

駐車記録を支援する車載器における対話アーキテクチャ

徳久雅人 木村周平

鳥取大学 工学部

tokuhisa@tottori-u.ac.jp

1. はじめに

本研究では、自動車旅行を支援する車載器の開発を進めている[1]. 車載器はユーザに音楽再生や観光ガイド[2]などのサービスを提供する. ユーザと同行することでユーザに適したサービスとしたい. ユーザ適応のためにユーザに関する情報が必要である. 本稿ではユーザの情報として駐車地点名および駐車期間中の事態(ユーザの体験や感想等)を扱う.

駐車地点名および事態は, ユーザから入力される. しかし, ユーザは乗車時に入力し忘れることがある. そこで, ユーザからの入力(ユーザ主導対話)を待つだけでなく, 車載器(システム)からユーザに問い掛けを行うこと(システム主導対話)を可能とする対話アーキテクチャの構築を本稿の目的とする.

本稿では, 欲求・感情を考慮した処理部で対話アーキテクチャを構成することを提案する. ユーザ主導対話のためには認識およびサービスの処理部を働かせ, またシステム主導対話のためには欲求に起因する対話管理を追加して働かせる. さらに, 対話行為の論理的な側面だけでなく感情の主観的な側面も含めた素性設計により, 対話管理モデルの機械学習を容易に実現する方法を示す.

2. 対話アーキテクチャ

車載器の対話アーキテクチャを, 岡田の心のモデル[3]をベースにした処理部で構成する(図1). 本章では, 処理部の動きを対話例に沿って概説する.

対話例1: ユーザ主導対話

1User (以下U): Wikipedia で検索してください

2System (以下S): 何を調べますか

3U: 鳥取砂丘です

(S: 鳥取砂丘のページを読み上げる)

ユーザの発話(1U)を認識部が解析する. 観光ガイドサービスの検索メソッドが起動する. クエリをユーザに質問する(2S). 3Uを受けて認識部は観光ガイドサービスに「鳥取砂丘」を添えてコールする. 観光ガイドサービスがガイドを始める[4].

対話例2: システム主導対話

(初期: 駐車地点名が未登録である.)

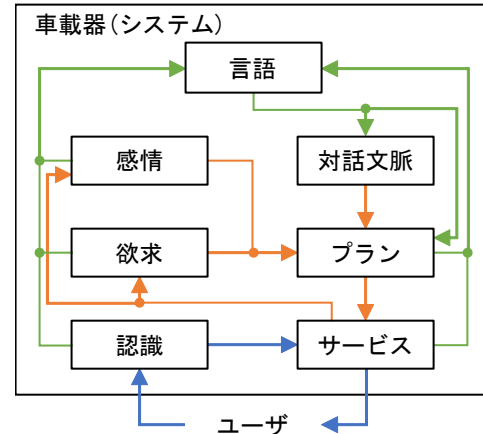


図1. 心のモデルに基づく対話処理部の構成

1S: 駐車地点名を教えてください

2U: ここは砂の美術館です

(S: 地点名を車載モニタに表示する)

初期状態は走行軌跡サービス (ServiceTraceDB) に問い合わせることで得られる. 欲求部がその確認結果(未登録)を受けてプラン部に反応を要請する. 反応として ServiceTraceDB に地名の推定値を要求する. 推定値がないので質問する(1S). 2Uを受けて, 地点名を ServiceTraceDB に登録する.

対話例3: 感情の関わり

1U: ここはエスマート湖山店です

(S: 駐車地点名を登録する.

事態が未登録である. 事態を推定する.)

2S: 食品を買いましたか

3U: はいそうです

(S: 事態を車載モニタに表示する)

ユーザ主導で駐車地点名が登録される(1U). 欲求部は事態が未登録であることを受けてプラン部に反応を要請する. その結果, ServiceTraceDB に事態の推定値を要求する. 過去の「エスマート湖山店」での記録から事態が推定できたので, 感情部が期待を生起させる. 期待できるので推定値を用いた確認を発話する(2S).

