

料理レシピ共有サービスのための 音声対話シナリオ編集手法

三浦 旬夏¹ 梶岡 慎輔¹ 山本 大介¹ 高橋 直久¹

概要: 近年では、投稿者のアイデアが詰まったレシピが閲覧できるサービスの楽天レシピが普及している。また、本研究室では、音声インタラクションシステム構築ツールキット MMDAgent を用いた研究を行っている。MMDAgent は、FST 形式の対話シナリオを編集することでユーザが対話の内容を自由に構成することができる。しかし、FST 形式の対話シナリオの編集には知識が必要であり、一般ユーザが編集するには困難である。簡単にユーザ生成できる方法として Google Blockly を用いた編集システムが挙げられる。しかし、対話シナリオは対話の文脈に基づいた FST 形式にのみ対応しており、料理に関する音声対話を作成することは困難であった。この問題を解決するために、本論文では、料理レシピ共有サービスのための Google Blockly の技術を利用した音声対話シナリオ編集手法を提案する。また、提案システムに基づいて実装したプロトタイプシステムを用いて、評価実験を行った。その結果、提案システムを使用することで、短い時間で簡単に、料理レシピの対話シナリオを作成や編集できることがわかった。さらに、提案システムは、Web ブラウザと比べて、必ずしも有用であるという結果にはならなかった。そのため、字幕機能や自動スクロール機能などの改良が必要である。

A Method for Editing Speech Dialogue Scenarios for Cooking Recipe Sharing Services

Shunka MIURA¹ Shinsuke KAJIOKA¹ Daisuke YAMAMOTO¹ Naohisa TAKAHASHI¹

1. はじめに

近年では、楽天レシピやクックパッドなどのユーザ投稿型レシピサイトが普及している。投稿されているレシピは一般的な家庭料理からおせちやクリスマスなどのイベントの料理、節約料理、健康を意識した料理など幅広いジャンルがある。また、近年、音声対話システムも普及されている。音声対話システムは人間の発話を入力として、それに対して応答を返す一問一答形式のシステムである。Siri や Amazon Alexa があり、一般ユーザに広く使われ始めている。これに付随し、本研究室では、音声インタラクションシステム構築ツールキット MMDAgent[1] を用いた研究を行っている。MMDAgent は、画面上に表示される 3D キャラクタと音声によって対話を行うことができる。また、対

話シナリオを編集することにより、ユーザが対話の内容を自由に変更することができる。対話シナリオは FST と呼ばれる有限状態遷移の形式で記述し、状態番号、次状態番号、遷移条件、出力の 4 つの項目を組とした行の集合で表す。そのため、ユーザが編集するためには FST 形式について学習する必要がある。そこで、簡単にユーザ生成できる方法として Google 社が提供する Google Blockly を用いた編集システムが挙げられる。Google Blockly はビジュアルプログラミング言語の一つで、パズルのようにブロックを組み合わせることで、プログラムを作成することができ、命令文やコマンド名を覚えたり、打ち込んだりする必要がないという特徴がある。

一方、従来手法の対話シナリオは対話の文脈に基づいた FST 形式にのみ対応しており、料理に関する音声対話を作成することは困難であった。さらに、料理は手作業で行うため、レシピサイトを見ながらスムーズに料理をすることが困難である。また、既存システムを使用すると、公式の

¹ 名古屋工業大学大学院工学研究科
Graduate School of Engineering, Nagoya Institute of Technology

おすすめのレシピを自動的に選択され、作りたいレシピを自由に選択し、音声対話で料理をすることができない。そこで、本研究では楽天レシピの投稿者がBlocklyを用いることにより、簡単に料理レシピの音声対話シナリオを作成や編集を行い、音声対話によってより料理をしやすくすることを目的とした料理レシピ共有サービスのための音声対話シナリオ編集手法を提案する。提案手法のアプローチを以下に示す。

アプローチ 1 Web から材料や手順などの情報を取得し、Blockly の形式に変換する機能を実装する。

アプローチ 2 料理の対話シナリオを記述するために、Blockly で利用可能な料理ブロックを提案する。

アプローチ 3 Blockly の形式から FST を作成する。

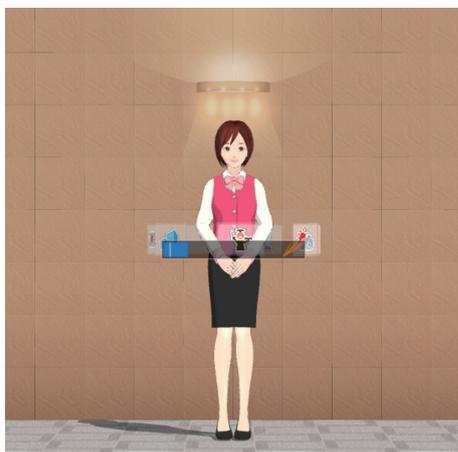


図 1 MMDAgent

2. 関連研究

EFDE[2] は、タブレット端末のタッチ操作で状態遷移図を描くことにより、対話シナリオを作成できるシステムである。データを入力することで手軽に対話シナリオを編集することができるという点が本研究と同様である。しかし、EFDE では状態遷移図を描くことで対話シナリオを編集しているが、本研究ではブロックを組み合わせることにより対話シナリオを編集する手法を提案している。

