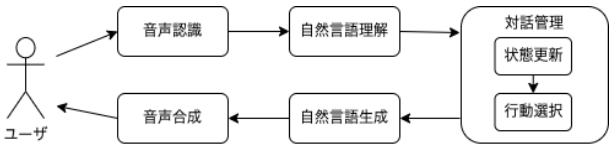

図 1 一般的な音声対話システム

さらに, 音声による情報に加えて, 画像等のマルチモーダルな入出力情報も取り入れることによってユーザにより良いインターフェースを提供でき, マルチモーダル情報の対話システムの研究も盛んに行われている [2].

1.2 見守りロボットへの応用

高齢化が進んでおり, 認知症患者数の増加が問題となっている. 認知症は早期発見することで進行の予防や症状の改善が見込める. しかし, 認知症の発見には認知症専門医による診断が必要であり, 高齢化が進んでいく中で専門医への負担はさらに大きいものとなる. 長谷川式簡易知能評価スケール (HDS-R) は認知機能の程度を測ることが可能である [3]. HDS-R は 9 つの質問を順番にするだけのように見られるが, 実は, 解答に応じてさまざまな場合分けと分岐が必要であり, いままでの直線的な質問ー応答フロー言語, 例えばチャットボット用シナリオ記述言語 AIML では, 記述することが難しい. また, 日常会話によって認知症予備軍を発見することも可能である [4]. 対話シナリオは対話システム分野の専門家するよりも現地の介護者や専門医に定義してもらう方が有効であり, 対話シナリオを柔軟に定義できるようなシステムが必要である.

そこで本研究では複数のシナリオを柔軟に編集できる, マルチモーダル情報を用いたパイプライン型のタスク指向型対話システムを開発する.

2. 関連研究

拡張性のあるシナリオを定義できる対話システムの研究を紹介する

2.1 対話記述言語

AIML はルールを記述する言語である [5]. AIML はパターンマッチングによってルールを記述し, ロボットが返答をする. 入力と出力を対にしてルールを与えられるため, 容易にシナリオを記述することができる. また, Python のパッケージが利用でき, xAIML SUNABA[6] は GUI によるシナリオの記述が可能である. しかし, シナリオが複雑になるとシナリオの可読性が下がるため, シナリオの追加・変更, 対話状態の推定が困難になる.

2.2 音声対話システム

PyDial[7] はドメイン独立性、設定可能性、拡張性という主要な原則に従って、キーボード入力や音声認識を用いた対話システムを提案した. 対話制御にはルールベースによる手法または統計的な手法 (ガウス過程による強化学習) を使用することができ, ドメイン毎にモジュールを作成することによってドメインの独立性を保っている. また設定ファイルによってドメインのモジュールのパラメータを設定できる. また, ドメインモジュールのインターフェースクラスを継承することによって拡張できる.

2.3 マルチモーダル対話システム

桂田らはマルチモーダル対話システムの標準仕様である MMI(MuitiModal Interaction) アーキテクチャを提案した [8]. MMI は 6 階層の階層型アーキテクチャであり, 1 層の入出力デバイスの制御, 2 層の入力解釈, 出力生成, 3 層のデータの統合・分化, 最後に 4,5 層の対話管理を行っている. また, Web ブラウザをインタフーフェースとして使用した. 対話記述言語には XISL, SCXML が用いられている.

本研究では, PyDial, MMI のアーキテクチャを参考にしてシステムの提案を行う. さらに, シナリオの繰り返し・キャンセル等の現在の状態を操作できるコマンドを使用することによって対話の柔軟性を向上する.

3. タスク指向型対話システムにおけるシナリオ要件

タスク指向型対話システムにおけるシナリオの仕様を HDS-R を例に挙げていく. まず, HDS-R の状態遷移図を図 2 に示す.

