



# Implementing a Chatbot using the DeepPavlov Framework

Mohammed Yasser BENIGMIM

Supervisors : Raphaël TRONCY, Thomas SCHLEIDER, Thibault EHRHART

Data Science Department- EURECOM

# “ Create a chatbot on tourism using DeepPavlov framework ”



**Duration of the Project : 8th March ➡ 12th June (~3 months)**



**Weekly meetings**



**Feedbacks and Improvement Tracks**

# Summary :

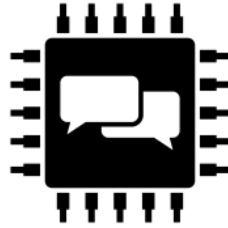


1. State of the Art : NLP, chatbots
2. DeepPavlov : Named Entity Recognition, Goal oriented bot
3. Tourism chatbot
4. Future Work : RoadMap
5. Conclusion

---

# State of the Art

# Natural Language Processing



- NLP is a subfield of Artificial Intelligence.
- NLP describes the ability of a machine to understand the meaning of a sentence and finally determine the appropriate action.
- NLP can be divided into two parts :
  - **Natural Language Understanding (NLU)** : understands the intent and the parameters of the sentence.
  - **Natural Language Generation (NLG)** : generate the according answer to the sentence.

# 85%

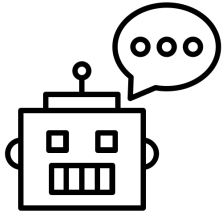
The percentage of customer interactions that will be handled with a chatbot in 2020.

- Gartner Report



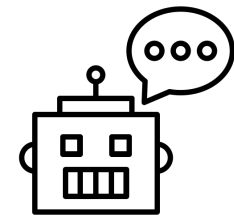
# Chatbots

---



- A chatbot is considered as : “ A computer program designed to simulate conversation with human users, especially over the Internet ”.
- Artificial intelligence made it possible for chatbots to “learn”.
- “Intelligence” to perform tasks, solve problems without human intervention.

# How it works

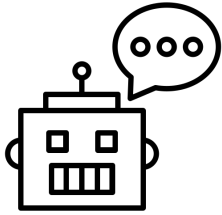


- General mechanism :





# How it works



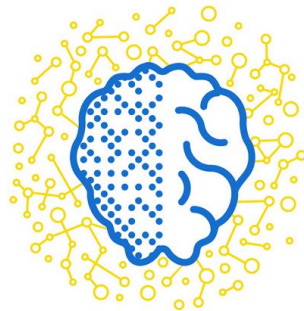
- Functioning of the NLU :



---

# DeepPavlov

# Presentation of DeepPavlov



- Deep learning-based library for creating chatbots
  - Created in June in 2018 by Neural Network and Deep Learning Lab of the *Moscow Institute of Physics and Technology* (MIPT) in a project called *iPavlov*
  - DeepPavlov contains lot of components : *Named Entity Recognition, Intent classifier, etc...*
  - Different kind of bots : *Goal-oriented bots, chit-chat models, etc ...*
- One of the 10 teams selected for the Alexa Prize Socialbot Grand Challenge 3 !

# Named Entity Recognition



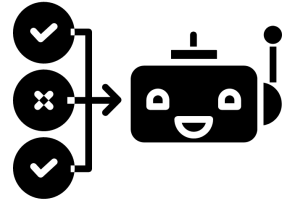
- Subtask in the NLU, classifies named entities present in a text in to pre-defined categories like “individuals”, “companies”, “cities”, “dates” etc ...



- We should not confuse between the **intent** and the **entities** !