MMDAE[3] とは、音声対話エージェントのための Web ブラウザを用いたシナリオエディタである。対話シナリオをより簡単に扱うことができるという点では本研究と同様である。しかし、MMDAE は FST 形式で書かれた対話シナリオを編集しやすいようなエディタであり、本研究では FST 形式より編集しやすい対話シナリオ編集システムを提案している。

ブロック型ビジュアルプログラミング機能を有する音声

対話シナリオ編集システム [4] は、Google Blockly を用いることで簡単に対話シナリオの作成や編集を行うことができるシステムである。ブロックを使用することで対話シナリオの構造が見やすくなり、把握しやすいという点では本研究と同様である。しかし、このシステムは対話の文脈に基づく状態遷移にのみ対応しており、本研究では料理に関する音声対話システムを提案している。

スマートグラスと音声対話による料理支援 [5] は、ハンドジェスチャ操作や音声認識による操作で、ユーザが料理支援を受けるシステムを提案している。料理の音声対話システムを提案しているという点では本研究と同様である。しかし、このシステムはシステム提案者があらかじめ対話シナリオを記述しているが、本研究では誰でも対話シナリオを作成できるシステムを提案している。

ロボット対話と IoT 機器を活用したストレスフリーキッチンの提案 [6] は、ロボットの対話によって、料理中に発生する待ち時間や片付けなどのストレスを解決するシステムを提案している。料理の音声対話システムを提案している点では本研究と同様である。しかし、このシステムはロボットを活用しているが、本研究では 3D エージェントを使用しているシステムを提案している。

音声認識とジェスチャ機能を用いた料理者支援 [7] は、Microsoft 社の体感型インタフェース Kinect を用いて動きの認識をし、ジェスチャ機能により料理者を支援するシステムを提案している。料理者を支援する点では本研究と同様である。しかし、このシステムは料理を作るために最低限必要な機能を実装しているが、本研究ではタイマー機能など料理をする際に便利な機能を含めたシステムを提案している。

3. 提案手法

3.1 提案システムの構成

提案システムの構成図を図 2 に示す。

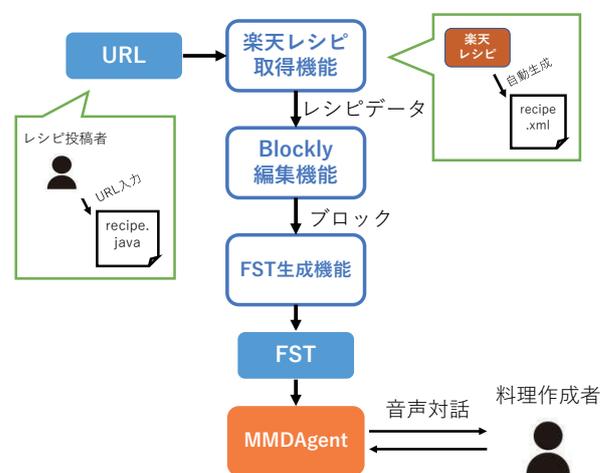


図 2 システム構成図

提案システムとして、楽天レシピ取得機能、Blockly 編集機能、FST 生成機能を実装した。まず、楽天レシピ取得機能によってレシピを取得し、Blockly で使用する XML ファイルを自動生成する。Blockly 編集機能によって、料理ブロックを用いて対話シナリオを作成する。FST 生成機能によって作成した対話シナリオを FST ファイルに変換する。その後、出力し、MMDAgent で読み込み、料理作成者と料理の音声対話をする流れとなる。

3.2 提案システムの機能

3.2.1 楽天レシピ取得機能

スペースの削除機能と単位変換機能を作成し、抽出機能と組み合わせ、レシピの URL を入力することで、レシピをブロックに自動生成する機能である。

3.2.2 抽出機能

Java の jsoup のライブラリを使用し、楽天レシピのレシピ投稿文の中にある材料名と材料の大きさ、手順を抽出する機能である。jsoup は Java で HTML の解析や編集を行うライブラリの中で、URL を指定することで Web ブラウザで指定した部分を抽出することができる。抽出部分を図 3 と図 4、図 5、図 6 に示す。例えば、図 4 のようにタグを指定することで、豚こま肉という値を抽出することができる。材料の大きさ、手順も同様に取り出すことができる。