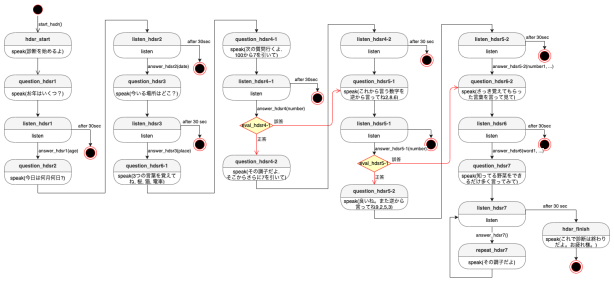

図 2 HDS-R の状態遷移図

3.1 対話シナリオの記述能力

まず, 順次に質問を挙げていくような行動選択が必要である. 次に HDS-R の 4, 5 問目では最初の質問が合っているかに応じて次の質問をするか, その質問を飛ばして次の問題に遷移する必要がある. そのため, 問題が合っているかどうかで条件付けを行い, 遷移する状態を変えなければならない. また, 問題の正誤判定を対話状態に保持できるようにする必要がある. 最後の野菜の名前を出来る限り聞

く質問では, ユーザが答えられなくなるまで, 同じ質問に遷移する必要がある. また, 同じ言葉を答えるだけではユーザが飽きてしまうため, 同じ質問でも言い方を変えられるような機能が必要である.

3.2 メタ対話制御

状態遷移図においてユーザの答えを待つ状態のように, 30 秒後にタイムアウトとして正常に遷移する時と別の遷移が必要になる場合ある. 他にも, ユーザが質問を聞き取れなかった場合や, その内容が分からない時にその質問を飛ばす場合, 別の用事を思い出し, 現在のシナリオを終了させたい場合が考えられる. これらの遷移は固有のドメインに関わらず, 共通の遷移方法として持っておくことによってシナリオの記述が少なくなり, 可読性の高いシナリオが実現できると考えられる.

3.3 シナリオ間の移動と階層型シナリオ

シナリオが増えていくと, 共通するシナリオが発生することが考えられる. 例えば, ロボット「何ができるの？」と質問した場合, ロボットが持っているシナリオを紹介し, ユーザは紹介されたシナリオを実行したい場合が考えられる. このシナリオを実現するために, スタックのようにシナリオを階層型にしてシナリオ間での移動が必要になる.

3.4 マルチモーダル入出力

話声テキスト以外にもマルチモーダル情報を加えることによってユーザの意図理解が向上することが考えられる. 例えば, ユーザが質問に悩んでいる場合, カメラによる表情認識を加えることによって, 音声認識が反応しない場合でも, ロボットがユーザが悩んでいると理解できタイムアウトするのではなく, 同じ質問を繰り返す, または理解しやすい質問に変えると言ったことができるようになる. また, 実際の対話でも, 質問では画像，応答では，ジェスチャーやボタンなどが使われていて，それらに対処できるようにする必要がある.

4. 提案システム

4.1 システム概要

提案システムを図 3 に示す. 対話はユーザとロボットが 1 対 1 で行われる. 提案システムは 4 つのモジュールから構成されている. モジュールの各役割は既存の対話システムと似ており, システムはインターフェース管理, 入力解釈, 対話管理, 出力生成から構成されている. インターフェース管理では, ロボットの行動に使用するアクチュエータ群 (スピーカ, ディスプレイ等)· ユーザの要求を理解するためのセンサ群 (マイク, カメラ等) の制御, 各センサから取得したデータの統合を行う. 入力解釈では, 自然言語理解等のインターフェースから取得したデータを意味理解し, 構造化する. 対話管理では, シナリオの状態の更新し, ロボットの行動を選択する. 出力生成では, 自然言語生成等対話状態からロボットの行動を生成する.

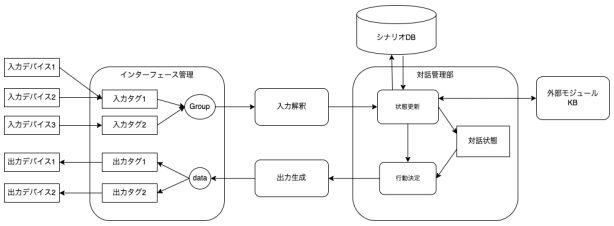

図 3 システム概要

4.2 マルチモーダル対話

マルチモーダルな対話を成立させるために, 入出力デバイス, 解釈・出力生成モジュールを制御する必要がある. まず, 対話管理からのロボットの行動群, 要求するスロット群データが出力生成へ送られる.