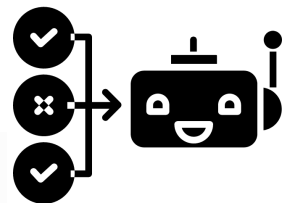
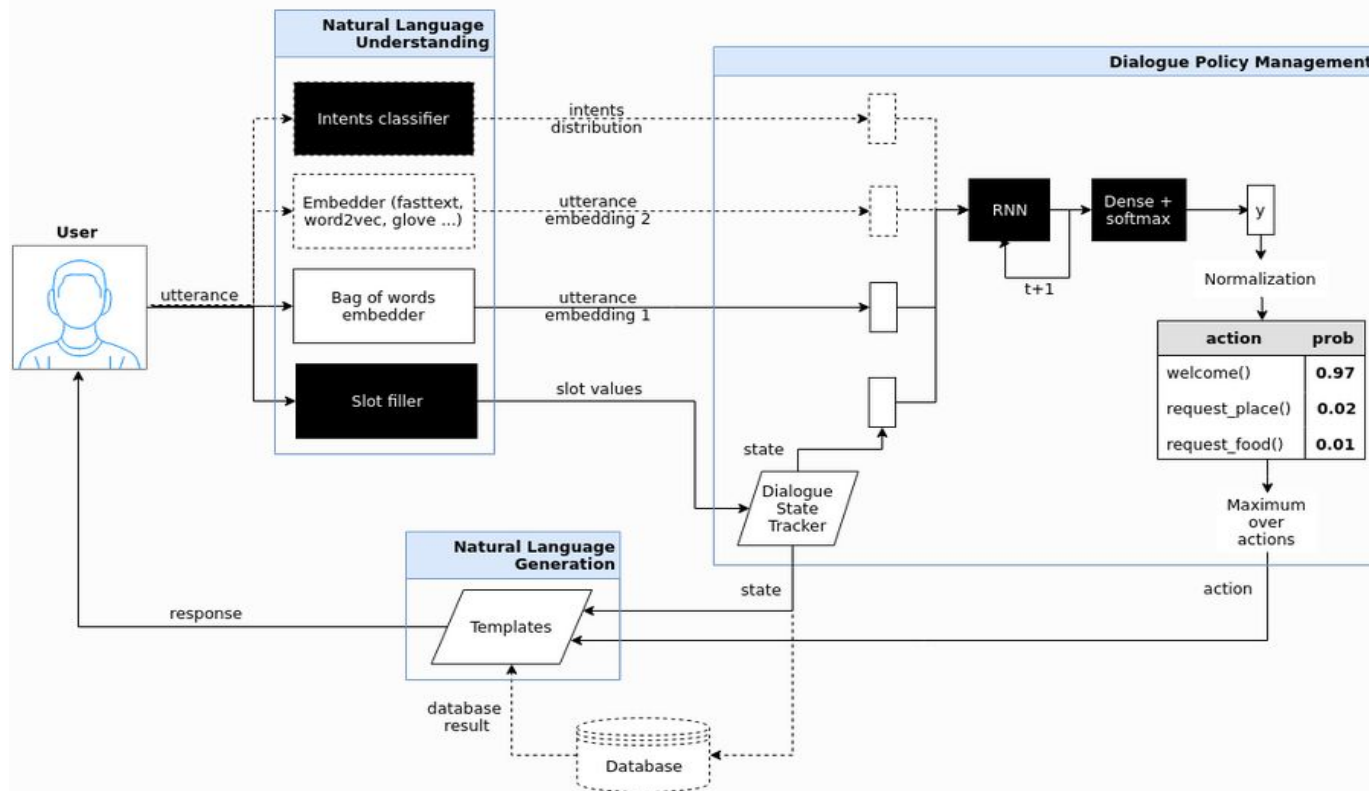
# Goal oriented bot

---



- A goal-oriented (GO) chatbot attempts to solve a specific problem for a user, it can help people book a ticket, find a reservation, etc ...
- 2.5 Bn hours saved for businesses and consumers combined in 2023 ! - Juniper Research
- Also called “virtual assistant”

