Language Processing with Perl and Prolog Chapter 17: Dialogue

Pierre Nugues

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



Pierre Nugues

Language Processing with Perl and Prolog

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 (1)
- reaction interventions, which are answers to initiatives (R)
- *evaluation interventions*, which assess exchanges and possibly close them (*E*)

Utt. no.	Turns	Utterances	
1	S:	Which type of information do you want: loans, de- posits, opening hours?	
2	U:	Loans	
3	S:	We are happy to give you information on loans	
		Anna and Inna ang Processing with Pert and Pholog Inna Inna	

Pairs with Closed Questions

Utt.	Turns	Utterances	Tags
no.			
1	S:	Which type of information do you want: loans, deposits, opening hours?	l ₁
2	U:	Deposits	R_1
3 4	S: U:	Are you interested in deposits? Yes	$\begin{bmatrix} I_1^2 \\ R_1^2 \end{bmatrix}$



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"</pre>
  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></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

Arrisk Augus Language Processing with Perl and Prolog Series Institute The Company of the Company Institute

13 / 43

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	
	 Speaker has evidence (reasons, etc.) for the truth of P
	It is not obvious to both Speaker and Hearer that Hearer knows (does not need to be reminded of, etc.) P
Sincerity	Speaker believes <i>P</i>
Essential	Counts as an undertaking to the effect that resents an actual state of affairs
	·····································

Searle's Conditions: Request, Order, Command

Conditions	Values
Propositional content	Future act A of Hearer
Preparatory	
	 Hearer is able to do A. Speaker believes Hearer is able to do A
	It is not obvious to both Speaker and Hearer that Hearer will do A in the normal course of events of his own accord
	 (For order and command) Speaker must be in a position of authority over Hearer
Sincerity	Speaker wants Hearer to do <i>A</i>
Essential	Counts as an attempt to get Hearer to do / Priandfolo
	 < □ > < ⊡ > < ⊡ > < ⊡ > < □ > < □

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)

Language Technology

Speech Acts: Examples (Gazdar and Mellish, 1989)

Turns	Utterances	Illocutionary acts
А	I really think the automobile needs servicing	Statement
В	But we had done it recently	Challenge
А	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:	Guten Tag. Hier ist die automatische Interci	ty- Greeting
	Auskunft. Welche Information benötigen Sie?	
	'Hello. This is the Automatic Travel Informatic	on
	System. What information do you need?'	
U:	Ich möchte morgen nach Hamburg fahren.	Request
	'Tomorrow I want to go to Hamburg.'	
S:	Sie möchten morgen nach Hamburg fahren?	Confirmation
	'Tomorrow you want to go to Hamburg?'	
U:	Ja morgen.	
	'Yes tomorrow.'	
S:	Von wo möchten Sie abfahren?	Request for de-
	'Where do you want to start?'	tails Language
U:	München	Processing with Perl and Prolog
	'Munich'	
Pierre Nu	ugues Language Processing with Perl and Prolog	20 / 43

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?'
- U: Morgen früh 'Early in the morning'
- S: Morgen früh von München nach Hamburg 'Tomorrow morning from Munich to Hamburg'
- U: *Ja*
 - 'Yes'

Request for specification

EVAR Dialogue

- S: Ich habe mehere Verbindungen gefunden. Nach Answer Hamburg HBF, fahren Sie ab München um 7 Uhr. Ankunft in Hamburg HBF um 15 Uhr, ...
 'I found connections. From Munich departure 7, Arrival Hamburg 15, ...'
- U: Vielen Dank 'Thank you'
- S: Möchten Sie weitere Auskünfte? 'Do you want further information?'
- U: *Nein danke* 'No thank you'
- S: *Auf wiederhören.* 'Have a nice trip.'

Closing



22 / 43

Language Technology

Chapter 17: Dialogue

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
<pre>yn_question(P, Val)</pre>	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 straw-</i> <i>berry</i> ?
repeat(P)	S/U	Repetition request
<pre>inform(P)</pre>	S/U	Inform of P
recap(P)	S	Recapitulation of solved problems
		Language Projection with

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
<pre>select(P, Val)</pre>	U	Gives a value Val on an alternative ques-
		tion on P
<pre>accept(P, Val)</pre>	U	Accept or confirm the value Val of P
reject(P, Val)	U	Reject the value Val of P
Evaluations		
<pre>impl_valid(P, Val)</pre>	S	Implicit validation of confirmation of the
		value Val of P
<pre>correct(P, Val)</pre>	U	Gives a new value Val to P

Language Processing with Perl and Prolog

A Dialogue Example

-

Utterances
Bonjour puis-je vous aider?
'Hello, can I help you?'
Je veux une réservation de Londres à Paris
'I want a reservation from London to Paris'
Londres Paris à quelle date voulez-vous voyager?
'London Paris which date?'
Je veux voyager le 20 juin
'I want to travel June 20'
Le 20 juin à quelle heure?
'June 20 which time?'
À 17 heures 15
'5.15 pm'
17 heures 15?
'5.15 pm?'
Oui (1) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2

A Dialogue Example

Turns	Utterances
S5	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?
	'Flight BA-123 departure 5.15 pm and arrival
	6.39 pm at Charles-de-Gaulle airport. Is that OK
	for you?'
U5	Oui
	'Yes'
S6	Quel est votre nom?
	'What is your name?'
U6	Luc Martin

