



Goal oriented dialog systems

Машинное обучение в диалоговых системах

16 апреля 2019

Cheskova E - cheskidova@phystech.edu

Links

- Dan Jurafsky book, Draft chapters in progress, [Sep 23, 2018]
[Advanced dialog systems](#)
- Dan Jurafsky & James Martin, Stanford [2017]
[Conversational Agents overview](#)
- Matthew Henderson, Google [2015],
[Machine learning for dialog state tracking overview](#)

Plan

- Chatbots
 - Types
 - Pros & cons
- Goal-based dialog agents
 - Finite state dialog manager
 - Frame based dialog agents
 - Simple frame based system
 - Dialog state architectures
 - Dialog acts
 - Dialog structure elements
 - Dialog state
 - Dialog policy
 - NLG
 - NLU
 - Evaluation

Two classes of systems

1. Chatbots
2. (Goal-based) Dialog agents
 - SIRI, interfaces to cars, robots, booking flights or restaurants

Chatbots

- Pattern-action rules
 - Eliza
 - Parry
- Simple decision tree
 - [Woebot](#) 
- Ranking human chats
 - Microsoft Xiaoice
 - Microsoft Tay
 - [Cleverbot](#)

Eliza: Weizenbaum (1966)

Men are all alike.

IN WHAT WAY

They're always bugging us about something or other.

CAN YOU THINK OF A SPECIFIC EXAMPLE

Well, my boyfriend made me come here.

YOUR BOYFRIEND MADE YOU COME HERE

He says I'm depressed much of the time.

I AM SORRY TO HEAR YOU ARE DEPRESSED

...

WHO ELSE IN YOUR FAMILY TAKES CARE OF YOU?

My father

YOUR FATHER

Rogerian psychologist ELIZA

Chatbots: pro and con

Pro

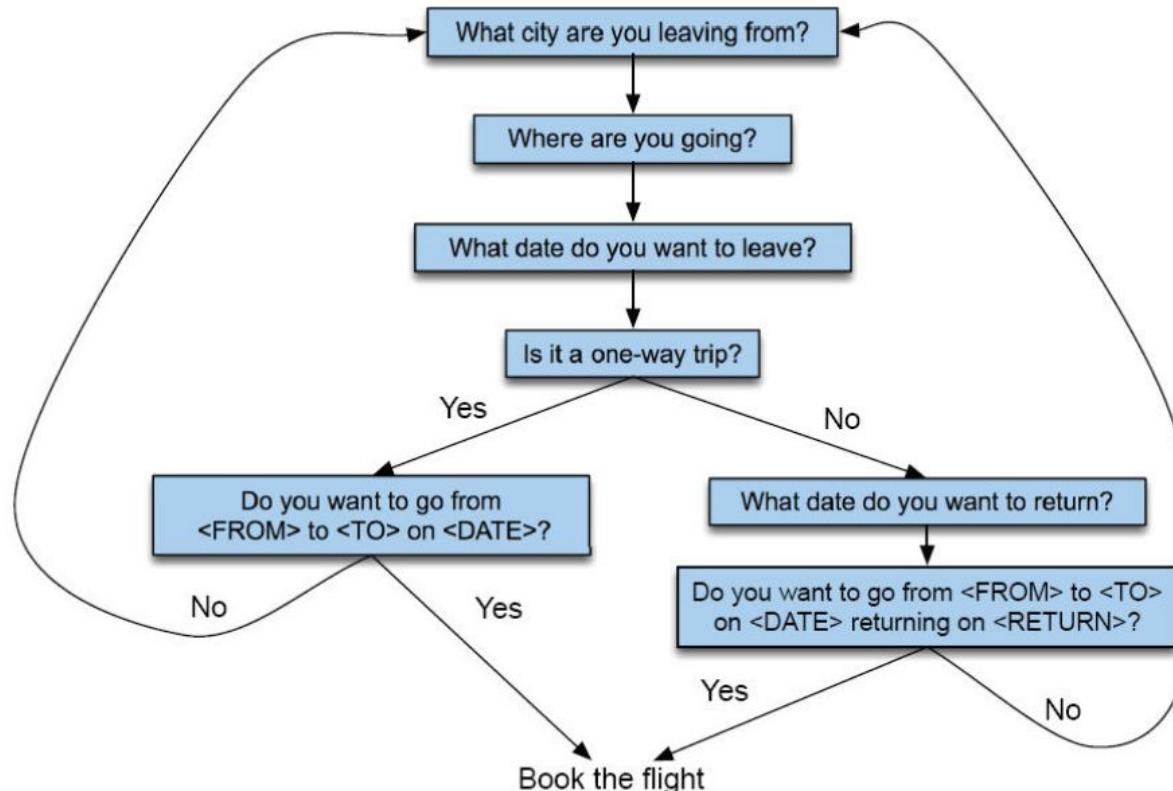
- Fun□
- Applications to counseling□
- Good for narrow, scriptable applications

