

Language Technology

<http://cs.lth.se/edan20/>
Chapter 17: Dialogue

Pierre Nugues

Pierre.Nugues@cs.lth.se
http://cs.lth.se/pierre_nugues/

October 15, 2020



Dialogue

Interacting persons:

- Information can be missing
- Some words or constructions can be ambiguous,
- Errors in speech recognition.

Coreferences are central in a dialogue context.



Automata



Dialogue Pairs

First member	Preferred second member	Dispreferred second member
Offer, Invitation	Acceptance	Refusal
Request	Compliance	Refusal
Assessment	Agreement	Disagreement
Question	Expected answer	Unexpected answer, no answer
Blame	Denial	Admission



More Elaborate Pairs

- *initiative interventions*, which open an exchange (*I*)
- *reaction interventions*, which are answers to initiatives (*R*)
- *evaluation interventions*, which assess exchanges and possibly close them (*E*)

Utt. no.	Turns	Utterances
1	S:	<i>Which type of information do you want: loans, deposits, opening hours?</i>
2	U:	<i>Loans</i>
3	S:	<i>We are happy to give you information on loans</i>



Pairs with Closed Questions

Utt. no.	Turns	Utterances	Tags
1	S:	<i>Which type of information do you want: loans, deposits, opening hours?</i>	I_1
2	U:	<i>Deposits</i>	R_1
3	S:	<i>Are you interested in deposits?</i>	I_1^2
4	U:	<i>Yes</i>	R_1^2

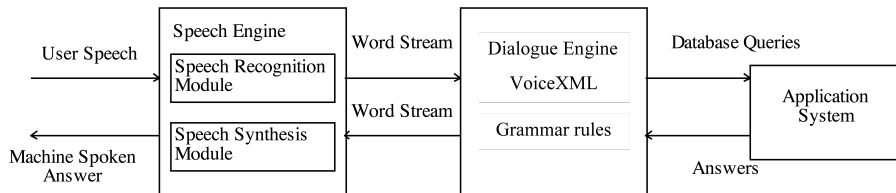


VoiceXML: A Language for Simple Dialogues

VoiceXML is a programming language to describe simple dialogues
It can process touch-tones, isolated words, and phrases with the help of a grammar

It uses external speech recognition and synthesis modules

It is frequently used in speech server applications



A VoiceXML Example

```
<?xml version="1.0" encoding="UTF-8"?>
<vxml xmlns="http://www.w3.org/2001/vxml" version="2.1"
  xml:base="http://localhost:8080/demo2/">
<form>
  <field name="process" type="bool">
    <prompt bargein="false">You have chosen the welding
      process. Do you want to continue?</prompt>
    <option>yes</option>
    <option>no</option>
    <filled>
      <prompt>You said <value expr="process"/></prompt>
    </filled>
  </field>
```



A VoiceXML Example

```
<field name="calibration" type="calibration">
  <prompt bargein="false">Calibration of work piece...
  </prompt>
  <option>no</option>
  <option>manual</option>
  <option>automatic</option>
  <filled>
    <prompt>You said <value expr="calibration"/></prompt>
  </filled>
</field>

...

</form>
</vxml>
```



Bühler's Sprechakte

- A representation (*Darstellung*) of objects and the state of affairs that is being described;
- An expression (*Ausdruck*) materializing the psychological state of mind of the speaker – the sender of the message;
- An appeal (*Appell*) corresponding to an effect on the hearer – the receiver of the message.



Speech Acts

Another classification due to Austin:

- *Locutionary* – saying something: syntactic structure, formal semantics content,
- *Illocutionary* – a conversational act: to inform, to suggest, to answer, to ask, ...
- *Perlocutionary* effects: frighten, to worry, to convince, to persuade, ...



Speech Acts Classes

- Assertives, such as stating, asserting, denying, informing;
- Directives, such as requesting, asking, urging, commanding, ordering;
- Commissives, such as promising, committing, threatening, consenting, refusing, offering;
- Declaratives, such as declaring the war, resigning, appointing, confirming, excommunicating. Declarative speech acts change states of affairs.
- Expressives, that are related to emotions or feelings such as apologizing, thanking, protesting, boasting, complimenting.



Links Between Syntax and Speech Acts

Classical grammar recognizes certain links between locutionary and illocutionary content:

Classical speech acts	Syntactic forms
Assertions, statements	Affirmatives or declaratives
Orders, commands	Imperatives
Questions	Interrogatives



Problem with Syntax

Syntactical form is sometimes misleading:

Can you open the door? Question?

Have a good day! Order?

A syntactical classification is too coarse to reflect the many needs of interaction analysis.