図 3 抽出部分例

```
<span class="recipe_material_item_name">豚こま肉</span>
```

図 4 材料名の HTML

```
<span class="recipe_material_item_serving">1 5 0g</span>
```

図 5 材料の大きさの HTML

```
<span class="recipe_howto_text">
キャベツはざく切り、ニンジンも短冊切り、ピーマンは縦半分
に切って種を取り除き、細切りにしておきましょう。
</span>
```

図 6 手順の HTML

3.2.3 スペース削除機能

抽出機能を使用して抽出したものを全角から半角に変換し、半角のスペースを削除する機能である。この機能を使用することで、Blockly で使用するときエラーを防ぐことができる。

3.2.4 単位変換機能

スペース削除機能を使用した材料の単位を記号から日本語に変換する機能である。レシピ投稿文には“g”や“1/4”など記号や数字で表示されている場合がある。しかし、その対話シナリオをメイちゃんに使用すると“ジー”や“1 スラッシュ 4”と発話し、不具合が生じる。そのため、“グラム”や“4分の1”と発話するようにした。

3.3 Blockly 編集機能

料理ブロックを作成することで、料理に関する対話を誰でも簡単に Blockly で編集できる機能である。従来手法では、料理を音声対話に組み込むことを想定しておらず、料理を扱うブロックが存在していなかった。そこで、Blockly に新たに料理ブロックを作成した。料理ブロックの作成には Blockly Developer Tools を使用した。Blockly Developer Tools を図 7 に示す。Blockly Developer Tools では、元となるブロックにブロックの名前、形、色などを記述すると、ブロックのプレビューと JavaScript 形式のブロックの定義が表示される。命令コマンドのカテゴリごとにブロックの色を分け、命令コマンドの意味を分かりやすい表現でブロックに記す。また、料理ブロックには以下の 5 個のブロックがある。

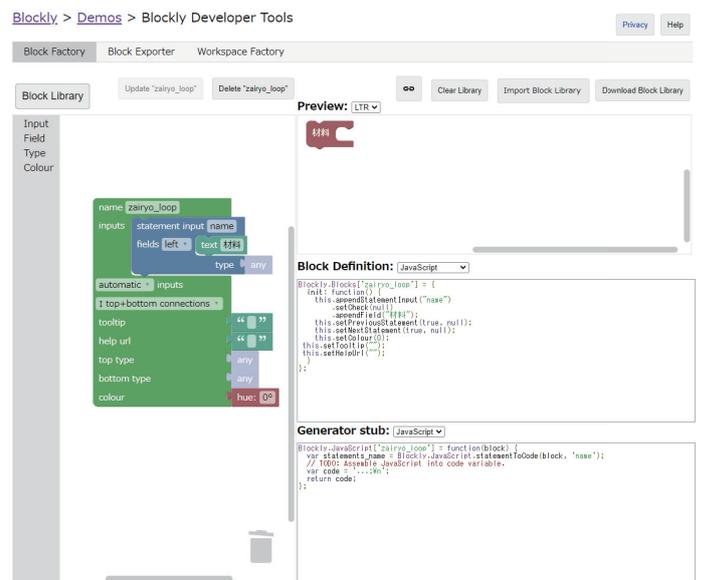


図 7 Blockly Developer Tools

3.3.1 手順と質問ブロック

Google Blockly の Mutators 機能を利用する。Mutators 機能とは、ユーザが動的にブロックの形状を変更できる機

能である。作成した手順と質問ブロックは図8である。“ダイアログボタン”という青色の歯車のマークを押すと図9のように“Mutator UI”が表示される。“Mutator UI”内の“手順”ブロックに“質問”ブロックを接続すると、図10のように手順と質問ブロックが変化する。手順と質問ブロックは手順の部分に手順の説明が挿入され、質問の部分に料理用語がドロップダウン形式になっている質問ブロックが挿入される。



図 8 手順と質問ブロック



図 9 “Mutator UI” の表示



図 10 手順と質問ブロック変形後

3.3.2 質問ブロック

ユーザの質問ブロックを図11に、メイちゃんの質問ブロックを図13に示す。青色のユーザの質問ブロックには定型文の料理用語がドロップダウン形式になっているブロックであり、ピンク色のメイちゃんの質問ブロックには定型文の料理用語の説明がドロップダウン形式になっているブロックである。ドロップダウン使用後のユーザ、メイちゃんの質問ブロックを図12と図14に示す。レタスクラブを参考にし、それぞれ50音作成した。マウスで選択するためタイピングを減らすことができ、音声対話の作成をより簡単にした。また、音声対話で質問できるため、料理を中断することなくスムーズに料理することができる。質問ブロックを使用するときは手順の説明から想定される質問を考え、挿入する。



