

VoiceXMLを用いた柔軟な対話生成と話題管理モデル

呉 浩東、下川 浩

A Flexible Dialog Generation and Topic Management Model Using VoiceXML

GO Kotoh, SHIMOKAWA Yutaka

Abstract

In recent years, more and more researchers are attending to design spoken dialogue systems to produce friendly user-machine environments.

We in this paper develop a cooperative approach to dialogue systems which uses VoiceXML as a tool for dialogue generation and topic management. To achieve this goal, we discuss how to dispose the problems of robust language understanding, dialogue management, and cooperative answer generation.

1. はじめに

対話システムは人間と機械またはソフトウェアの間で、自然言語によるコミュニケーションを行うものである。対話システムは、ユーザによる自然言語の発話入力を理解し、それに適切に応答していく。そのようなやりとりを繰り返して、ユーザとともに何らかの目標を達成していく。

ユーザは自然言語を用いてさまざまな情報システムを簡単に操作できる情報環境の実現を期待している。自然言語や音声を用いた対話システムの実現に向けて、昔から多くのシステムが提案された。近年、対話システムについての研究開発が盛んに行われている。その理由はサービス費用の削減と人工知能環境の提供である。

しかしながら、あらゆる場面で使用できる対話システムの開発は現段階では実現できない。従って、音声・言語処理に関するさまざまな要素技術の高度化が不可欠である。その実現には、ユーザの意図同定や意味解析、感情推定や適切な発話文生成など、多くの問題が関わる。いままで、対話システムに必要な

要素技術はそれぞれ独立して研究されている。

本論文で、われわれはタスク向きの対話システムにおける対話生成と話題管理のモデルを提案し、その実現に関わる要素技術を論究する。

2. VoiceXMLによる知識表現

VoiceXML (Voice Extensible Markup Language, VXML) は、人間とコンピュータの間の音声のやり取りを記述するもので、W3Cの標準XMLフォーマットの一種である。視覚的アプリケーションをHTMLが記述するのと同じような方法で、音声アプリケーションの開発や配布を可能にする。HTML文書がWebブラウザで解釈されるように、VoiceXML文書はボイスブラウザで解釈される。

ここでは、新幹線の予約を例とする。

```
<form id="新幹線予約情報">
  <field name="source">
    <grammar src="shinkansen.grxml"/>
    <prompt>乗車駅は？</prompt>
  </field>
  <field name="destination">
    <grammar src="shinkansen.grxml">
    <prompt>到着駅は？</prompt>
  </field>
</form>
```

3. 話題管理

われわれは、対話システムにおける対話の主導権はシステム主導、ユーザ主導と混合主導の三種類に対応し、お互いにシフトできる体制を実現するものである。

例1：下記の対話でSはシステムを指す。Uはユーザを指す。

S1：こちらは新幹線の案内システムです。ご質問をどうぞ。

U1：新幹線で大阪に行きたい。

S2：大阪に行く新幹線はひかりとのぞみがあります。どちらのご利用ですか？

U 2 : のぞみを利用したいと思います。料金はいくらですか？

S 3 : 指定席のご利用ですか？

U 3 : そうです。

S 4 : 東京から京都までの料金は14,450円です。

U 4 : 9時20分ぐらい発車の列車は何番線ですか？

S 5 : 9時23分発車の「のぞみ211号」の乗り場は11番線です。

対話システムにおいては、対象とするタスクの違いやユーザグループの違いによって実現すべき対話の主導権が異なる。

3.1 システム主導

システム主導スタイルは、基本的にシステムからの情報要求に対してユーザが応答するというケースを繰り返すことが多い。

メリット：システム主導に対しては、ユーザの回答を限定することができる。

デメリット：システムの質問に対しては、一般に対話のタスクの達成に要する時間が長くなる傾向がある。また、ユーザが意図した結果が得られるかわからない場合もある。

3.2 ユーザ主導

ユーザ主導の場合では、ユーザがシステムに対して質問し、システムから情報を引き出す。

例 2 :

U 1 : 新宿の近くに大きな本屋はありませんか？

S 1 : 紀伊國屋書店はいかがですか？

U 2 : わかりました。どのように行ったらいいですか？

S 2 : J R 新宿駅東口の新宿通りのそばです。

ユーザ主導の場合、ユーザの入力発話が適切であれば、対話の効率がいい。一方、ユーザの発話が複雑になる傾向がある。また、システムを使い慣れないユーザには、システムにどのように話せばよいのかはわかりにくい。言い換えると、発話の修復がしばしば必要とされる。

3.3 混合主導

この方式では、システムとユーザの間の発話の修復が可能である。

例3：沖縄的那覇空港においての会話を想定する。

パターン1：

U1：海洋水族館に行きたいのですが、バス停はどこですか？

S1：3番乗り場です。

パターン2：

S1：3番乗り場です。あと5分で発車します。

パターン3：

S1：すみません、今日水族館は休みです。

パターン4：

S1：今日水族館は休みですが、琉球村はいかがですか？

U1：じゃ、琉球村に行きます。ありがとう。

対話においては、機械とユーザの間で相互に相手のことを理解しているほうが情報の伝達に必要な場合が多い。このような場合、システムはユーザの「モデル」を持ち、言外の情報を補いながら話を理解していく。以下の4つの公準（mixim）に従って話すことになる。

1. 質の公準：告げることは根拠のある事実であり、実際に思っていることである。
2. 量の公準：過不足のない情報を伝える。
3. 関係の公準：話し手と聞き手は相互に関連した事柄を話す。
4. 様態の公準：明確に簡潔に順序立てて話す。

