

マルチモーダル情報に基づく見守りシステム開発への取り組み

飯島采永[†]

小林 一郎[‡]

[†]お茶の水女子大学大学院 人間文化創成科学研究科[‡]お茶の水女子大学基幹研究院

{iijima.sae, koba}@is.ocha.ac.jp

1 はじめに

近年、家庭用ロボットが多く普及されてきている。ロボットと共に生活していく上で、ロボットのコミュニケーション能力のさらなる充実が、今後、益々必要と考えられる。そこで、本研究では家庭用ロボットの身体性を利用し、生活を見守るためのシステムの構築を目的とし、その実現に向けて、マルチモーダル情報を用いた部分観測マルコフ決定過程 (POMDP) に基づいたロボットとの対話処理と、ロボットの観測に基づく1日の要約文やそれに基づく質問応答の作成に取り組む。

2 ロボットとのマルチモーダル対話

2.1 マルチモーダル情報の観測

ロボットはソフトバンクロボティクス社とアルデバランロボティクス社が共同開発した、感情認識ヒューマノイドロボット Pepper¹ を使用する。Pepper の様々なセンサからマルチモーダル情報を取得し、それに基づくコミュニケーションを実現する。具体的には、マイクから音声情報、RGB カメラから表情などの画像情報、タッチセンサから触覚情報、レーザーセンサやソナーセンサから距離情報を取得する。画像情報を用いた顔認識では、ユーザに対して、個体の識別、年齢の推定、笑顔度の判定、5種類の表情 { 無表情, 幸せ, 驚き, 怒り, 悲しみ } の識別を行うことができる。

2.2 POMDP

本研究では、実環境での観測情報の不確実性を考慮するため、部分観測マルコフ決定過程 (POMDP: Partially Observable Markov Decision Process) の枠組みを用いる。一般的に POMDP の観測状態は $\{S, A, T, O, Z, R, b_0\}$ で表される。 $s \in S$ はユーザ状態, $a \in A$ はシステムの行動を表す。また, T は行動 a によって状態 s が s' へと遷移する確率 (状

態遷移確率 $P(s'|s, a)$ の集合であり, $o \in O$ はユーザから観測される観測値を表す。 Z は行動 a によって状態が s' に遷移し, 観測値 o' が観測される確率 (観測値出力確率 $P(o'|s', a)$) の集合である。 $r(s, a) \in R$ は状態 s で行動 a を行った時の報酬を表す。

POMDP では、観測値 o から直接観測できない状態 s を確率分布として推測し、その分布を信念状態 $b(s)$ とする。初期信念状態を b_0 と表す。信念状態 $b(s)$ が既知のとき、状態遷移確率と観測値出力確率により、次の時刻の信念状態 $b'(s')$ は以下の漸化式で記述される。

$$b'(s') = k \cdot P(o'|s', a) \sum_s P(s'|s, a) b(s)$$

ここで係数 k は $\sum_s b'(s') = 1$ を満たす正規化項である。

2.3 マルチモーダル状態表現への拡張

ユーザとのインタラクションを想定して、下記に示す3つのユーザ状態 s^e, s^p, s^l が考えられる。

- 心理状態: s^e
喜怒哀楽のようなユーザの心理的な状態を示す。画像情報を用いた表情認識を用いて観測 o^e を取得する。
- 物理状態: s^p
ユーザがロボットからどれくらいの距離にいるのか、ロボットに触っているかいないかなどの物理的な状態を示す。観測 o^p はレーザーセンサやソナーセンサ、タッチセンサから取得する。
- 言語による情報交換: s^l
「おはよう」などの挨拶や「～してほしい」という要求のような、ユーザの発話による情報交換を示す。観測 o^l は音声情報から取得する。

この3つの状態に対応する観測をそれぞれ o^e, o^p, o^l とする。

マルチモーダル対話例を表1に示す。

¹<http://www.softbank.jp/robot/>

表 1: 対話例

話者	発話・行動	観測	$b(s)$
ユーザ Pepper	(遠くにいる) 近くにおいでよ	o^p (遠い)	
ユーザ Pepper	(近くにくる) 僕とお話しよう	o^p (近い)	
ユーザ Pepper	こんにちは こんにちは	o^l (こんにちは)	
ユーザ Pepper	(暗い顔) 疲れた顔をしてるね	o^e (悲しい)	

3 要約文の生成

次に, Pepper が観測した内容を時系列データとしてデータベースに記録し, 1 日分のデータに対する要約文の生成を行う.