図 11 質問ブロック (ユーザ)

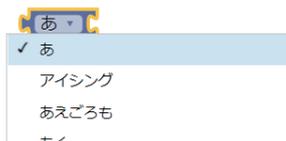


図 12 ドロップダウン使用後



図 13 質問ブロック (メイちゃん)

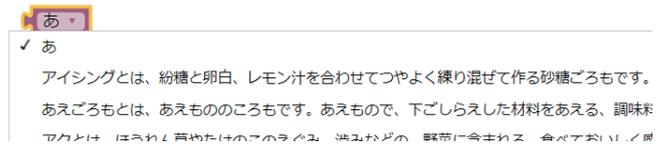


図 14 ドロップダウン使用後

3.3.3 時間と費用ブロック

時間と費用ブロックを図15に示す。時間と費用ブロックはレシピ投稿文に表示されている時間と費用を説明が入るブロックである。これを使用することで時間と費用の説明部分を一目で判断することができ、対話シナリオの構造がわかりやすくなる。



図 15 時間と費用ブロック

3.3.4 材料ブロック

材料ブロックを図16に示す。材料ブロックはレシピ投稿文に表示されている材料の人数と材料の説明が入るブロックである。これを使用することで材料の説明部分を一目で判断することができ、対話シナリオの構造がわかりやすくなる。



図 16 材料ブロック

3.3.5 レシピ画像または URL ブロック

レシピ画像または URL ブロックを図17に示す。レシピ画像または URL ブロックはレシピ画像またはレシピの URL を入力することで画像やサイトを開き、表示するブロックである。



図 17 レシピ画像または URL ブロック

3.4 FST 生成機能

従来の FST 生成機能を使用することで、それぞれの料理ブロックに FST を対応付ける機能である。

3.4.1 質問ブロックを挿入した手順と質問ブロック

手順と質問ブロックのアルゴリズムを図 18 と図 19 に示す。また、図 19 に対応した遷移の場合分けを以下に示す。

パターン 1 状態 A, X (繰り返しあり), B と遷移する場合

パターン 2 状態 A, X (繰り返しあり), Q1, B と遷移する場合

パターン 3 状態 A, X (繰り返しあり), Q1, X (繰り返しあり), B と遷移する場合



図 18 手順と質問ブロック

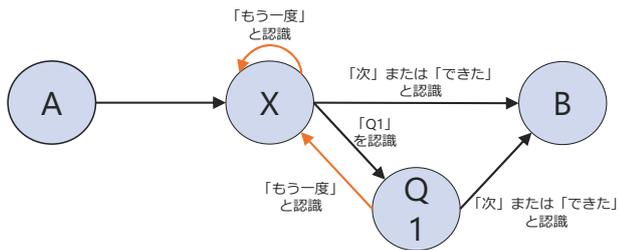


図 19 図 18 に対応した状態遷移図

3.4.2 時間と費用ブロック

時間と費用ブロックのアルゴリズムを図 20 と図 21 に示す。この例では、状態 A, X (繰り返しあり), B と遷移する。また、材料ブロックも同様のアルゴリズムである。



図 20 時間と費用ブロック

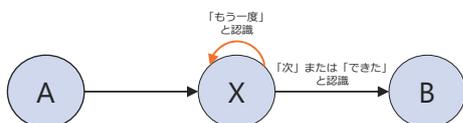


図 21 図 20 に対応した状態遷移図

3.4.3 レシピ画像または URL ブロック

レシピ画像または URL ブロックのアルゴリズムを図 22 と図 23 に示す。この例では、状態 A, X, B と遷移する。

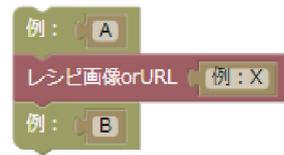


図 22 レシピ画像または URL ブロック

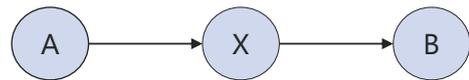


図 23 図 22 に対応した状態遷移図

3.5 データ構造

データ構造の構造図を図 24 と図 25, 図 26 に、それぞれの説明を以下に示す。

図 24 料理ブロックが Blockly Developer Tools で JavaScript データとして作成される。ブロックの名前や形、色などを定義することができる。