# Goal oriented bot



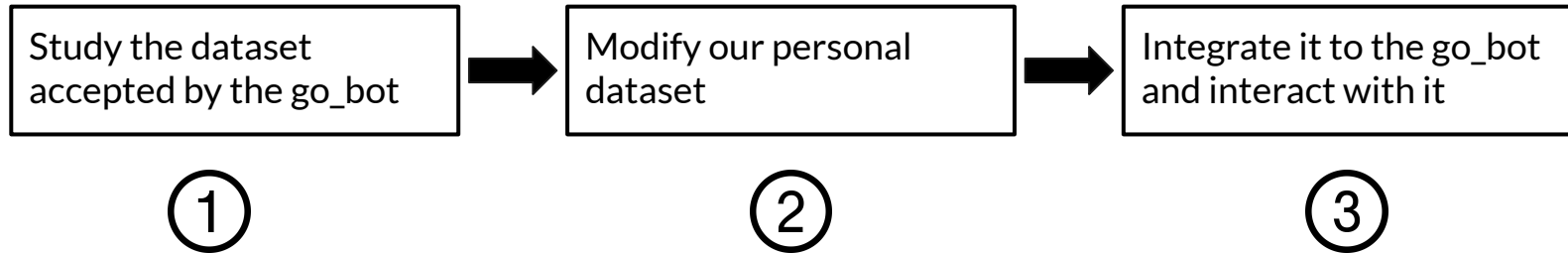
---

# Personal work

# Tourism chatbot



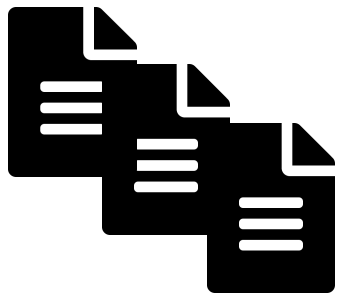
- Objective: “ Use the goal oriented bot provided by DeepPavlov to train it on a our personal dataset about tourism ”





# ① Study the dataset : DSTC 2

- Dialogue State Tracking Challenge 2 : dataset about **restaurant reservation**



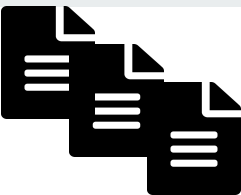
train/val/test file



SQLite Database



Template file



# ① train/val/test file

- Train the chatbot to recognize the restaurants and respond accordingly

```
{
  "speaker": 2, "text": "Hello, welcome to the Cambridge restaurant system. You can ask for restaurants by area, price range or food type. How may I help you?", "dialog_acts": [{"act": "welcomemsg", "slots": []}],
  "speaker": 1, "text": "im looking for a cheap restaurant in the south part of town", "goals": {"pricerange": "cheap", "area": "south"}, "dialog_acts": [{"slots": [{"pricerange": "cheap"}], "act": "inform"}, {"slots": [{"area": "south"}], "act": "inform"}],
  "speaker": 2, "text": "api_call area='south' food='dontcare' pricerange='cheap'", "dialog_acts": [{"act": "api_call", "slots": [{"pricerange": "cheap"}, {"area": "south"}]}, {"db_result": {"pricerange": "cheap", "area": "south", "addr": "cambridge leisure park clifton way", "phone": "01223 327908", "food": "portuguese", "postcode": "c.b 1, 7 d.y", "name": "nandos"}}],
  "speaker": 2, "text": "Nandos is a nice place in the south of town and the prices are cheap.", "dialog_acts": [{"act": "inform_area+inform_pricerange+offer_name", "slots": [{"name": "nandos"}, {"pricerange": "cheap"}, {"area": "south"}]}],
  "speaker": 1, "text": "what is the address", "goals": {"pricerange": "cheap", "area": "south"}, "dialog_acts": [{"slots": [{"slot": "addr"}], "act": "request"}],
  "speaker": 2, "text": "Sure, nandos is on Cambridge Leisure Park Clifton Way.", "dialog_acts": [{"act": "inform_addr+offer_name", "slots": [{"name": "nandos"}, {"addr": "cambridge leisure park clifton way"}]}],
  "speaker": 1, "text": "thank you good bye", "goals": {"pricerange": "cheap", "area": "south"}, "dialog_acts": [{"slots": [], "act": "thankyou"}, {"slots": [], "act": "bye"}],
  "speaker": 2, "text": "You are welcome!", "dialog_acts": [{"act": "bye", "slots": []}]
}
```

