

# 特別研究報告

題目  
タイトル

指導教官  
教授 石黒 浩 先生

報告者  
名前

平成〇年〇月〇日

大阪大学 基礎工学部 システム科学科  
知能システム学コース

概 要

概要

# 目 次

# 第1章 序論

序論

## 第2章 本研究の背景と目的

### 2.1 哲学対話

#### 2.1.1 哲学対話の目的

#### 2.1.2 哲学対話に必要な支援

### 2.2 関連研究

### 2.3 本研究の目的

### 2.4 本稿の構成

## 第3章 議論支援システムの開発と実証実験1

本章では、哲学対話を支援することを目的としたロボットが介在する選択式議論システム（Robot Mediated Selection Based Discussion System: RMSBD）のプロトタイプの開発と評価実験について報告する。3.1 では設計指針を 2.3 で行った目標設定に対応させて整理し、3.2 で実装したシステムの構成を記す。??で評価実験について報告する。

### 3.1 システムの設計

### 3.2 システムの構成

実装した議論システム（以下、RMSBD-1 と呼称する）は、(a) 発話データベース、(b) 中央処理部、(c) ユーザインターフェース、(d) 発話ロボットからなる。以下に、それぞれの部分の詳細と全体の概略を示す。

#### (a) 発話データベース

##### 意見データベース

2.3 で述べたように、議論中にユーザに提示する選択肢は議題ごとに事前に用意した。発話候補は以下のような手続きで収集、編集を行った。

#### 1. Google Form を利用した意見の収集

オンラインで募集した研究協力者に、議題に関連する複数の意見に対する賛否と、その理由を答えさせた。以下に、論題「愛とは何か」の発話データベースを構築するために作成した質問を例示する。

これからいくつかの意見を提示させていただきます。それに賛成か、反対かを回答した後、その回答の理由を別の角度から二つ記述してください。

- 「お金は愛よりも大切である」という意見に対して

- － 上記の意見に賛成か反対かを教えて下さい。
- － それはなぜですか? 「～から」で終わる一文で答えてください。かならず「～から」もご自身で記入してください。
- － それはなぜですか? 異なる観点からもう一つ理由を答えてください。「～から」で終わる一文で答えてください。かならず「～から」もご自身で記入してください。
- 「愛は人の判断を誤らせる」という意見に対して
  - － 上記の意見に賛成か反対かを教えて下さい。
  - － それはなぜですか? 「～から」で終わる一文で答えてください。かならず「～から」もご自身で記入してください。
  - － それはなぜですか? 異なる観点からもう一つ理由を答えてください。「～から」で終わる一文で答えてください。かならず「～から」もご自身で記入してください。

## 2. 収集した意見からの論点抽出

上記の Google Form の回答のうち、それぞれの質問に対する賛否の理由の記述を整理し、論点の抽出を行った。論点の抽出にあたって、まずそれぞれの記述の前提となっていると考えられる信念をいくつかの類型に分類した。その後、分類をもとに論点を構成した。この作業はシステムの設計者である筆者が行った。以下に、この作業の進行過程の例を示す。

「お金は愛よりも大切である」という意見に賛成する理由として、以下のようない記述が得られた。

- お金があるのは手段でしかなく、愛は目的になりうるから。
- お金は二次報酬だから。

また、反対の理由として、以下のような記述が得られた

- 費用対効果の面から愛はある一定値に収束する一方、お金は単調増加で豊かになるから。

- (取引されるサービスやものに対する) 愛の度合いを相対的に示す数値  
がお金だから

以上のような記述の背景には、「愛とは〇〇のようなものである。それに対し、お金とは△△のようなものである」といった信念が存在すると考えられる。以上を踏まえ、「愛とお金はどう違うのか」という論点を形成した。

### 3. 論点の構造化と発話候補の割り当て

前述の要領で抽出された論点を階層的に整理した。以下の手順によって構造化を行った。

- (1). それぞれの論点に対し、最も関連すると思われる他の論点を選び、二論点間にリンクを張る。
- (2). リンクが多く、かつ程度抽象的だと考えられる論点を、最上位論点として、3個設定する。
- (3). 1で張ったリンクのうち、最上位論点以外の論点に対し、最上位論点から最短経路で辿るときに経由するもの以外を削除する