図 25 図 24 の JavaScript データを Blockly で読み込み、HTML 形式に変換し、料理ブロックを表示させる。そのため、Blockly で料理ブロックを使用できる。

図 26 図 25 のブロックは MMDAgent では利用できない。そのため、料理ブロックを FST 形式に変換し、FST を料理ブロックに対応付けることで簡単に料理に関する FST を生成することができる。

```
Blockly.Blocks['zairyo_loop'] = {
  init: function() {
    this.appendStatementInput("name")
    this.setCheck(null)
    this.appendField("材料");
    this.setPreviousStatement(true, null);
    this.setNextStatement(true, null);
    this.setColour(0);
    this.setTooltip("");
    this.setHelpUrl("");
  }
};
```

図 24 JavaScript



図 25 Blockly

```

1 101 <eps> SYNTH_START|mei|mei_voice_normal| (材料の説明)
101 102 <eps> <eps>
102 103 RECOG_EVENT_STOP|次 <eps>
102 104 RECOG_EVENT_STOP|もう一度 <eps>
104 1 <eps> <eps>
103 105 <eps> <eps>
105 2 <eps> <eps>

```

図 26 FST

4. プロトタイプシステム

システムの開発は、Windows10 上で Eclipse の環境の下で行い、プログラミング言語として Blockly は JavaScript を、楽天レシピ抽出機能では Java を使用した。プロトタイプシステムは編集システムは Web ブラウザ上で利用可能で Google Chrome97.0.4692.71 で動作を確認した。また、作成された料理の対話シナリオは MMDAgent version1.7[8] で動作を確認した。プロトタイプシステムのイメージを図 27 に、動作画面を図 28 に示す。プロトタイプシステムは、提案システムの料理レシピ共有サービスのための音声対話シナリオ編集手法を実装した。

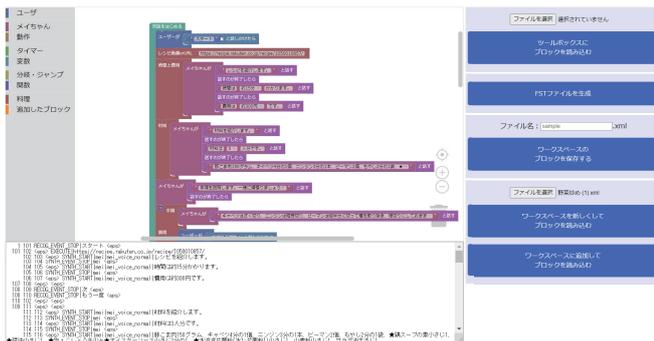


図 27 プロトタイプシステム

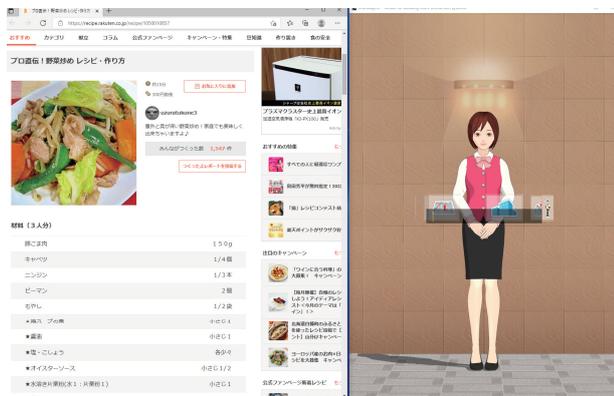


図 28 プロトタイプシステム動作画面 (図は楽天レシピから引用)

5. 評価実験

実験の目的は、提案システムを使用することにより料理

がしやすくなったか、レシピの音声対話シナリオの作成や編集が簡単になったかを検証することである。

5.1 評価実験 1(料理の音声対話の精度)

5.1.1 評価実験の方法

被験者(本学学生 8 人)に、指定した楽天レシピを Web ブラウザと提案システムを使用して料理しやすいか想定してもらった。実験後、料理の工程ごとに 5 段階のアンケート評価を行った。順番に影響されないように、被験者をグループ A とグループ B に分け、グループ A は先に Web ブラウザを使用して評価実験を行い、その後、提案システムを使用して行った。反対に、グループ B は先に提案システムを使用して評価実験を行い、その後、Web ブラウザを使用して行った。

5.1.2 指定したレシピ

使用したレシピを図 29 と図 30 に示す。今回は、包丁を使う作業が多いレシピである野菜炒めと火を使う作業が多いレシピであるオムライスを選択した。また、使用した野菜炒めのレシピには“ざく切り”や“短冊切り”など料理用語が説明文に出てくるのが特徴である。