□Cons

- □They don't really understand□
- Rule-based chatbots are expensive and brittle□
- Ranking-based chatbots can only mirror training data□
 - The case of Microsoft Tay□ (or, Garbage-in, Garbage-out)□

The future: combining chatbots with frame-based agents

Finite-state dialog managers



Finite-state dialog managers

- System completely controls the conversation with the user
- □It asks the user a series of questions□
- Ignoring (or misinterpreting) anything the user says that is not a direct answer to the system's questions

Finite-state dialog managers

- System completely controls the conversation with the user
- □It asks the user a series of questions□
- Ignoring (or misinterpreting) anything the user says that is not a direct answer to the system's questions
- **Single initiative + universals** can make it better
 - **Help**
 - **Start over**
 - **Correct**

Frame-based dialog agents

- 1977

GUS, A Frame-Driven Dialog System¹

**Daniel G. Bobrow, Ronald M. Kaplan, Martin Kay,
Donald A. Norman, Henry Thompson and
Terry Winograd**

*Xerox Palo Alto Research Center, 3333 Coyote Hill Road,
Palo Alto, CA 94304, U.S.A.*

Artificial Intelligence Journal, 1977



- Still the industrial state of the art □
- SIRI based on GUS architecture

GUS: Hello. My name is GUS. I can help you plan a simple trip by air. Where do you want to go ?

**The state of the
art in 1977 !!!!**

Client: I want to go to San Diego on May 28

GUS: What time do you want to leave?

Client: I must be in San Diego before 10 am

GUS: Would you like P.S.A. flight 102 that arrives at 9.15 am ?

Client: What is the next flight ?

GUS: Air California flight 310 that leaves at 8.30am and arrives at 10.00am

Client: I'll take the first one

GUS: In what name should I make a reservation, please?

Client: Dan Bobrow

GUS: I have confirmed the following flight: P.S.A. flight 102 on Wednesday May 28 from San Jose to San Diego that leaves at 7.30 am and arrives at 9.15 am.

GUS: What date do you want to return on ?

Client: On Friday in the evening.

GUS: Would you like the flight that leaves at 7.45 pm ?

Client: That's fine.

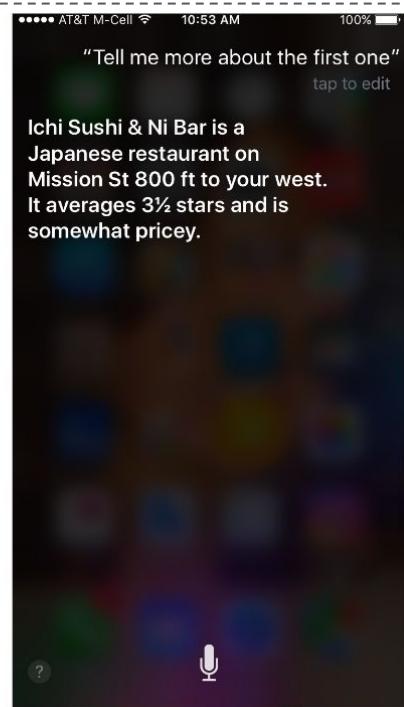
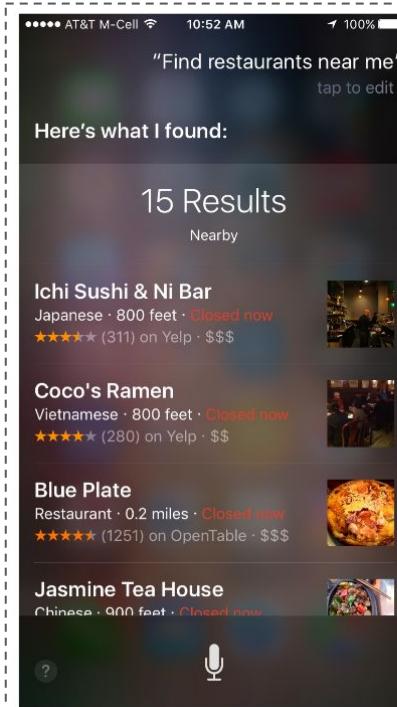
GUS: I have confirmed the following flight: P.S.A. flight 307 on Friday May 30 from San Diego to San Jose that leaves at 7.45 pm and arrives at 9.30 pm Thank you for calling. Goodbye