training file : 967 dialogs

test file : 576 dialogs

validation file : 575 dialogs

# ① SQLite Database



- Contains all the restaurants mentioned in the files with their attributes

id	addr	pricerange	food	phone	postcode	area	name
	cambridge leisure park clifton way ...	cheap	chinese	01223 244277	c.b 1, 7 d.y	south	the lucky star
	71 castle street city centre	expensive	indian	01223 366668	c.b 3, 0 a.h	west	cocum
	290 mill road city centre	expensive	gastropub	01223 247877	c.b 1, 3 n.l	east	royal standard
	cambridge leisure park clifton way	cheap	portuguese	01223 327908	c.b 1, 7 d.y	south	nandos
	20 milton road chesterton	cheap	italian	01223 351707	c.b 4, 1 j.y	north	da vinci pizzeria
	2g cambridge leisure park cherry hi...	expensive	mexican	01223 400170	c.b 1, 7 d.y	south	chiquito restaurant bar
	17 magdalene street city centre	cheap	vietnamese	01223 362456	c.b 3, 0 a.f	west	thanh binh
	169 high street chesterton chesterton	expensive	asian oriental	01223 356555	c.b 4, 1 n.l	north	saigon city
	st. michael's church trinity street ci...	expensive	european	01223 309147	c.b 2, 1 s.u	centre	michaelhouse cafe

101 different restaurants



# ① Template file :

- provide a mapping from actions to text templates

```
bye      You are welcome!
canthear      Sorry, I can't hear you.
canthelp_area  I'm sorry but there is no #area american restaurant in the #area of town.
canthelp_area_food  Sorry there is no #food restaurant in the #area of town.
canthelp_area_food_pricerange  Sorry there is no #pricerange restaurant in the #area of town serving #food food.
canthelp_area_pricerange  Sorry there is no #pricerange restaurant in the #area of town serving #area american food.
canthelp_food  I am sorry but there is no #food restaurant that matches your request.
canthelp_food_pricerange  Sorry there is no #food restaurant in the #pricerange price range.
confirm-domain  You are looking for a restaurant is that right?
expl-conf_area  Did you say you are looking for a restaurant in the #area of town?      Ok, a restaurant in any part of town is that right?
expl-conf_food  You are looking for a #food restaurant right?  You are looking for a restaurant serving any kind of food right?
```

## ② Modify our own dataset : RASA dataset

```
{  
  "intent": "find.accommodation",  
  "entities": [  
    {  
      "start": 15,  
      "end": 20,  
      "value": "today",  
      "entity": "datetime"  
    },  
    {  
      "start": 25,  
      "end": 29,  
      "value": "Nice",  
      "entity": "city"  
    }  
  ],  
  "text": "I want to stay today in Nice"  
},
```