図 29 野菜炒め



図 30 オムライス

5.1.3 アンケート内容

Web ブラウザを使用して料理しやすいか想定する場合と、提案システムを使用する場合のそれぞれの場合についてアンケートに答えてもらう。アンケート項目として5項目を用意し、5段階評価を行う。1番から4番は料理の過程ごとに評価を行ってもらい、5番は最後に総評価として行ってもらい。評価基準は、「1(当てはまらない), 2(やや当てはまらない), 3(どちらとも言えない), 4(やや当てはまる), 5(当てはまる)」である。アンケートの内容を表1に示す。

表 1 料理の音声対話の精度に対する評価項目

番号	質問内容
1	わかりやすい
2	使いやすい
3	スムーズに料理ができると感じた
4	手間であると感じた
5	また使ってみたい

5.1.4 結果と考察

各レシピのアンケートの解答結果を平均して棒グラフに表したものを図31と図32に示す。

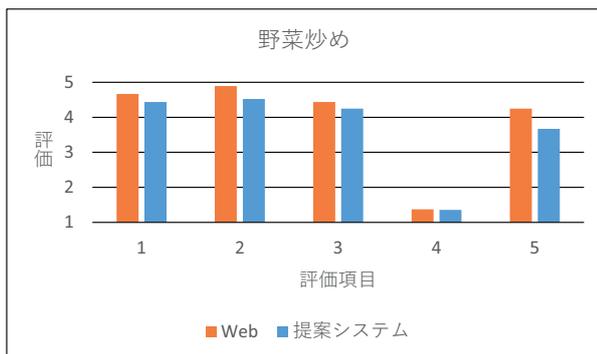


図 31 野菜炒めのアンケート結果

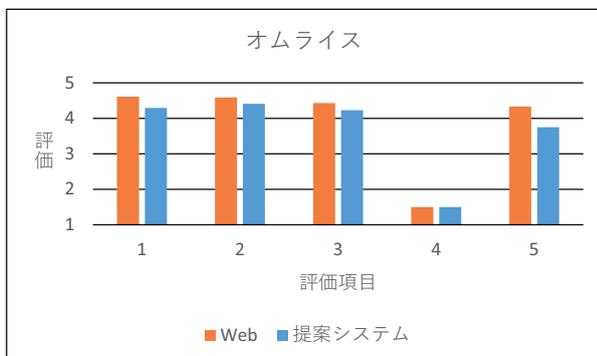


図 32 オムライスのアンケート結果

アンケートの結果を見ると、各レシピのどの評価項目についても Web ブラウザを使用して料理をする方が良い評

価を得ていることがわかる。また、実験後のコメントでは、「理解の助けになるというより、レシピを読んでいるだけのように感じた.」、「単純な工程なら音声で理解しやすいが、複雑な説明の部分は難しく感じた.」といったコメントが見られた。これは音声でレシピの手順を理解するよりも、画像や文を見るほうが理解しやすいからだと考えられる。提案システムではレシピを表示しながら料理の音声対話を行うことができるが、レシピを見る際には手を使用してスクロールする必要がある。そのため、この結果を踏まえて、レシピサイトの自動スクロール機能もしくは、MMDAgentの画面に音声対話中の手順の画像や説明を添付する機能が必要であると考えられる。また、MMDAgentの字幕機能や自動生成でのパネル表示の機能も必要であると考えられる。一方、実験後のコメントの中には「手が空いてなくても使えるのはかなり便利」といった好印象のコメントも見られた。この結果を踏まえて、提案システムを改良することで、料理をしやすくする目的を満たすことができると考えられる。

5.2 評価実験 2(編集システム)

5.2.1 評価実験の方法

被験者(本学学生8人)に、従来システムと提案システムを使用して指定したレシピの対話を再現する対話シナリオを作成してもらった。作成にかかった時間と実験後に5段階のアンケート評価を行った。順番に影響されないように、被験者をグループAとグループBに分け、グループAは先に従来システムを使用して対話シナリオを作成し、その後、提案システムを使用して作成した。反対に、グループBは先に提案システムを使用して対話シナリオを作成し、その後、従来システムを使用して作成した。

5.2.2 指定したレシピ

使用したレシピを図29と図30、図33、図34に示す。



図 33 ハンバーグ



図 34 パンケーキ

今回は、野菜炒めとオムライスより手順の1文が短いレシピであるハンバーグと材料の説明が短いレシピであるパンケーキを選択した。また、4つのレシピを選択することでレシピによる差の有無も検証した。