次に、構造化された論点に対し、前述の方法で収集された意見を割り振った。この際、同一内容と思われるもの、単体の発言としては理解が困難と考えられるものを省いて割り振った。また、論点に対する意見が十分な数ではない場合には、個別の論点に対する意見を数人の研究協力者に記述させ、発話候補を確保した。

また、意見収集時に、研究協力者が同じ質問に対して賛成の立場から記述した意見と、反対の立場から記述した意見は、対立する意見としてラベルした。この作業により、ある発話候補  $u$  に対し、その発話と矛盾する発話候補の集合が確定する。

最後に、別の論点からある論点に遷移させるときのための発話を構成した。

以上の作業と、得られ意見データベースをまとめると [図] のようになる。

これらの作業は、質問や論点に対する意見の記述を除き、システムの設計者である筆者が行った。以上の作業により、トピックに対する様々な意見を整理したデータベースを作成した。



## その他の発話

RMSBD-1 は、構造化された発話データベースの他に、あいづち、接続語、ファシリテーション発話のデータベースを持つ。

### ● あいづち

あいづちを用いることで、間接的に意見を主張することができる。

#### － 同意、共感を表すあいづち

「やっぱそうだよね」、「たしかに」、「わかるわかる」、「そのとおりだよ」、「わかる気がする」

#### － 非同意、反論をあらわすあいづち

「それは違うと思う」、「うーん」、「そうなのかな」、「えー」、「それには反対だな」

#### － 傾聴を表すあいづち

「なるほど」、「うんうん」

### ● 接続語

接続語を意見の前に発話することで、発言が持つニュアンスや前の発言に対する関係を明示的にすることができる。

#### － 順接

「じゃあ」、「だから」、「たしかに」、「そして」

#### － 逆説

「だとしても」、「でも」、「だけど」、「とはいえ」

#### － 並列

「なおかつ」、「それから」、「しかも」、「その上」、「さらに」

#### － 対置

「反対に」、「むしろ」、「逆に」、「一方で」

#### － 転換

「それじゃあ」、「ところで」、「話は変わるけど」、「そういえば」、「じゃあさ」

- ファシリテーション発話

ファシリテーション発話は、議論の進行を円滑化させる働きを持つと考えられる。

- 議論の流れに言及する発話

「ちょっと込み入ってきたし、違う視点から考えられないかな」、「なにか質問の仕方を変えてみるのはどうかな」

- 議論の現状を確認する発話

「難しくなってきたね」、「こうしてみるといろいろな意見があるね」

## (b) 中央処理部

中央処理部は以下の三つの処理を行う。

1. 論点遷移の管理と発話選択肢の抽出
2. ユーザインタフェースとの通信
3. ロボットとの通信

以下の説明のために、ユーザインタフェースとロボットを以下のように表示する。

議論に参加するユーザ数を  $N$  とする。接続されたユーザインタフェース端末を  $T_n (n = 1, 2, \dots, N)$  とする。接続されたロボットを  $R_n (n = 1, 2, \dots, N)$  とする。ここで、ユーザインタフェース端末とロボットの添え字の番号はユーザ番号を表し、それぞれの数字は同じユーザに対応しているものとする。

### 1. 論点遷移の管理と発話選択肢の抽出

各インターフェース端末上に表示させる発話候補は、以下のアルゴリズムによって決定される。

「(a) 発話データベース」に記した方法により構成した発話データベース上では、各論点  $A_x$  は、親論点  $A_{parent}$  と、子論点の集合  $A_{child}$  を持つ。ただし、末端の論点の場合には子論点の集合は空集合である。ある論点  $A_x$  に割り当てられた意見の集合を  $Op(A_x)$  とする。さらに、他の論点からある論点  $A_y$  に移動する際にまた、以下の記述では、ある論点上で発話が行われた回数をターン数  $t$  とする。

ユーザ  $p_i$  がそれまでに行った発話の集合を  $U(p_i)$  とする。また、各  $v \in U(p)$  に対し、 $v$  と矛盾する発話の集合を  $Op_{contradict}(v)$  とし、各ユーザの過去の発話のどれかに矛盾する発話の集合を  $Op_{contradict}(p_i)$  とする。また、議論の発展速度を  $pace$  とする。