背景にある処理

認識部, サービス部, 欲求部, 感情部の処理結果は言語部に報告される. 言語部は対話文脈部やプラン部に報告を配信する. 対話文脈部は対話処理用に

報告を取捨選択しシステムの状態としてまとめ、プラン部に提供する。プラン部はシステムの振る舞いにタイミングを合わせてアクションを実行する。

3. システム主導と対話管理

本章では、情報収集のための車載器において、システムが対話を動機付ける部分、および、状況に応じて対話の進行を管理する部分について説明する。

3.1 欲求による動機付け

本稿では車載器に次の2つの欲求を設ける。

- 駐車地点名 (parking) を知る欲求
 - 駐車期間中の事態 (affairs) を知る欲求
- なお、欲求の生起については4章で述べる。

3.2 管理されるアクション

欲求を基準にアクションを定義する。「 X を知る欲求」に対応するアクションの一覧を付録A1に示す。「act_」はシステムが行うアクション、「res_」はユーザが行うアクションである。アクションのタイプとして、質問 (ask)、確認 (confirm)、伝達 (inform)、肯定や同意 (yes)、否定や拒否 (no)、不明 (unknown)、主張の取り込み (acknowledge)、および、話題終了 (close) がある。一般に対話行為と呼ばれる。話し手から聞き手に向けられるため、本稿では「外向きのアクション」と呼ぶ。さらに、「内向きのアクション」を導入する。「infer」は推論を行うアクションでありシステム内部での動きを指す。推論処理は負荷が高いため頻繁に実行してはいけない。発話のタイミングに合わせて実行することが適切であるため、管理されるアクションとする。

3.3 アクション選択の基準と素性

アクションの選択基準として以下を設ける。

- (a) 論理的基準
- (b) 主観的基準

3.3.1 論理的基準と素性

論理的基準は、(a1) 行為に必要な事柄、(a2) 行為により確定する事柄、および、(a3) 重複発話の回避のための事柄がある。

例えば、「駐車地点名の確認」の行為は「ここはエスマート湖山店ですか」という発話文に対応するのだが、確認内容として具体的な店名が得られていること、および、何度も確認しないことに注意して選択される。他に、「駐車地点名の肯定」(はい、そうです)の行為は、「仮定された地点名」があるときに意味をもつ行為であり、行為の結果、「真の地

点名」が得られることになる。

論理的基準を扱うために素性として以下の3つを導入する。

- $\text{var_temp_}X = 0 \text{ or } 1 : X$ の仮り値を有するか
 - $\text{var_real_}X = 0 \text{ or } 1 : X$ の真値を有するか
 - $\text{hst_}A_X = 0 \text{ or } 1 : X$ の行為 A を実行したか
- ここで X は欲求内容 {parking, affairs}, A はアクションタイプ {infer, ask, ..., close} である。

3.3.2 主観的基準と素性

発話を選択する理由が多様である。例えば、駐車地点名が推定されているとき、「エスマート湖山店ですか」とも「ここはどこですか」とも尋ねることができる。地点名を言うことが面倒だったり、珍しい名前なので無理に言ってみたりするのもかもしれない。細かな事柄を素性に入れることができれば良いが、車載器のサービスと釣り合う実装作業であるかを考えておきたい。本稿では、感情というラベルで細かな状況をまとめることとし、非 end-to-end 型コーパスベース機械学習において簡単に学習データを作成できることを狙う。

主観的基準は、(b1) 欲求、および、(b2) 感情である。欲求は、既に述べた2つを本稿では扱う。感情は、「ユーザから事態についての返事が得られる期待」を扱う。素性として以下の2つを導入する。

- $\text{desire_know_}X = 0 \text{ or } 1 : X$ を知る欲求が生起
- $\text{emotion_expectancy_response_affairs} = 0.0 \sim 1.0$: 上記の期待が生起

3.3.3 アクション知識の記述

対話管理での選択において、複数欲求の生起に対しての選択、アクション間の選択、対話継続の選択などが多様にあるように見えるかもしれない。これらを考慮してアクション知識を記述するのだろうか？本稿ではそうせず、1つのアクションの知識にはそのアクションのためだけの条件的な記述を行うこととする。他のアクションとの競合解消は、対話文脈で得られる全体的な状態における識別問題として扱う。

アクション知識には、プランニングに用いられる「前提」と「効果」の属性を参考にする。しかし、本稿は、ゴール状態を目指す効果やその前提を連鎖させるプランニングを行うものではない。