5.2.3 アンケート内容

従来システムを用いて対話シナリオを作成するする場合と、提案システムを使用する場合のそれぞれの場合についてアンケートに答えてもらう。アンケート項目として5項目を用意し、5段階評価を行う。評価基準は、「1（当てはまらない）、2（やや当てはまらない）、3（どちらとも言えない）、4（やや当てはまる）、5（当てはまる）」である。アンケートの内容を表2に示す。

表 2 編集システムに対する評価項目

番号	質問内容
1	簡単である
2	対話シナリオが編集しやすい
3	対話シナリオが見やすい
4	与えられているブロックで足りている
5	直感的にできた

5.2.4 結果と考察

各レシピの平均作成時間を表3に、アンケートの解答結果を平均して棒グラフに表したものを図35と図36、図37、図38に示す。

表 3 対話シナリオ作成時間

	従来システム	提案システム
野菜炒め	10分43秒	0分47秒
オムライス	12分01秒	1分11秒
ハンバーグ	11分22秒	0分53秒
パンケーキ	9分52秒	0分54秒

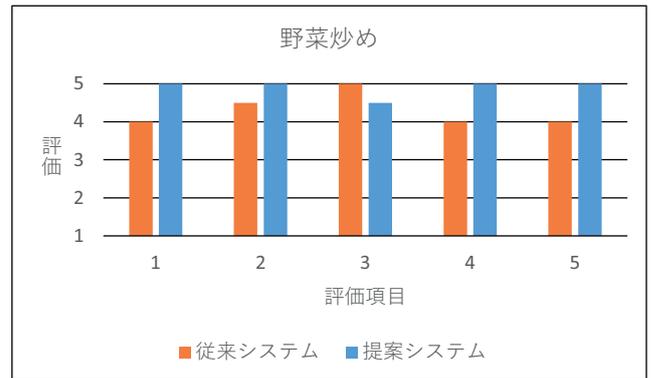


図 35 野菜炒めのアンケート結果

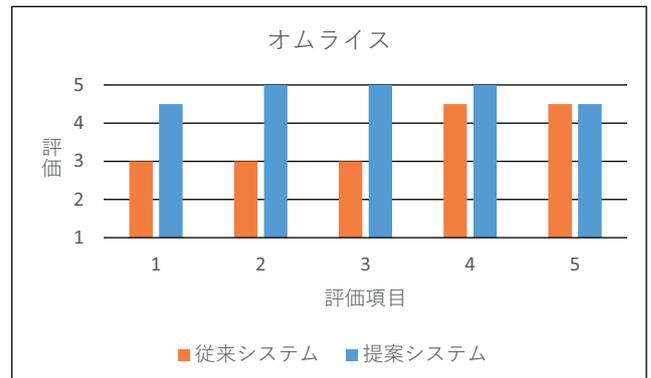


図 36 オムライスのアンケート結果

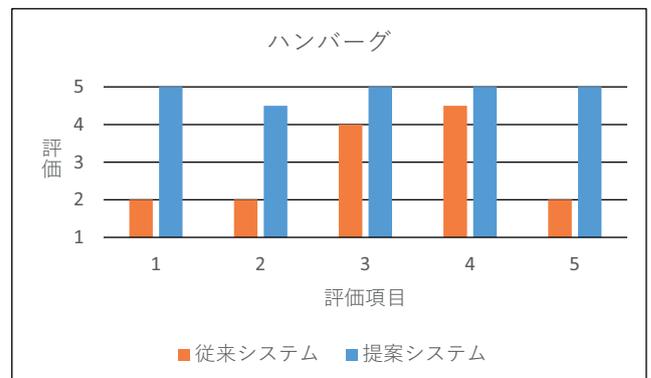


図 37 ハンバーグのアンケート結果

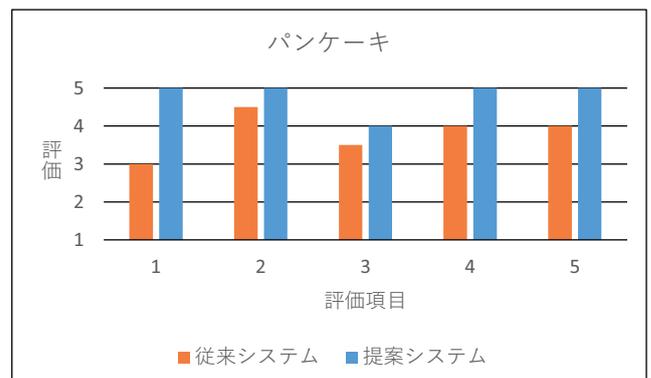


図 38 パンケーキのアンケート結果

各レシピの対話シナリオの作成の平均時間を比較すると、従来システムを使用する場合は9分から12分であったが、提案システムを使用する場合は47秒から1分で約10分の短縮となった。また、アンケートの結果をみると、野菜炒めの3番以外の評価項目については提案システムを使用する方が良い結果を得られていることがわかる。これは、手順が多いブロックが下に連なるため、このような結果になったと考えられる。ブロックを縮小や拡大できる説明をしていなかったため、Blockly 使用時の詳細な説明が必要であると考えられる。また、実験後のコメントでは提案システムに対して、「すぐできた」、「簡単だった、使いたい」といった好印象なコメントが見られた。これらの結果から、提案システムを使用することで、レシピの音声対話シナリオの作成や編集が簡単になったといえる。一方、実験後のコメントの中には「簡単だが、初心者の方がソースコードを使うのは難しいと思う。」といったコメントも見られた。この結果を踏まえて、レシピの URL をソースとして入力するのではなく、レシピ入力用のフォームを作成する必要があると考えられる。

6. おわりに

本研究では、従来手法を拡張し、Google Blockly を利用したレシピ共有サービスのための音声対話シナリオ編集システムを提案し、その実現法について述べた。また、提案システムに基づいたプロトタイプシステムを実装し、プロトタイプシステムを用いて評価実験を行った。