Frame-based dialog agents

- SIRI around 2014



- SIRI around 2017



The Frame

- A set of **slots**, to be filled with information of a given **type**
- Each associated with a **question** to the user

Slot	Type	Question
ORIGIN	city	What city are you leaving from?
DEST	city	Where are you going?
DEP DATE	date	What day would you like to leave?
DEP TIME	time	What time would you like to leave?
AIRLINE	line	What is your preferred airline?

Examples

Wake me tomorrow at six.

DOMAIN: ALARM-CLOCK

INTENT: SET-ALARM

TIME: 2017-07-01 0600-0800

Show me morning flights from Boston to SF on Tuesday.

DOMAIN: AIR-TRAVEL

INTENT: SHOW-FLIGHTS

ORIGIN-CITY: Boston

ORIGIN-DATE: Tuesday

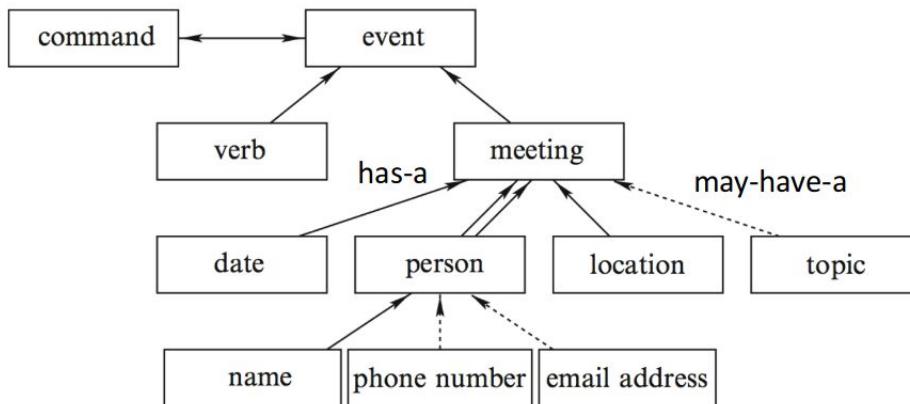
ORIGIN-TIME: morning

DEST-CITY: San Francisco

How to make a dialog?