論理的基準のための素性を、前提と効果の位置に記述する。欲求の素性を、前提の位置に記述する。感情の素性は、必ずしもアクションの知識に記述するわけではない。

具体例を付録A3、A4に示す。前者は駐車地点名

を推定するアクションの知識である．内向きのアクションである (*:inward*)．前提 (*precondition*) には，駐車地点名を知る欲求の生起，仮りの駐車地点名が無いこと，真の駐車地点名が無いこと，駐車地点名の推論が未実行であることが記述されている．効果 (*effect*) には，推論が実行済みであること，仮りの駐車地点名が推論処理に応じて決まること (*exsit*) が記述されている．

3.4 論理と諸事情を考慮した学習

アクション選択を対話状態からの識別問題として扱う．まず，対話文脈として徐々に送られてくる素性 *feature* を蓄積する．次に，アクション選択の要請があると，素性をまとめて対話状態 *state* に変換し，状態に応じてアクションの予測を行う．

$$action^* = \underset{a \in Actions}{\operatorname{argmax}} Predict(a | state)$$

$$state = vector(feature, \dots)$$

学習用のコーパスには，初期設定として欲求や感情の素性を記入できる．対話の進行に合わせてシステムやユーザの発話に対応するアクションを記述できる．対話の途中で生起する感情や推定される感情などの処理結果も素性として追記できる．

コーパスからの学習データの自動生成を次の手順で行う (図 2)．まず，初期設定の素性に，最初のシステムのアクションの前提に記述されている素性を重ねる．得られた素性集から状態ベクトルを生成する．この状態ベクトルを入力状態とし，システムのアクションを出力目標とするペアを，学習データに加える．次に，システムの 2 番目のアクションのための状態ベクトルは，先ほど得た素性集合に最初のアクションの効果に記述されている素性を重ね，さらに 2 番目のアクションの前提の素性を重ねることで生成する．状態ベクトルと 2 番目のアクションのペアを学習データに加える．なお，ユーザのアクションや各種処理による素性を重ねられる．

3.5 プラン実行における同期と割り込み

プラン部の処理は図 3 の状態遷移に従う．システム全体の状態は言語部からの配信を通じて把握する．そのためアクションの実行結果が得られてから，その影響がシステム全体に伝わり終えるまでを考慮した同期が必要である．

しかし，感情や欲求に対応した判断はこうした同期にさらに割り込むものである．そこで，欲求・感情の状態は言語部の配信と異なる経路でプラン部に伝わるようにしている．これにより，生起や躊躇に敏感に反応してアクションの中止が可能となる．

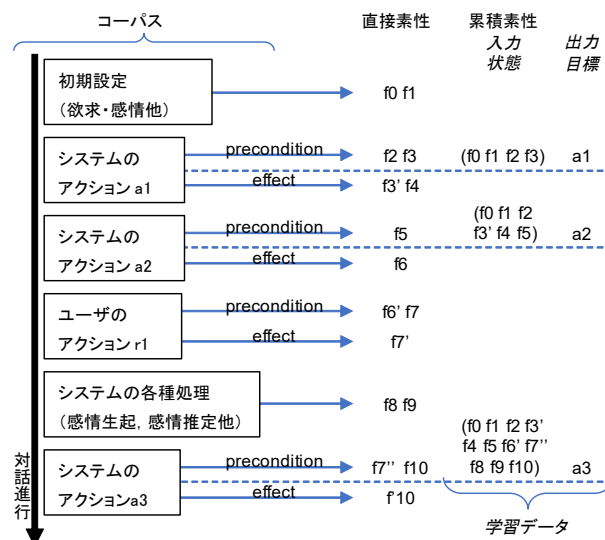


図 2. コーパスからの学習データの生成

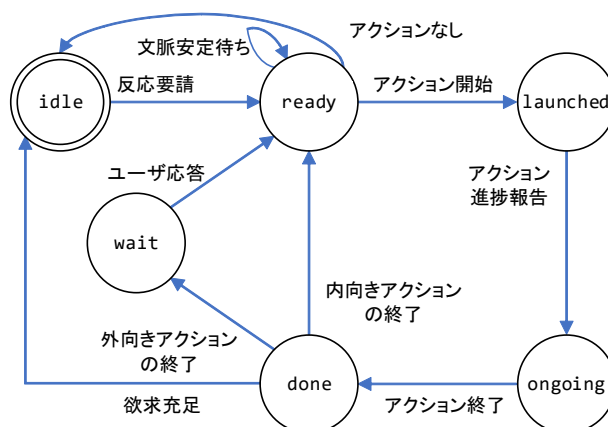


図 3 プラン部の状態遷移図

4. 対話に至る処理

2 章で処理の全体像を示し，3 章で対話処理の中心部として欲求の生起後からアクション選択までの部分を示した．本章では，欲求の生起前について処理部ごとに説明する．

4.1 サービス部

サービス部の影響で，感情や欲求が生起するので，まずはサービス部を説明する．走行軌跡サービス *ServiceTraceDB* は走行軌跡の保存と検索を行う．自動車のエンジンの始動から停止までの地点の時系列を 1 つの軌跡とする．地点は約 100m 毎に記録する．駐車地点には地点名および事態を記録する．同じ地点名を有する近隣地点の集合を場所として記録する．ユーザとともに訪れた施設などの場所が管理できる[5]．