3.1 Pepper による情報の取得

Pepper から観測した情報をデータベースに格納する. 今回は記録する情報として, { 日付, 時間, ユーザ名, 距離情報, 表情情報 } と設定する. ここで, ユーザ名とは, Pepper が対話, もしくは観測した一人の相手のことを指す. 距離情報は Pepper からの距離を記録し, 表情情報は, 5 種類の表情 { 無表情, 幸せ, 驚き, 怒り, 悲しみ } を数値で記録する. また, ユーザ名, 距離情報, 表情情報については観測状況によっては取得できないことがある. その場合, 取得できなかった情報は “-” として記録する. ユーザ名, 距離情報, 表情情報のすべてのデータが観測できなかった場合には観測がなかったとしてデータを記録しないこととした.

3.2 重要情報の抽出

あるデータにおいてひとつ前のデータと比較し, 変化のない場合にはそのデータを削除することにより, データベースの冗長性とデータサイズを削減する. データベースへ格納した時系列データに重要度を付与することで, 重要な情報を効率的に伝えることができるようになる. ここで, 今回のデータにおける重要度について考える. 本提案では以下の点に該当するデータを重要だと判断する.

- ユーザ名, 距離情報, 表情情報ができるだけ多く観測できたデータ

- 観測対象 (ユーザ) が変わったデータ
- ユーザの表情情報が一時刻前より大きく変動したデータ

これらの条件のいずれか, または全てをを満たすデータに重要度を付与する.

3.3 テンプレートによる文生成

重要度に対して適切な閾値を設定し, その閾値よりも値の高い文を選ぶことによって重要な情報を取り出し, その情報にテンプレートを適用することにより要約文を生成する. 今回は「“いつ”、“だれ”と会いました. その人は“どんな様子”でした。」というテンプレートに情報を入れて文を生成する. 「“どんな様子”だったか」については表情情報において最も数値の大きい値を使用し, その値と他の値との差分を計算し, 差分が大きい場合には「とても」, 小さい場合には「少し」という副詞をつける. 実際に Pepper の実機を動かして, 取得したデータについて文生成を行った. 表 2 に示したデータから表 3 に示す重要なデータを取り出し, そのデータに対して次の文を生成した. 「今日は 16 時 11 分に飯島さんに会いました. 少し驚いたような様子でした. また, 16 時 24 分にも飯島さんに会いました. 少し幸せそうでした。」

表 2: 取得データ抜粋

時間	ユーザ名	距離	表情 { 無表情, 幸せ, 驚き, 怒り, 悲しみ }
14:52	-	0.44	-
14:53	-	0.82	-
16:11	-	0.37	-
16:11	飯島	0.26	{0.08, 0.0, 0.5, 0.0, 0.42}
16:12	飯島	0.03	{0.03, 0.04, 0.46, 0.07, 0.4}
16:12	飯島	0.64	-
16:12	-	0.79	-
16:24	-	0.94	-
16:24	飯島	0.37	{0.02, 0.42, 0.38, 0.16, 0.02}
16:25	飯島	0.50	{0.0, 0.45, 0.4, 0.12, 0.03}
16:25	飯島	0.13	{0.0, 0.48, 0.35, 0.17, 0.0}

表 3: 重要データ抜粋

時間	ユーザ名	距離	表情 { 無表情, 幸せ, 驚き, 怒り, 悲しみ }
16:11	飯島	0.26	{0.08, 0.0, 0.5, 0.0, 0.42}
16:24	飯島	0.37	{0.02, 0.42, 0.38, 0.16, 0.02}