Simple frame based dialog system

Part of ontology for meeting task

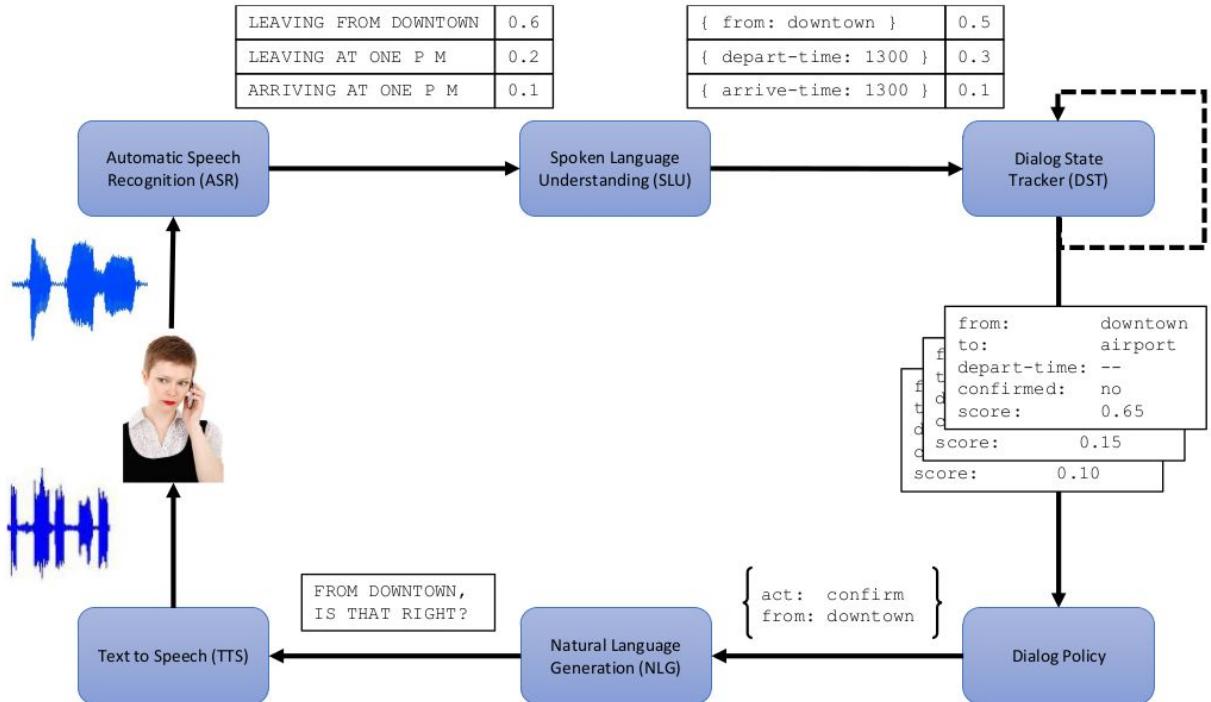


meeting concept: if you don't yet have a location, ask for a location

1. Understand intent, domain, fill slots
2. Ask questions about necessary slots to fill all of them
3. Make database query

Dialog state architectures

- also fill slots
- can do more than just ask:
make a proposal, reject etc
- have dialog policy
to decide what to say
- have dialog state
other than just slots states



Dialog acts

Если рассматривать каждое сообщение в диалоге как действие, то эти действия можно поделить на 3 большие группы

Action group	What does it mean?	Examples
Directives (команды)	Высказывание с целью заставить адресата сделать что-то, указание на действие с ожиданием ответной реакции	advising, asking, forbidding, inviting, ordering, requesting
Constatives (информирование)	Сообщение адресату какой-либо фактической информации	answering, claiming, confirming, denying, disagreeing, stating, dontcare
Acknowledgments (выражение отношения)	Выражения социальных аспектов беседы	apologizing, greeting, thanking, accepting an acknowledgment

Dialog acts in the restaurant domain

Tag	Sys	User	Description
HELLO($a = x, b = y, \dots$)	✓	✓	Open a dialog and give info $a = x, b = y, \dots$
INFORM($a = x, b = y, \dots$)	✓	✓	Give info $a = x, b = y, \dots$
REQUEST($a, b = x, \dots$)	✓	✓	Request value for a given $b = x, \dots$
REQALTS($a = x, \dots$)	✗	✓	Request alternative with $a = x, \dots$
CONFIRM($a = x, b = y, \dots$)	✓	✓	Explicitly confirm $a = x, b = y, \dots$
CONFREQ($a = x, \dots, d$)	✓	✗	Implicitly confirm $a = x, \dots$ and request value of d
SELECT($a = x, a = y$)	✓	✗	Implicitly confirm $a = x, \dots$ and request value of d
AFFIRM($a = x, b = y, \dots$)	✓	✓	Affirm and give further info $a = x, b = y, \dots$
NEGATE($a = x$)	✗	✓	Negate and give corrected value $a = x$
DENY($a = x$)	✗	✓	Deny that $a = x$
BYE()	✓	✓	Close a dialog

Figure 25.4 Dialog acts used by the HIS restaurant recommendation system of [Young et al. \(2010\)](#). The Sys and User columns indicate which acts are valid as system outputs and user inputs, respectively.