本稿では以下の 4 つの機能を使用する．

地点名検索: 指定地点の地点名を返す.

事態検索: 指定地点に登録された事態を返す.

地点名推定: 指定座標を含む場所の地点名を返す.

事態推定: 指定座標を含む場所の事態を返す.

2 つの検索は、単に登録された値を返す. 駐車地点名を知る欲求, 事態を知る欲求の直接の生起要因となる. 地点名推定は, 座標に関するガウス分類器により指定座標が強く属する場所(地点のクラス)を識別し, その場所の名前を返す. 事態推定は同様に場所を識別し, その場所を構成する地点における事態集を返す.

事態推定においては, 未登録頻度, 高頻度語(一般名詞 N , サ変接続名詞 NS , 自立動詞 V), 高頻度語を含む単語 Trigram ($N+P+V$, $N+P+NS$. ただし P は助詞) を求める. 例えば, 「エスマート湖山店」の事態推定を行うと, 高頻度語が「買う」であり高頻度の Trigram が「食品を買う」となる.

4.2 感情部

システムの感情の生起を処理する. 処理の原理は生起要因を検出する処理を, 検出先のモジュールに組み込む形式である[6].

期待の感情「ユーザが事態を答える」の生起を EmotionExpectancyResponseAffairs が担当する. その生起は, 4.1 節で示した事態推定の内容に基づく方法である. Trigram が得られた場合を期待 1.0, Trigram は無いが高頻度語が得られた場合を期待 0.8, 未登録頻度の方が高いが訪問頻度が低い地点であれば期待 0.3, 未登録頻度が高く訪問頻度が高い地点であれば期待 0.0 とする. 期待の度合いは現在のところ試験的に与えた.

4.3 欲求部

システムの欲求の生起・躊躇・充足を処理する. 欲求は充足されていない状況が続くことで生起への度合いが高まるものとした. 時間周期で活性化するスレッドにより状況を検査し, 閾値を超えることで欲求の生起とする.

駐車地点名を知る欲求 DesireKnowParking および駐車期間での事態を知る欲求 DesireKnowAffairs を, 4.1 節の地点名検索および事態検索をそれぞれ用いて実装する.

自動車のエンジンを始動後, 車載器が起動する. ある程度の待ち時間の後に知る欲求が生起する. ユーザ主導による地点名・事態の入力が無いと判断する時間の設定は試験的に与えている.

ユーザの応答が事態入力を拒むものであれば, 知

る欲求は躊躇の状態となり, システム主導の対話が中止となる. 躊躇基準によりシステムの主張の強さという性格付けを行うことが可能である.

5. 動作確認

学習データの規模は, 素性 57 次元, アクション 41 次元, ペア数 52 件であった(本稿以外の素性・アクションを含む). 走行試験にて欲求生起による駐車地点名や事態の問い掛けを確認した. 感情の効果として, 車載器からの質問の抑制が見られた. 例えば, 自宅において, 事態がほぼ登録されていないため期待されず, 質問されなかった.

6. おわりに

本稿では, 車載器に駐車記録に登録するための対話処理のアーキテクチャを示した. 構築すべきものとしてアクション知識とコーパスがあるが, 知識はアクションの論理を注意するのみであり, アクション間や欲求間の競合はコーパスからの学習を通じて解決されるため, 実装しやすくてきた. 今後は欲求の種類を追加しスケジュールの聞き取りを行う.

謝辞

本研究は JSPS 科研費 JP19K12548 の助成を受けたものです.