the concept of dialogue is  
not present

not suitable for our case

## ② Minotour Chat logs

A	B	C	D	E	F	G	H	I	J	K	L	M	N
id	date	timestamp	platform	channel	user	intent	confidence	lang	raw_message	clean_message	timestamp	response	rating
cc7b5e	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	7110469a-1f	Default Welcome Int	1	en	"Hi"	"Hi"	1543916	"Hi! My name is Minotour I'm here to advice you on evi	0
ce22fe	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	7110469a-1f	find.event	1	en	"show events in Nice fo	"show events in Nice for tomorrow"	1543916	<result_card>	0
eee4e1	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	abandon	1	en	"reboot"	"reboot"	1543916	"Of course let's see how I can help you"	0
b75269	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	1	en	"show me events"	"show me events"	1543916	"What is the city?"	0
1572ef	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	0.87140	en	"nice"	"nice"	1543916	"What is the datetime?"	0
a87b7e	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	1	en	"this month"	"this month"	1543916	<result_card>	0
d192ba	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	0.83409	en	"can you show me thea	"can you show me theater plays ? "	1543916	"What is the city?"	0
5b7325	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	0.87140	en	"Nice"	"Nice"	1543916	"What is the datetime?"	0
598dc8	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	1	en	"this month"	"this month"	1543916	<result_card>	0
bf5bce	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.generic	0.51999	en	"can you show me thea	"can you show me theater-plays ? "	1543916	"are you interested in events activities, sightseeings, k	0
e65b84	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	abandon	1	en	"reboot"	"reboot"	1543916	"Ok maybe I understood something wrong! Let's start a	0
191b95	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	abandon	1	en	"reboot"	"reboot"	1543917	"Sure what do you want to do now?"	0
7479fa	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.activity	1	en	"i would like to see thea	"i would like to see theater"	1543917	"Where do you want to go?"	0

the key concept of dialogue is present

## ② Comparison of the two datasets

Personal data:

```
{"speaker": 1, "text": "i need list of shopping and not restaurant ", "intent": "find.restaurant", "text_bot": "What is the city?", "act": "find.restaurant"}  
{"speaker": 1, "text": "i dont need this list", "intent": "Default Fallback Intent", "text_bot": "Can you say that again?", "act": "Default Fallback Intent"}  
{"speaker": 2, "text": "you speak frensh ?", "intent": "Default Fallback Intent", "text_bot": "What was that?", "act": "Default Fallback Intent"}  
{"speaker": 1, "text": "it's a question", "intent": "Default Fallback Intent", "text_bot": "What was that?", "act": "Default Fallback Intent"}  
{"speaker": 2, "text": "by", "intent": "Default Fallback Intent", "text_bot": "Can you say that again?", "act": "Default Fallback Intent"}  
{"speaker": 1, "text": "good by ", "intent": "Default Fallback Intent", "text_bot": "Sorry what was that?", "act": "Default Fallback Intent"}  
{"speaker": 2, "text": "ciao", "intent": "Default Fallback Intent", "text_bot": "Say that again?", "act": "Default Fallback Intent"}
```

VS

DSTC 2:

```
{"speaker": 2, "text": "Hello, welcome to the Cambridge restaurant system. You can ask for restaurants by area, price range or food type. How may I help you?", "dialog_acts": [{"act": "welcomensg", "slots": []}]  
{"speaker": 1, "text": "im looking for a cheap restaurant in the south part of town", "goals": {"pricerange": "cheap", "area": "south"}, "dialog_acts": [{"slots": [{"pricerange": "cheap"}], "act": "inform"}, {"slots": [{"area": "south"}], "act": "inform"}]  
{"speaker": 2, "text": "api_call area='south' food='dontcare' pricerange='cheap'", "dialog_acts": [{"act": "api_call", "slots": [{"pricerange": "cheap"}, {"area": "south"}]}, {"db_result": {"pricerange": "cheap", "area": "south", "addr": "cambridge leisure park clifton way", "phone": "01223 327908", "food": "portuguese", "postcode": "c.b 1, 7 d.y", "name": "nandos"}]  
{"speaker": 2, "text": "Nandos is a nice place in the south of town and the prices are cheap.", "dialog_acts": [{"act": "inform_area+inform_pricerange+offer_name", "slots": [{"name": "nandos"}, {"pricerange": "cheap"}, {"area": "south"}]}]  
{"speaker": 1, "text": "what is the address", "goals": {"pricerange": "cheap", "area": "south"}, "dialog_acts": [{"slots": [{"slot": "addr"}], "act": "request"}]  
{"speaker": 2, "text": "Sure, nandos is on Cambridge Leisure Park Clifton Way.", "dialog_acts": [{"act": "inform_addr+offer_name", "slots": [{"name": "nandos"}, {"addr": "cambridge leisure park clifton way"}]}]  
{"speaker": 1, "text": "thank you good bye", "goals": {"pricerange": "cheap", "area": "south"}, "dialog_acts": [{"slots": [], "act": "thankyou"}, {"slots": [], "act": "bye"}]  
{"speaker": 2, "text": "You are welcome!", "dialog_acts": [{"act": "bye", "slots": []}]}
```