Dialog acts in the restaurant domain

Utterance	Dialog act
U: Hi, I am looking for somewhere to eat.	hello(task = find, type=restaurant)
S: You are looking for a restaurant. What type of food do you like?	confreq(type = restaurant, food)
U: I'd like an Italian somewhere near the museum.	inform(food = Italian, near=museum)
S: Roma is a nice Italian restaurant near the museum.	inform(name = "Roma", type = restaurant, food = Italian, near = museum)
U: Is it reasonably priced?	confirm(pricerange = moderate)
S: Yes, Roma is in the moderate price range.	affirm(name = "Roma", pricerange = moderate)
U: What is the phone number?	request(phone)
S: The number of Roma is 385456.	inform(name = "Roma", phone = "385456")
U: Ok, thank you goodbye.	bye()

Figure 25.5 A sample dialog from the HIS System of [Young et al. \(2010\)](#) using the dialog acts in Fig. 25.4.

Dialog structure elements

- adjacency pair
 - QUESTIONS and ANSWERS
 - PROPOSAL and ACCEPTANCE (or REJECTION)
 - COMPLIMENTS and DOWNPLAYERS
 - GREETING and GREETING

Dialog structure elements

- **subdialogs** (as a result of bad ASR)
 - clarification question between REQUEST and RESPONSE

User: What do you have going to UNKNOWN_WORD on the 5th?
System: Let's see, going where on the 5th?
User: Going to Hong Kong.
System: OK, here are some flights...

- user correction acts (bad recognition of hyperarticulation)

BAL-TI-MORE, not Boston

- **pre-sequence questions** (just ask, does bot ever have such skill)

User: Can you make train reservations?
System: Yes I can.
User: Great, I'd like to reserve a seat on the 4pm train to New York.

Dialog State

State example after each dialog turn

User: I'm looking for a cheaper restaurant
inform(price=cheap)

System: Sure. What kind - and where?

User: Thai food, somewhere downtown
inform(price=cheap, food=Thai, area=centre)

System: The House serves cheap Thai food

User: Where is it?
inform(price=cheap, food=Thai, area=centre); request(address)

System: The House is at 106 Regent Street

Dialog policy

(confirmation)

- Explicit confirmation

S:	Which city do you want to leave from?
U:	Baltimore.
S:	Do you want to leave from Baltimore?
U:	Yes.
U:	I'd like to fly from Denver Colorado to New York City on September twenty first in the morning on United Airlines
S:	Let's see then. I have you going from Denver Colorado to New York on September twenty first. Is that correct?
U:	Yes

- Implicit confirmation

U:	I want to travel to Berlin
S:	When do you want to travel to Berlin?
U2:	Hi I'd like to fly to Seattle Tuesday Morning
A3:	Traveling to Seattle on Tuesday, August eleventh in the morning.
	Your full name?

Dialog policy

(progressive prompting
if agent didn't get answer)

- Reject

System: When would you like to leave?

Caller: Well, um, I need to be in New York in time for the first World Series game.

System: <reject>. Sorry, I didn't get that. Please say the month and day you'd like to leave.

Caller: I wanna go on October fifteenth.

Dialog policy

(progressive prompting
if agent didn't get answer)

- Reject

System: When would you like to leave?

Caller: Well, um, I need to be in New York in time for the first World Series game.

System: <reject>. Sorry, I didn't get that. Please say the month and day you'd like to leave.

Caller: I wanna go on October fifteenth.

- Simple dialog policy

$< \alpha$	low confidence	reject
$\geq \alpha$	above the threshold	confirm explicitly
$\geq \beta$	high confidence	confirm implicitly
$\geq \gamma$	very high confidence	don't confirm at all

Dialog policy

RL?



Tim Vieira

@xtimv



When you say, "This is a reinforcement learning problem," you should say it with the same excitement as "This is NP-hard."

123 2:48 AM - Nov 6, 2016



33 people are talking about this



NLG (natural language generation)

- State of the art templates:3
- Delexicalized string
 - What time on [depart date] would you like to leave [depart airport]?

QUERY DEPART_TIME And what time would you like to leave [depart_city **Pittsburgh**]?

QUERY ARRIVE_CITY And you're flying into what city?

QUERY ARRIVE_TIME What time on [arrive_date **May 5**]?

INFORM FLIGHT The flight departs [depart_airport **PGH**] at [depart_time **10 am**] and arrives [arrive_city **Seattle**] at [arrive_time **12:05 their time**].

NLU (natural language understanding)

- Slot filling
 - IOB Tagging
 - tag for the beginning (B) and inside (I) of each slot label,
 - plus one for tokens outside (O) any slot label.
 - $2n + 1$ tags, where n is the number of slots.