参考文献

- [1] 徳久雅人, 木村周平: 旅行支援のための情報を提示する車載器の試作, 観光情報学会第 14 回研究発表会講演論文集, pp.5-8, 2016.
- [2] Tokuhisa, M., Ishihara, Y., Kimura, S., Oku, K.: Recommending Paragraphs of Wikipedia Pages as a Travel Guide, Proc. of IEEE 9th International Workshop on Computational Intelligence and Applications, pp.57-62, 2016.
- [3] 岡田直之: 語の概念の表現と蓄積, 電子情報通信学会, コロナ社, 1991.
- [4] 徳久雅人, 木村周平: 小型計算機におけるサービス指向発話文解析, 自然言語処理, 26 (3), pp.545-578, 2019.
- [5] 徳久雅人, 木村周平: 自動車旅行の走行軌跡と走行目的の収集支援 --- 地点名について, 観光情報学会第 20 回研究発表会講演論文集, pp.9-12, 2019.
- [6] 徳久雅人, 岡田直之: パターン理解的手法に基づく知能エージェントの情緒生起, 情報処理学会論文誌, 39 (8), pp.2440-2451, 1998.

付録

A1. 「 X を知る欲求」に対応するアクション

※ $X \in \{ \text{parking, affairs} \}$

act_はシステムの行為, res_はユーザの行為

act_infer_ X : X を推定 (内向きの行為)

act_confirm_ X : X を確認 (Yes/No 質問)

act_ask_ X : X を質問 (具体値を尋ねる)

act_ask_another_ X : 他の X を質問

act_acknowledge_ X : ユーザの発言を受理

act_close_ X : X の話題をやめる

res_yes_ X : 確認内容 X は正しい

res_no_ X : 確認内容 X は誤り

res_unknown_ X : 確認内容 X は不明

res_inform_ X : X の具体値を伝達

res_close_ X : X の話題をやめる

A2. 対話文脈を構成する素性

var_temp_ X =0 or 1

var_real_ X =0 or 1

hst_ A _ X =0 or 1

desire_ X =0 or 1

emotion_ E _ T =0.0 ~ 1.0

※ $A \in \{ \text{infer, confirm, ask, ask_another,} \\ \text{acknowledge, close} \}$

※ $E \in \{ \text{expectancy} \}$

※ $T \in \{ \text{response_affairs} \}$

A3. アクション知識の例 1

```
acFrame(  
  # name  
    "act_infer_parking",  
  # desire  
    "desireKnowParking",  
  # executor  
    "serviceTraceDB", "actInferParking",  
  # direction  
    :inward,  
  # precondition  
    [ "desire_know_parking=1",  
      "var_temp_parking=0",  
      "var_real_parking=0",  
      "hst_infer_parking=0" ],  
  # effect  
    [ "hst_infer_parking=1",  
      "var_temp_parking=exist(infer_parking)"])
```

A4. アクション知識の例 2

```
acFrame(  
  # name  
    "act_confirm_parking",  
  # desire  
    "desireKnowParking",  
  # executor  
    "serviceTraceDB", "actConfirmParking",  
  # direction  
    :outward,  
  # precondition  
    [ "desire_know_parking=1",  
      "var_temp_parking=1",  
      "var_real_parking=0",  
      "hst_confirm_parking=0" ],  
  # effect  
    [ "hst_confirm_parking=1" ])
```

A5. アクション知識の例 3 (抜粋)

```
acFrame(  
  "res_yes_parking", - - -,  
  # precondition (effectで改訂される場合, 不要)  
  # effect  
    [ "var_temp_parking=1",  
      "var_real_parking=1"])
```

A6. コーパスの例

```
dialogue step actor utterance action  
1 1 S "" desire_know_parking=1  
2 S "" act_infer_parking  
3 S "ここは<loc>エスマート湖山店</loc>ですか"  
    act_confirm_parking  
4 U "はいそうです" res_yes_parking  
5 S "" desire_know_affairs=1  
6 S "" act_infer_affairs  
7 S "何をしましたか" act_ask_affairs  
2 1 S "" desire_know_parking=1  
2 S "" act_infer_parking  
3 S "ここはどこですか" act_ask_parking  
- - - 略 - - -  
14 1 S "" desire_know_affairs=1  
2 S "" act_infer_affairs  
3 S "" emotion_expectancy_response_affairs=1.0  
4 S "" act_confirm_affairs  
15 1 S "" desire_know_affairs=1  
2 S "" act_infer_affairs  
3 S "" emotion_expectancy_response_affairs=0.3  
4 S "" act_ask_affairs
```

※ 発話文は参考表現 (無記入可).

※ 対話文脈の部分を与えることが可能.