4.2.1 出力生成

出力生成では, 対話シナリオに記述された出力タグ・行動内容によって生成モジュールを選択し, 行動内容をその生成モジュールの入力として処理したデータをインターフェース管理に送る. 要求スロット群はそのままインターフェース管理に渡す.

4.2.2 インターフェース管理

インターフェース管理にまとめて渡される. 渡されたロボットの行動データ群は, データ群に含まれる出力タグと紐付けられた出力デバイスモジュールに入力され各行動を同時に実行する. 全ての行動が実行完了した後, ユーザへ入力を求める.

ユーザへの入力要求は, まず送られた要求スロットに必要な一つ以上の入力タグを取得し, それぞれの入力タグと紐付けられた入力デバイスモジュールを同時に実行する. ここで入力タグとデバイスモジュールは 1 対 N の対応であり, デバイスモジュールがどれか一つ入力が入ると, 入力タグにデバイスの値が送られ, スロットに必要な全ての入力タグが揃った時, 入力解釈に送信する.

4.2.3 入力解釈

入力解釈の役割は 2 つあり, 意図の推定とスロット値の抽出がある. 意図が定まっていない場合は意図の推定を行い, 意図が決まっている場合, メタコマンドの認識後, 要求されたスロットに応じてスロット値の抽出をする.

受け取った入力タグデータをスロット毎に割り当てられた解釈モジュールに渡し, 意味的な構造に変換する. その後, 対話管理に送信し, このサイクルを繰り返すことによってユーザと対話を行う.

4.3 対話管理

対話管理では, 対話状態更新と, 行動選択を行っており,

有限状態マシンを用いて対話を制御している. 対話状態には現在の意図とフレーム表現を用いている. 有限状態マシンにおける一つの状態の目的は, あるフレーム表現を埋めることである. 対話状態は現在行われているシナリオをが終わるまで保持される. 状態遷移方法は 2 つあり, あるフレーム表現が入力された場合とタイムアウト, 繰り返し, キャンセル等のメタコマンドを用いた場合によって状態遷移を行っている. フレーム表現が入力された時の状態遷移方法についてはフレームのスロット値によって決まり (例えば, スロット値が入力されたか), シナリオで定義できる. メタコマンドは, シナリオを柔軟にし, シナリオ定義時の複雑性を解消できる. また, 入力解釈からの入力だけでなく, 外部の KB, モジュールへ接続し, データを連携することも可能である.

4.4 シナリオ定義

シナリオの定義は独自のルールの XML ファイルを記述することによって定義できる. シナリオにはドメイン, ユーザに要求するフレーム表現の制御, そのフレーム表現をユーザに求めるためのロボットの行動, 外部との連携方法が定義されている. HDS-R によるシナリオ記述例を示す.

```
1  <scenario name="hdsr">
2      <sequential>
3          <frame name="greeting" order='1'>
4              <actions>
5                  <action device="speaker"> こ
                       んにちは、今から診断を始め
                       ます。</action>
6              </actions>
7          </frame>
8          <frame name="qname" order="2">
9              <actions>
10                 <action device="speaker"> お
                       名前は何ですか？</action>
11             </actions>
12             <request timeout="15">
13                 <slot name="person"/>
14             </request>
15         </frame>
16        ...
17    </sequential>
18 </scenario>
```

scenario タグはルートタグであり, 属性値 *name* は意図を表す. *sequential* タグによって子要素が全て実行完了するまで順に選択していく. *sequential* が行動選択の役割をしており, *sequential* の他にも, 条件によって子要素を選択する *conditional*, ランダムに子要素を選択する *random*, 条件を満たすまで選択し続ける *loop* を用意した. これらの行動選択タグは子要素に行動選択タグ, *frame* タグ等を定義でき, 複雑な状態遷移のあるシナリオにも対応することができる. また, 行動選択のタグは Python によって実装されており, 親クラスを継承することによって拡張機能を実装することができる.

frame タグはフレーム表現を埋めるために使用される. *actions* タグはロボットの行動をまとめるタグであり子要素の *action* タグの内容を同時に実行する. *request* タグはユーザに埋めることを期待するスロット群を指定する. *slot* タグはスロットを定義し, デフォルト値や値がスロット値のリスト化等の属性を指定できる.