言語の伝達においては、互いに協調し、これらを満たすべきことを表現したこの原理は協調原理といわれる。

発話には文の意味解釈の基になる文法構造と、言語表現レベルでの伝達意図解釈レベルとの双方に関連性がある。一般に、文法的な関連性を首尾一貫性と呼び、言語表現レベルの関連性を結束性と呼ぶ。

例4：

S1：〇〇旅行社です。お客様のご用は何でしょうか？

U1：ANAの航空券を取りたいのですが。

S2：ご利用はいつでしょうか？

- U 2 : 7月5日午前中です。
 S 3 : どちらからどちらまで行かれますか？
 U 3 : 羽田から新千歳までお願いします。
 S 4 : 10時30分発の航空券でよろしいでしょうか？
 U 4 : けっこうですよ。

話者	対話行為	発話例
A	はい/いいえ質問	あなたは今大学に行きますか？
A	発話権放棄	あなたは…
B	肯定回答	ええ。
B	陳述	来年3月に卒業します。
A	称賛	いいですね。
B	相槌	ええ。

表 1. 対話行為の例

- S 1 : ○○観光案内システムです。ご質問があったらどうぞ。
 U 1 : 上高地に行きたいのですが。
 S 2 : 日帰りですか？それとも一泊ですか？
 U 2 : 日帰りにします。お金はどれぐらいかかりますか。
 S 3 : 新宿発の高速バスをご利用になれば9千円前後です。
 U 3 : わかりました。

4. 対話管理

対話管理 (Dialogue Management, DM) は対話システムの中心部である。その役割はユーザの表現を入力して、対話システム内部状態の更新などを行い、ユーザに応答するための表現を決定するものである。対話管理方法としては、状態遷移のネットワークに基づくものである。具体的な役割は以下の通りである。

- ユーザ入力の確認
- ユーザにさらなる入力を促すこと
- ユーザに適切な情報の提供

システム主導対話には、必須のスロットに最新情報を埋めていくものにする。話題が想定範囲を超える場合に適切な話題に切り替える。ユーザとシステムの間の混合主導対話の場合は、オートマトンで対話の状態を表現する。

VoiceXMLを用いた対話管理の例は以下の通りです。

```
<field name="confirm_exit" slot="YES_NO">
  <grammar type="application/srgs+xml"
    src="../grammar/boolean.grxml"/>
  <prompt>
    対話を終了しますか？
  </prompt>
  <filled>
    <if cond="confirm_exit=='true'">
      ありがとうございます。
      <exit/>
    <else/>
      それでは、質問をどうぞ。
      <clear namelist="q1 q2 send confirm_exit"/>
    </if>
  </filled>
  <catch event="noinput nomatch">
    <throw event="exit"/>
  </catch>
</field>
```

対話生成の例：

```
<form id="welcome">
  <block>
    <prompt cond="(new Date()).getDay() == 0" bargein="false">
      すみません。〇〇病院は火曜日が休みです。
    </prompt>
    <prompt cond="(new Date()).getDay() != 0" bargein="false">
      もしもし、お電話ありがとうございます。
    </prompt>
  </block>
</form>
```

5. 意図の理解と発話生成

対話システムにおいて、ユーザの発話の意図を把握することが非常に重要である。発話行為のタイプから意図を読み取ることが大事である。

発話行為の例：

1. 未知情報要求

“TCP/IPというのは何ですか？”

2. 真偽情報確認

“そうですか？”

“次の地下鉄の発車時刻は9時15分ですか？”

3. 依頼

“清水寺への道を教えてください。”

対話をうまく成立させるには、信念、願望、意図の順に基づき次の処理を行う。

- 知覚
- 計画立案
- 意思決定
- 行動

ユーザの心理を理解するには、場所の理解が不可欠である。例えば、

U：今日は暑いですね？

S：そうですね。

より

(S：冷房を入れますか？

S：冷たいビールはいかがですか？)

などの応答がふさわしいだろう。

6. まとめ

本論文では、対話システムがユーザの意図と発話行為によって、システムが柔軟に応答を生成することができる方法を提案した。さらに、対話の主導権を対話の進み具合によってシフトできるようにした。

今後の課題としては、心理相談などの応用面で実効性を確かめながら、知識ベース、ユーザ記憶モデル、心内辞書などを構築する。さらに、コーパスやWeb情報を利用した用語間の関連性の自動取得 [Patrick 2006] などを用いて、

より広範囲で適用する対話システムの開発に努める。

参考文献：

1. Patrick, P., Macro, P.: Espresso: leveraging generic patterns for automatically harvesting semantic relations. *Proc. ACL*, pp.134-120, 2006.
2. 島津 明、他：「話し言葉対話の計算モデル」電子情報通信学会、2014.
3. 河原達也・荒木雅弘：「音声対話システム」オーム社、2006.
4. 中川聖一 編著：「音声言語処理と自然言語処理」コロナ社、2013.
5. 奥村 学 監修：「対話システム」コロナ社、2015.
6. Jokinen, K., McTer, M., Spoken Dialogue Systems, Morgan&Claypool Publishers, 2010.
7. Bonus D., Rudnicky: The Ravenclaw dialog management framework: Architecture and systems, *Computer Speech and Language*, 23(3), pp.332-361, 2009.
8. Cassell, J., et.al, Conversation as a system framework: designing embodied conversational agents, *Embodied Conversational Agents*. MIT press, 2000.
9. 金森克俊、大和田勇人：知的対話システムの形式的モデルとその実装、*The 27th Annual Conference of the Japanese Society for artificial Intelligence*, 2013.
10. W3C:Voice Extensible Markup Language (VoiceXML) version 2.0, W3C Recommendation, 2004.
11. Negi S., et.al, Automatically Extracting Dialog Models from Conversation Transcripts, *The 9th IEEE International Conerence on Data Mining*, 2009.