Language Processing with Perl and Prolog

A Dialogue Example

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

< 一型

Language Processing with Perl and Prolog

Dialogue Decomposition

Exch.	Interventions	Recursive int.	Turns
	<pre>Ii(s, [request])</pre>		S1
E1	<pre>Ir(u, [answer])</pre>		U1
	<pre>Ie(s, [impl_valid]</pre>])	S2a
	<pre>Ii(s, [request])</pre>		S2b
E2	<pre>Ir(u, [answer])</pre>		U2
	<pre>Ie(s, [impl_valid]</pre>])	S3a
	<pre>Ii(s, [request])</pre>		S3b
E3	<pre>Ir(u, [answer])</pre>		U3
E3e		<pre>Ie(s, [impl_valid])</pre>	S4
		<pre>Ir(u, [accept])</pre>	U4
			Language
			Processing Peri and P Special Information
		< □ > < □ > < □ > < □ >	

Dialogue Decomposition

Exch.	Interventions	Recursive int.	Turns
E4	<pre>Ii(s, [recap, yn_question])</pre>		S5a S5b
	<pre>Ir(u, [accept])</pre>		U5
	<pre>Ii(s, [request])</pre>		S6
E5	<pre>Ir(u, [answer])</pre>		U6
	<pre>Ie(s, [impl_valid])</pre>		S7a
E6	<pre>Ii(s, [recap])</pre>		S7b
	Ir(u, [accept])		U7
	<pre>Ie(s, [impl_valid])</pre>		S 8



Speech Acts Recognition

The 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 logical constraints or statistical tagging with POS tagging



Examples

Syntactic features	Candidate speech acts
Interrogative sentence	yn_question,
	altern_question, request
yes, right, all right, OK	accept, impl_valid
no, not at all	reject
Declarative sentence	<pre>inform, impl_valid</pre>
sorry, pardon, can you repeat	repeat
not X but Y, that's not X it's Y in fact.	correct



Beliefs and Intentions

Dialogue can be modeled in terms of agents with capacities:

- wants(A, X), which means that agent A wants to do X,
- can_do(A, X), which means that agent A can do X,
- believes(A, X), which means that agent A believes X,

• knows(A, X), which means that agent A knows X, and acts: informs(A, B, P) (A informs B of P) Preconditions and postconditions:

- Preconditions: knows(A, P), wants(A, inform(A, B, P))
- Postconditions: believes(B, P)



An Operational Dialogue System: The SJ Train Information System

A service that answers questions on train times and fares in Sweden Let the customers order tickets Accessible by telephone: 0046 771-75-75-75 Based on a previous work done at Telia research Paper reference: Johan Boye, Mats Wirén, Manny Rayner, Ian Lewin, David Carter, and Ralph Becket, "Language-Processing Strategies and Mixed-Initiative Dialogues", *IJCAI-99 Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, July 1999



Corpus Collection

The authors started the development with a corpus collection Used the wizard-of-oz technique: ~130 dialogues and 50 subjects:

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

Considerable variation amongst the customers.

Architecture



Pierre Nugues

Dialogue Example

Hej jag beställer en flygbiljett den artonde i sjätte tisdag från Stockholm till Sundsvall.

recognized as

vad hej jag beställer jag vill jag den artonde i sjätte i jag mmm då Stockholm till Sundsvall.



Formal Representation

The system uses a "flat utterance description": a predicate whose argument is a list of parameter/value pairs

Four possible predicates corresponding to the customers' requests:

Act	Description
yn	Are there objects with property P?
wh	Find X with property P
wh_agg	Find the maximal/minimal X with property P
yn_agg	Does the maximal/minimal X with property P also have
	property P'



Examples

The utterance I want to arrive in Stockholm before 6 pm is interpreted as Find flights arriving Stockholm before 6 pm. It is translated as: wh(X, [slot(trip, trip_id, X), slot(trip, trip_mode, plane), slot(trip, to_city, Stockholm), slot(trip, arr_time, T), exec(before(T, 1800))])

Examples

- The utterance Is that a direct flight?
- is represented by:

```
yn([slot(trip, trip_mode, plane),
    slot(trip, stops, 0),
    slot(trip, trip_id, X),
    ref(X, det(def, sing))])
```

Examples of the two remaining acts:

wh_agg: I want the first flight to Stockholm? Which is the cheapest ticket? yn_agg: Is that the first flight?



Dual system:

- A robust parser a partial parser identifies phrases and keywords (DCG). The phrases are used to fill in the slots.
- The core language engine from SRI directly outputs FUDs using compositional rules
- The robust parser is overall more efficient than the core language engine. See discussion in the paper.

41 / 43

Dialogue Acts

An example of dialogue between a user (U) and the system (S).

	Utterance	Act
U	I want to go from Gothenburg to	user:constraint
	Stockholm on Friday	
S	At what time do you want to leave?	system:ask-for-constraint
U	In the morning	user:constraint
S	There is a train at 5:30 am arriving at	system:suggestion
	9:45 am	
U	Is that a direct train?	user:ask-for-info
S	Yes	<pre>system:answer-with-info</pre>
U	Is there a later train?	user:ask-for-suggestion
S	There is a train at 6:06 arriving at 9:15	system:suggestion
U	Fine, I'll take that one	user:accept
		I FOR A DIRUCTION

Dialogue Act Identification

In total, 12 different dialogue acts identified using the dialogue state and rules:

- The existence of suitable contexts. For example, an utterance cannot be classified as a user:accept unless the system has proposed some train(s) and or flight(s) that the user can accept
- The difference between the propositional contents of the utterance and that of the context. For instance, if these two are inconsistent, the utterance cannot be classified as a user:accept; if they are consistent, it is unlikely that the utterance should be classified as a user:ask-for-suggestion.
- The presence of keywords in the utterance. For example, if the utterance contains "accept words" like *yes*, *ok*, etc., the user:ac score is increased