5. 実装

言語は Python3 で実装した. 介護施設で対話ロボットを設置するためにいくつかのシナリオを実装した.

5.1 対話管理

介護施設での使用を目的に自己紹介, HDS-R, 検温, スケジュール確認, ゲーム等のシナリオを定義した. また, ログ収集のために MongoDB を使用した.

5.2 インタフェース管理

入力にはマイクからの音声認識, キーボード入力に加えて検温シナリオのために可視光カメラ, サーモカメラ, 気温, 湿度センサを用い, ボタンによるメタコマンドの入力機能を実装した. 出力にはスピーカでの音声合成, 画像表示を使用した. 音声認識, 音声合成にはそれぞれ Google Cloud Speech API の Speech-to-Text[9], Text-to-Speech[10] を利用した.

5.3 入力解釈

入力解釈では, ユーザの発話テキストを意味理解する自然言語理解モジュール, サーモカメラ, 気温, 湿度から体温を推定する体温推定モジュールを実装した. 自然言語理解ではルールベースによる方法を用いており, 形態素解析, 正規表現を使用したテキスト抽出, KB を用いたキーワード抽出を実装した. 体温推定モジュールでは, 重回帰分析による機械学習を用いた.

5.4 出力生成

出力生成では, ロボットの発話テキストを生成する自然言語生成モジュール, 画像表示のための画像取得モジュールを実装した. 自然言語生成モジュールはシナリオで定義したテキストを対話状態に代入して発話テキストを生成する. 画像表示モジュールではパスか URL を指定することによって画像を取得する.

6. まとめ

本研究は簡単に記述でき, 柔軟性のあるシナリオを定義できるマルチモーダル対話システムを開発した. シナリオ

は独自のルールを用いた XML ファイルを対話管理で読み込むことによって対話を制御できるようにした．また, 介護施設に設置することを目標に幾つかのシナリオを定義した．

7. 今後の課題

介護施設で実証実験を行うことによって更なるシステムの改善を行っていく予定である．評価には高齢者との対話のタスク達成度, アンケートによる満足度によって評価する．また, XML のタグや, メタコマンドを追加していくことで, 更に柔軟性, 簡単性を向上できると考えている．さらに, GUI を実装することによって容易にシナリオの追加・編集を容易にすることを考えている．

参考文献

[1] Hongshen Chen et al.A Survey on Dialogue Systems: Recent Advances and New Frontiers. ACM SIGKDD Explorations Newsletter. Volume 19. 2018.

[2] Ming-Hao Y., Jian-Hua T.: Data fusion methods in multimodal human computer dialog, Virtual Reality & Intelligent Hardware, Volume 1, Issue 1, Pages 21-38, 2019.

[3] 加藤 伸司, 下垣 光, 小野寺 淳志ほか: 改訂長谷川式簡易知能評価スケール（HDS-R）の作成, 老年精神医学雑誌,pp.1339–47(1991).

[4] 株式会社こころみ. CANDy(日常会話式認知機能評価). http://cocolomi.net/candy/ (2022/03/22)

[5] Roberts, G., Beber, G. (Eds.): Parsing the Turing Test, Wallace, R.S. : The Anatomy of A.L.I.C.E.,pp 181–210, Splinger, 2007.

[6] SUNABA(オンライン): 入手先 ¡https://docs.xaiml.docomo-dialog.com/¿, (参照: 2022-05-23)

[7] Stefan U., Lina R., Pei-Hao S., et al.: Pydial: A multi-domain sta- 73 tistical dialogue system toolkit. In Proceedings of ACL 2017, System Demonstrations, pages 73–78, 2017.

[8] 工藤正志, 桂田浩一, 新田恒雄, 入部百合絵: MMI6 階層モデルに準拠した Web ベース MMI システムの開発, FIT2009 情報科学技術フォーラム，E-039 (2009-9).

[9] Google: Speech-to-Text（オンライン），入手先 ¡https://cloud.google.com/speech-to-text/?hl=ja¿, (参照: 2022-05-23)

[10] Google: Text-to-Speech, （オンライン），入手先 ¡https://cloud.google.com/text-to-speech/?hl=ja¿, (参照: 2022-05-23)