B-DESTINASTION

I-DESTINATION

B-DEPART_TIME

I-DEPART_TIME

O

0 0 0 0 B-DES I-DES 0 B-DEPTIME I-DEPTIME 0
I want to fly to San Francisco on Monday afternoon please

Evaluation

- Slot recognition error rate
- End-to-end evaluation (Task success)

“Make an appointment with Chris at 10:30 in Gates 104”

Slot	Filler
PERSON	Chris
TIME	11:30 a.m.
ROOM	Gates 104

Slot error rate: 1/3

Task success: At end, was the correct meeting added to the calendar?

Rasa



- Rasa Core

intents	things you expect users to say. See Rasa NLU
actions	things your bot can do and say
templates	template strings for the things your bot can say
entities	pieces of info you want to extract from messages. See Rasa NLU
slots	information to keep track of during a conversation (e.g. a user's age) - see Using Slots

Rasa

Quickstart



Hey! how are you?

great!

super sad

Here is something to cheer you up



hello

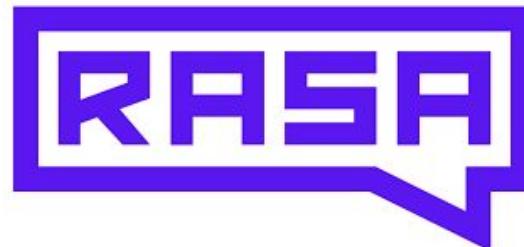
```
stories_md = """  
## happy path  
* greet  
  - utter_greet  
* mood_great  
  - utter_happy  
  
## sad path 1  
* greet  
  - utter_greet  
* mood_unhappy  
  - utter_cheer_up  
  - utter_did_that_help  
* mood_affirm  
  - utter_happy  
  
## sad path 2  
* greet  
  - utter_greet  
* mood_unhappy  
  - utter_cheer_up  
  - utter_did_that_help  
* mood_denry  
  - utter_goodbye  
  
## say goodbye  
* goodbye  
  - utter_goodbye  
"""
```

```
domain_yml = """  
intents:  
  - greet  
  - goodbye  
  - mood_affirm  
  - mood_denry  
  - mood_great  
  - mood_unhappy  
  
actions:  
  - utter_greet  
  - utter_cheer_up  
  - utter_did_that_help  
  - utter_happy  
  - utter_goodbye  
  
templates:  
  utter_greet:  
    - text: "Hey! How are you?"  
  
  utter_cheer_up:  
    - text: "Here is something to cheer you up:"  
      image: "https://i.imgur.com/nGF1K8f.jpg"  
  
  utter_did_that_help:  
    - text: "Did that help you?"  
  
  utter_happy:  
    - text: "Great carry on!"  
  
  utter_goodbye:  
    - text: "Bye"  
...  
%store domain_yml > domain.yml  
  
print("Done!")
```

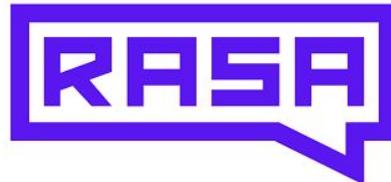
Rasa NLU

"I am looking for a Mexican restaurant in the center of town"

```
{  
  "intent": "search_restaurant",  
  "entities": {  
    "cuisine" : "Mexican",  
    "location" : "center"  
  }  
}
```



Rasa NLU



RasaHQ / rasa

Watch ▾

245

Star

5,037

Fork

1,464

Code

Issues 160

Pull requests 17

Projects 0

Wiki

Insights

Branch: master ▾

[rasa](#) / [rasa](#) / [nlu](#) / [classifiers](#) /

Create new file

Upload files

Find file

History



MetcalfeTom Merge branch 'master' into remove_outdated_warning_fix

Latest commit 568252d 6 days ago

..

[__init__.py](#)

moved files into rasa

27 days ago

[embedding_intent_classifier.py](#)

Reformatted code.

12 days ago

[keyword_intent_classifier.py](#)

Reformatted code.

12 days ago

[mitie_intent_classifier.py](#)

Reformatted code.

12 days ago

[sklearn_intent_classifier.py](#)

Merge branch 'master' into remove_outdated_warning_fix

6 days ago