料理の音声対話の精度についての評価実験では、Web ブラウザを使用して料理をする手法と提案システムで料理をする手法を比較した。5段階評価のアンケートではすべての項目でサイトを見ながら料理をする手法が高い評価を得られた。このことから、レシピの自動スクロール機能を作成するなど提案システムを改良することで、スムーズに料理することができるシステムになることがわかった。また、編集システムについての評価実験では、既存のブロックを用いて料理の音声対話シナリオを作成する手法と提案システムで料理の音声対話シナリオを作成する手法を比較した。提案システムを使用することで、対話シナリオの作成時間が約10分の短縮となり、5段階評価のアンケートでは「簡単である」、「対話シナリオが編集しやすい」という項目において高評価が得られた。このことから、提案システムを使用することで短い時間で簡単に、料理の音声対話シナリオを作成や編集できることがわかった。

今後の課題として、料理の音声対話の精度についてはレシピの自動スクロール機能や MMDAgent の字幕機能、手順の画像や説明のパネル表示を自動生成で行う機能の作成が必要であると考えられる。また、編集システムについては、レシピの URL をソースとして入力するのは初心者にとってわかりにくいので、レシピ入力用のフォームを Blockly

に作成する必要があると考える。また、質問ブロックを挿入するのは手間であるため、質問ブロックの自動生成も必要であると考えられる。

参考文献

- [1] 李晃伸, 大浦圭一郎, 徳田恵一, 魅力ある音声インタラクションシステムを構築するためのオープンソースツールキット MMDAgent, 電子情報通信学会技術研究報告.NLC, 言語理解とコミュニケーション, Vol.111, No.364, pp.159-164, 2011.
- [2] Keitaro Wakabayashi, Daisuke Yamamoto, Naohisa Takahashi, A Voice Dialog Editor Based on Finite State Transducer Using Composite State for Tablet Devices, Computer and Information Science 2015, Studies in Computational Intelligence, vol.614, pp.125-139, 2016.
- [3] 西村良太, 山本大介, 打矢隆弘, 内匠逸. 音声対話エージェントのための web ブラウザを用いたシナリオエディタの開発. マルチメディア、分散協調とモバイルシンポジウム 2013 論文集, Vol.2013, pp.1796~1799, 2013.
- [4] 古市 瑞希, 山本 大介, 高橋 直久, ブロック型ビジュアルプログラミング機能を有する音声対話シナリオ編集システム, 情法処理学会論文誌デジタルコンテンツ, No. 8, Vol.2, pp.1~15, 2020.
- [5] 柴田翔平, 打矢隆弘, 内匠逸. スマートグラスと音声対話による料理支援, 情報処理学会第 78 回全国大会講演論文集, No. 2016, Vol.1, pp.99~100, 2016.
- [6] 佐野 芳樹, 神林 優河, 鈴木 優希, 渡部 智樹, 藤田 裕之, 関家 一雄, 一色 正男, ロボット対話と IoT 機器を活用したストレスフリーキッチン の提案, 第 80 回全国大会講演論文集, Vol.1, pp.117~118, 2018.
- [7] 杉本拓也, 伊與田光宏, 音声認識とジェスチャ機能を用いた料理者支援, 第 76 回全国大会講演論文集, Vol.1, pp.879~881, 2014.
- [8] MMDAgent version 1.7
<http://www.mmdagent.jp/> (参照 2022-04-25).