Owl-based Frame Descriptions for Spoken Dialog Systems

Masahiro Araki

Kyoto Institute of Technology
Department of Electronics and Information Science
Matsugasaki Sakyo-ku Kyoto 606-8585 Japan
araki@dj.kit.ac.jp

Abstract

In this paper, we propose a new frame description method using OWL (Web Ontology Language) for our frame-driven dialogue system. By using a standard knowledge representation, OWL, as a frame description language, the task domain portability of the dialogue system increases. In addition, by using standard language for semantic web, the RDF annotated pages can be used for knowledge base for the dialogue system.

Introduction

"Tell me what wines I should buy to serve with each course of the following menu. And, by the way, I don't like Sauterne."

In [1], such a query is regarded as an ultimate target which an intelligent agent can answer using semantic web technology. However, there is another difficulty for speech recognition and natural language processing in understanding such a long, spontaneous utterance. Considering current status of these speech and language technologies, a system initiative dialogue is one solution for that problem (Figure 1).

S1: What do you want to drink?
U1: *Wine.*
S2: What type of course do you choose?
U2: *Sea food course.*
S3: O.K. How about Sauterne, white?
U3: *I don't like Sauterne.*
S4: O.K. How about Tanba, white?

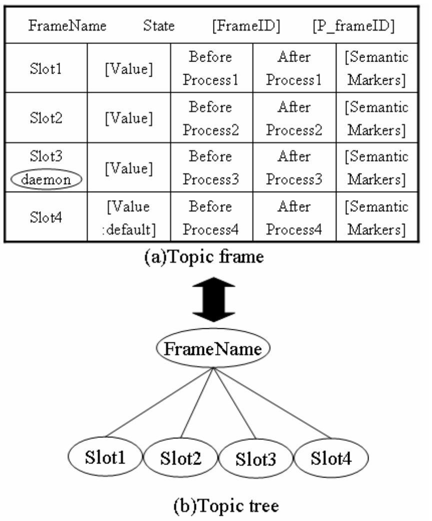
Figure.1 Sample dialogue

In order to manage such a system initiative dialogue, we proposed a frame-driven dialogue controller for multiple task domains [2]. The controller is applicable for such spoken interactions between a human and an intelligent agent. However, because it used a specific frame description embedded in Prolog program, it was difficult for web system developer to write such Prolog style frame descriptions.

In this paper, we propose a new frame description method using OWL (Web Ontology Language) for our frame-driven dialogue system. By using a standard knowledge representation, OWL, as a frame description language, the task domain portability of the dialogue system increases. In addition, by using standard language for semantic web, the RDF annotated pages can be used for knowledge base for the dialogue system.

Frame-driven Dialogue Controller

In our dialogue system, the knowledge of the task structure is represented by a frame. The frame has a status which shows the current topic and the dialogue history. Also, it has slots which consist of a value, a before process, an after process, and semantic markers. Because the value of the slot can be another frame, this structure can represent has-a relation. In addition, this frame representation can be viewed as a tree structure which represents a static task structure (Figure 2).



(a)Topic frame

(b)Topic tree

Figure.2 Topic frame and Topic tree

The dialogue proceeds as follows.

1. Top frame of the task is selected and expanded as a root of dynamic topic tree which represents dialogue history.

2. Top slot of the frame is selected and runs before process of this slot. The before process is typically starts the VoiceXML [3] interpretation which outputs the prompt (e.g. utterance S1 in Figure 1).
3. VoiceXML gets the answer and submit the value to the frame server process. The server fills the value to the slot and runs an after process. Typical after process is to move to the next slot or to search a database.
4. If the value of the slot is a frame, process 1 to 3 are applied recursively.

The system architecture is shown in Figure 3.

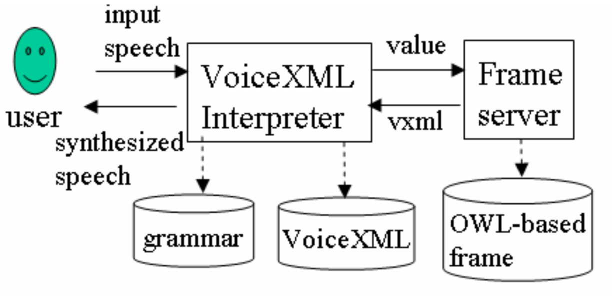


Figure.3 Dialogue system architecture

Frame Description in OWL

Originally, OWL is an ontology language for semantic Web. OWL can define a class, a property, and their instances. In class definition, we can use subClassOf and oneOf elements. Therefore we can express AND-OR tree by OWL which is equivalent to the frame representation explained in section 2.
For example, top level frame which handles the dialogue in Figure 1 is as follows (Figure 4).

```
<owl:Class rdf:ID=”ChoiceDrink”>
  <owl:oneOf rdf:parseType="Collection"
          rdf:ID=”Drink” vxml:before=”ask1.vxml”
          vxml:after=”#Food”>
    <Drink rdf:about="#Wine"/>
    <Drink rdf:about="#Beer"/>
    <Drink rdf:about="#Sake"/>
  </owl:oneOf>
  <owl:oneOf rdf:parseType="Collection" rdf:ID=”Food”
          vxml:before=”ask2.vxml”
          vxml:after=”SelectWine”>
   …
  </owl:oneOf>
</owl:Class>
```

Figure.4 Topic frame

At the beginning of dialogue, the top slot of the top frame is selected. The before process of the first slot is specified by the element of *vxml:before*. This element specifies the VoiceXML file named ask1.vxml (Figure 5).

```
<?xml version="1.0" encoding="Shift_JIS"?>
  <vxml version="1.0"  >
    <form>
      <field name=”Drink”>
        <grammar src=”Drink.gram”/>
        <prompt> What do you want to drink?</prompt>
        <filled>
          <submit next=”FrameServer”/>
        </filled>
      </field>
    </form>
  </vxml>
```

Figure.5 VoiceXML file (ask1.vxml)

This VoiceXML is interpreted in VoiceXML interpreter. The system opens the dialogue by prompting “*What do you want to drink?*” and waits user’s input expecting the vocabulary given by the grammar file Drink.gram. After listing the user’s input “*Wine.*”, this value is passed to the frame server which makes the instance of this frame. Such instances of frames consist of dynamic topic tree which expresses dialogue history. The value “*Wine*” is assigned into the first slot and after process of this slot begins.
In this example, the after process specifies next slot “*Food*” and proceeds the same way. Once the value of the food slot is given by the user, the system searches database or semantic Web under the condition of given values.

Conclusion

In this paper, we proposed OWL-based frame description for the spoken dialogue control and example dialogue driven by the proposed framework. By using OWL, we can expect future extensions listed below.

1. Vocabulary of the grammar file can be learned (expanded) using ontology described by OWL.
2. This dialogue component can seamlessly combined with planning component of the intelligent agent which is listed up at the use case of the ontology.
3. Multilingual spoken dialogue system can be easily implemented in this framework.

Acknowledgement

This work has been supported by the NEDO Industrial Technology Research Grant Program (No. 00A18004b)

References

Smith, M.K., McGuinness, D., Volz, R., and Welty, C. 2000. *Web Ontology Language (OWL) Guide Version 1.0* http://www.w3.org/TR/owl-guide/

Niimi, Y., Oku, T., Nishimoto, T., and Araki, M. 2001. *A rule based approach to extraction of topic and dialog acts in a spoken dialog system*, Proc. of EUROSPEECH2001, vol.3, pp.2185-2188.

VoiceXML Forum 2000. *Voice eXtensible Markup Language (VoiceXML™) version 1.0,* http://www.w3.org/TR/voicexml/.