現在、議論が論点  $A_i$  にあるとする。ターン数が  $t$  であるとする。

---

**Algorithm 1** 発話選択枝の抽出

---

```

1: if  $t < \min(pace, |Op(A_i)|)$  then
2:    $x \leftarrow pop(Op(A_i) \wedge \neg Op_{contradict}(p_i))$ 
3:   if  $aa$  then
4:      $aa$ 
5:   end if
6: else
7:    $aa$ 
8: end if

```

---

## 2. ユーザインタフェースとの通信

抽出された発話は TCP ソケットを通じてユーザインタフェースに送信される。また、ユーザインタフェース上で選択された発話を受信し、接続されたロボットに対して命令を送信する (以下の「3. ロボットとの通信」を参照)。

また、発話を受信した際には、その発話をロボットが行うのに要している時間中は全ての端末上での操作を禁止するよう、命令を送信する。

## 3. ロボットとの通信

ユーザインタフェースから発話の指令を受け取ると、ロボットに対しての操作命令を送信する。以下に、命令を生成する規則を説明する。

ロボットに対する操作命令の送信

$T_i$  から発話を受信したとする。この時、中央処理部は  $R_j (j \neq i)$  に対し、 $R_i$  が設置された方向を向くように命令を送る。また、 $R_j$  には発話を行うように命令を送り、同時に発話に合わせたジェスチャーを行わせるように命令を送る。

## (c) ユーザインターフェース

### 発話選択肢

ユーザインターフェースには、データベースから中央処理部で抽出された発話の選択肢が表示される。[図] に示すように、最上段には接続語の選択肢が3つ表示される。二段目には意見データベースから抽出された発話が表示される。三段目には、同意、共感を表すあいづちまたは傾聴を表すあいづちから抽出された発話が表示される。四段目には、非同意、反論を表すあいづちが表示される。

接続語を選択してから意見またはあいづちを選択することで、接続語を意見やあいづちの前に発話させることができる。接続語を選択せずに意見またはあいづちを発話することもできる。

### 選択肢更新ボタン

インターフェース上には「他の発話」というボタンを設置した。このボタン押すと、選択肢を更新する要求が中央処理部に送られる。その後、インターフェース上に表示される選択肢が別のものに更新される。

### 実装環境

ユーザインタフェースの実装には python を用いた。また、ユーザインタフェース用のプログラムを Microsoft Surface 上で実行し、タッチスクリーン上で選択肢を選ぶことができるようにした。上述のように (「(b) 中央処理部」を参照)、選択された発話は中央処理部を介して発話ロボットに送られる。

## (d) 発話ロボット

ユーザインタフェース上で選択された発話は、ロボットに読み上げられる。本システムでは、Vstone 社製の CommU<sup>1</sup>を使用した [図]。CommU は、全身 14 自由度を持ち、発話と同時に、表情や身振りの表出を行うことができる。発話を行う際には、胸部のスピーカから音声を流しつつ口部関節を振動させることで、実際に話しているような印象を与える。

---

<sup>1</sup><https://www.vstone.co.jp/products/commu/index.html>

本システムにおいて、CommU はユーザーインターフェースで選択された発話に連動し、以下の動作を行う。これらは、中央処理部からの操作命令に従って行われる（命令の送信については「(c) ユーザインタフェース」の項を参照）。

- 発話時にジェスチャーを行う。
- 発話しているロボットの方向を注視する。

### **3.2.1 システム動作のフローチャート**

## **3.3 実験**

### **3.3.1 検証項目**

### **3.3.2 実験結果**

## **3.4 考察**

## 第4章 議論支援システムの開発と実証実験2

### 4.1 システムの変更点

### 4.2 実験

#### 4.2.1 検証項目

#### 4.2.2 実験結果

### 4.3 考察

## 第5章 議論支援システムの開発と実証実験3

### 5.1 実験1、実験2で明らかになった問題点の整理

### 5.2 システムの設計

### 5.3 実験

#### 5.3.1 検証項目

#### 5.3.2 実験結果

### 5.4 考察

## 第6章 結論



# 謝辭

謝辭