Searle's Conditions

For each act, four conditions. For example: assert

Conditions	Values
Propositional content	Any proposition P
Preparatory	<ol style="list-style-type: none"> 1 Speaker has evidence (reasons, etc.) for the truth of P 2 It is not obvious to both Speaker and Hearer that Hearer knows (does not need to be reminded of, etc.) P
Sincerity	Speaker believes P
Essential	Counts as an undertaking to the effect that P represents an actual state of affairs



Searle's Conditions: Request, Order, Command

Conditions	Values
Propositional content Preparatory	Future act <i>A</i> of Hearer <ul style="list-style-type: none">1 Hearer is able to do <i>A</i>. Speaker believes Hearer is able to do <i>A</i>2 It is not obvious to both Speaker and Hearer that Hearer will do <i>A</i> in the normal course of events of his own accord3 (For <i>order</i> and <i>command</i>) Speaker must be in a position of authority over Hearer
Sincerity Essential	Speaker wants Hearer to do <i>A</i> Counts as an attempt to get Hearer to do <i>A</i>



Helen of Troy

Speech acts are not new. Gorgias (487-380 BC) wrote:

The effect of speech upon the condition of the soul is comparable to the power of drugs over the nature of bodies. For just as different drugs dispel different secretions from the body, and some bring an end to disease and others to life, so also in the case of speeches, some distress, others delight, some cause fear, others make the hearers bold, and some drug and bewitch the soul with a kind of evil persuasion.

Encomium of Helen (Trans. RK Sprague)



Speech Acts: Examples (Gazdar and Mellish, 1989)

Turns	Utterances	Illocutionary acts
A	I really think the automobile needs servicing	Statement
B	But we had done it recently	Challenge
A	No, not for two years. . .	Challenge Interruption
A	Incidentally did you hear that gas prices are about to double?	Concession



EVAR (Mast et al., 1994)

EVAR structures the dialogue process in a sequence of phases:

- A greeting by the system,
- A request from the user,
- A clarification consisting of a confirmation of data requested by the user and possible requests for details and specifications,
- An answer from the system, and finally
- A closing.



EVAR Dialogue

Turns	Dialogue	Phases
S:	<i>Guten Tag. Hier ist die automatische Intercity-Auskunft. Welche Information benötigen Sie?</i> 'Hello. This is the Automatic Travel Information System. What information do you need?'	Greeting
U:	<i>Ich möchte morgen nach Hamburg fahren.</i> 'Tomorrow I want to go to Hamburg.'	Request
S:	<i>Sie möchten morgen nach Hamburg fahren?</i> 'Tomorrow you want to go to Hamburg?'	Confirmation
U:	<i>Ja morgen.</i> 'Yes tomorrow.'	
S:	<i>Von wo möchten Sie abfahren?</i> 'Where do you want to start?'	Request for details
U:	<i>München</i> 'Munich'	



EVAR Dialogue

S: *Morgen von München nach Hamburg*
'Tomorrow from Munich to Hamburg'

U: *Ja*
'Yes'

S: *Wann möchten Sie fahren?*
'When do you want to leave?'

Request for
specification

U: *Morgen früh*
'Early in the morning'

S: *Morgen früh von München nach Hamburg*
'Tomorrow morning from Munich to Hamburg'

U: *Ja*
'Yes'



EVAR Dialogue

S: *Ich habe mehere Verbindungen gefunden. Nach Hamburg HBF, fahren Sie ab München um 7 Uhr. Ankunft in Hamburg HBF um 15 Uhr, ...* Answer

'I found connections. From Munich departure 7, Arrival Hamburg 15, ...'

U: *Vielen Dank*
'Thank you'

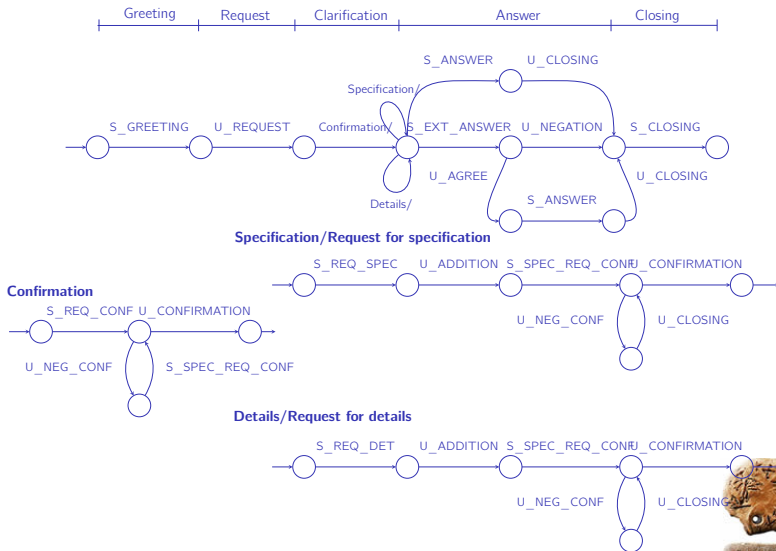
S: *Möchten Sie weitere Auskünfte?* Closing
'Do you want further information?'