3.4 観測情報に対する質問応答

さらに、観測した情報に対してユーザが欲しい情報を要約して返すことについて考える。今回の観測データでは { 日付, 時間, ユーザ名, 距離情報, 表情情報 } を取得するので、例えば「今日の15時ごろに誰と会ったか」などの質問に答えられるようにする。このとき、要約文生成の時に使用した重要データでは、情報が欠けているデータには高い重要度を付与していないので、情報を見落とすことも考えられる。見守りをする際には「誰かはわからなかったけれど誰かが近くにいた」という情報も重要度は高くないが必要な情報となる可能性がある。そこで、観測データを時間で区切ることを考える。時間で区切り、その時間内の重要情報をだけを残すことで、データ量を削減しつつ情報の取りこぼしを減らす。時間で区切ったデータを使用して質問応答を行う。時間によって区切る方法としては、ある時刻に観測したデータの時間とその一時刻前に観測したデータの時間を比べて、その差が5分以上あった場合には場面が変わっていると考えて区切ることにする。例の「今日の15時ごろ誰と会ったか」という質問では、時間で区切った重要データにおける14:30から15:30までに観測したデータを探索し、返ってきた結果に対して文生成を行う。今回の場合に返ってくるデータは

{ 時間, ユーザ名, 距離情報, 表情情報 }
= { 14:52, “-”, 0.44, “-” }

なので、「今日の14時52分にお顔は見えませんが、近くに人はいました。」と応答する。

表 4: 取得データを時間で区切ったもの

時間	ユーザ名	距離	表情 { 無表情, 幸せ, 驚き, 怒り, 悲しみ }
14:52	-	0.44	-
14:53	-	0.82	-
16:11	-	0.37	-
16:11	飯島	0.26	{0.08, 0.0, 0.5, 0.0, 0.42}
16:12	飯島	0.03	{0.03, 0.04, 0.46, 0.07, 0.4}
16:12	飯島	0.64	-
16:12	-	0.79	-
16:24	-	0.94	-
16:24	飯島	0.37	{0.02, 0.42, 0.38, 0.16, 0.02}
16:25	飯島	0.50	{0.0, 0.45, 0.4, 0.12, 0.03}
16:25	飯島	0.13	{0.0, 0.48, 0.35, 0.17, 0.0}

表 5: 時間ごとに重要なデータ

時間	ユーザ名	距離	表情 { 無表情, 幸せ, 驚き, 怒り, 悲しみ }
14:52	-	0.44	-
16:11	飯島	0.26	{0.08, 0.0, 0.5, 0.0, 0.42}
16:24	飯島	0.37	{0.02, 0.42, 0.38, 0.16, 0.02}

4 まとめと今後の課題

本研究では、Pepperを対象にした見守りシステムの構築を目的とし、マルチモーダル対話を POMDP の枠組みに沿って実装を行った。また、対話状態を含まないマルチモーダル情報に基づいた要約文生成についても実装を行った。今後の課題として、対話状態についても要約文を生成することや、生成した文に対して評価を行うことが挙げられる。

参考文献

- [1] J. Pineau, G. Gordon, and S. Thrun “Point-based value iteration: An anytime algorithm for pomdps”, In Proceedings of the International Joint Conference on Artificial Intelligence, pp. 1025-1032, 2003.
- [2] B. Bonet “An e-optimal grid-based algorithm for partially observable Markov decision processes”, In Proc. of ICML, pp. 51-58, 2002.
- [3] Jason D. Williams, Steve Young “Partially observable Markov decision processes for spoken dialog systems”, Computer Speech and Language, Volume 21, Issue 2, pp. 393-422, 2007.
- [4] 木村元, Leslie Pack Kaelbling “部分観測マルコフ決定過程下での強化学習”, 人工知能学会誌, pp. 822-830, 1997.
- [5] 南泰浩 “部分観測マルコフ決定過程を用いたインタラクション制御入門”, <http://www.lai.kyutech.ac.jp/sig-slud/SLUD63-minami-POMDP-tutorial.pdf>, 2011.