U: *Nein danke*
'No thank you'

S: *Auf wiederhören.*
'Have a nice trip.'



EVAR's Automata (Mast et al., 1994)



France Telecom CNET's Dialogue Acts (I)

Acts	S/U	Descriptions
Initiatives		
request(P)	S	Open question or request for the value of P
yn_question(P, Val)	S	Is value of P Val? Answer should be <i>yes</i> or <i>no</i>
altern_question(P)	S	Alternative question: <i>Vanilla or strawberry?</i>
repeat(P)	S/U	Repetition request
inform(P)	S/U	Inform of P
recap(P)	S	Recapitulation of solved problems



France Telecom CNET's Dialogue Acts (I)

Acts	S/U	Descriptions
Reactions		
answer(P, Val)	U	Gives a value Val on the request of P
select(P, Val)	U	Gives a value Val on an alternative question on P
accept(P, Val)	U	Accept or confirm the value Val of P
reject(P, Val)	U	Reject the value Val of P
Evaluations		
impl_valid(P, Val)	S	Implicit validation of confirmation of the value Val of P
correct(P, Val)	U	Gives a new value Val to P



A Dialogue Example

Turns	Utterances
S1	<i>Bonjour puis-je vous aider?</i> 'Hello, can I help you?'
U1	<i>Je veux une réservation de Londres à Paris</i> 'I want a reservation from London to Paris'
S2	<i>Londres Paris à quelle date voulez-vous voyager?</i> 'London Paris which date?'
U2	<i>Je veux voyager le 20 juin</i> 'I want to travel June 20'
S3	<i>Le 20 juin à quelle heure?</i> 'June 20 which time?'
U3	<i>À 17 heures 15</i> '5.15 pm'
S4	<i>17 heures 15?</i> '5.15 pm?'



A Dialogue Example

Turns	Utterances
S5	<i>Le vol BA-123 part à 17 heures 15 et arrive à Charles-de-Gaulle à 18 heures 39, est-ce que cela vous convient?</i> 'Flight BA-123 departure 5.15 pm and arrival 6.39 pm at Charles-de-Gaulle airport. Is that OK for you?'
U5	<i>Oui</i> 'Yes'
S6	<i>Quel est votre nom?</i> 'What is your name?'
U6	<i>Luc Martin</i>



A Dialogue Example

Turns	Utterances
S7	<i>Luc Martin une réservation pour Luc Martin sur le vol BA-123 arrivant à Charles-de-Gaulle à 18 heures 39</i> 'Luc Martin a reservation for Luc Martin on flight BA-123 arriving at Charles-de-Gaulle at 6.39 pm'
U7	<i>Oui</i> 'Yes'
S8	<i>Votre réservation a été confirmée au revoir</i> 'You reservation has been confirmed goodbye'
U8	<i>Au revoir</i> 'Goodbye'



Dialogue Decomposition

Exch.	Interventions	Recursive int.	Turns
E1	Ii(s, [request])		S1
	Ir(u, [answer])		U1
	Ie(s, [impl_valid])		S2a
E2	Ii(s, [request])		S2b
	Ir(u, [answer])		U2
	Ie(s, [impl_valid])		S3a
E3	Ii(s, [request])		S3b
	Ir(u, [answer])		U3
E3e		Ie(s, [impl_valid])	S4
		Ir(u, [accept])	U4



Dialogue Decomposition

Exch.	Interventions	Recursive int.	Turns
E4	Ii(s, [recap, yn_question])		S5a S5b
	Ir(u, [accept])		U5
E5	Ii(s, [request])		S6
	Ir(u, [answer])		U6
	Ie(s, [impl_valid])		S7a
E6	Ii(s, [recap])		S7b
	Ir(u, [accept])		U7
	Ie(s, [impl_valid])		S8



Speech Acts Recognition

They are based on:

- Cue words or phrases linked to specific speech acts
- Syntactic and semantic forms of the utterance
- Expectations to apply constraints on possible speech acts.

These are based on transitions from a previous state to the current state of the dialogue: When the system asks a question, it expects an answer, a rejection or a failure, and it can discard other acts.

- Task modeling and goal satisfaction.

It restrains possible user acts and parameter values according to the progress point where the user is in the dialogue.

- Recognition uses either machine-learning techniques to classify the speech acts or logical constraints.



Examples

Syntactic features

Interrogative sentence

yes, right, all right, OK

no, not at all

Declarative sentence

sorry, pardon, can you repeat

not X but Y, that's not X it's Y in fact.

Candidate speech acts

`yn_question,`

`altern_question, request`

`accept, impl_valid`

`reject`

`inform, impl_valid`

`repeat`

`correct`



Corpus Collection

Often uses the wizard-of-oz technique: dialogues between users and a fake robot:

- Gives the vocabulary
- Gives the basic structure of a dialogue: a specification phase followed by a negotiation phase

Considerable variation amongst the customers.