## ② The output of the Dataset Readers

TOURISMReader :

VS

DSTC2Reader :

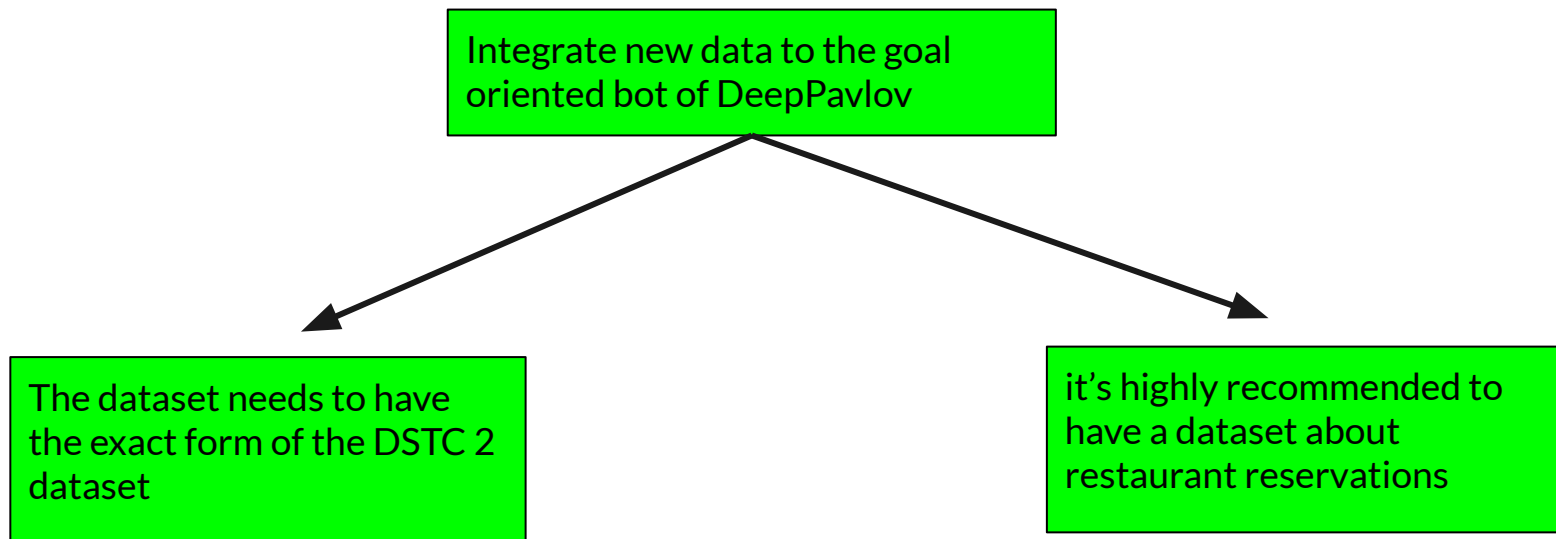
```
'intents': 'Default Fallback Intent',  
'text': 'how to go from nice airport to downtown ?'}},  
{'act': 'find.event', 'text': 'What is the datetime?'}},  
({'intents': 'Default Fallback Intent',  
'text': 'where is the tourist office'},  
{'act': 'Default Welcome Intent',  
'text': 'Good day! Are you bored? My name is MinoTour and I can suggest you concert  
hat category do you want me to look for?'}},  
({'intents': 'Default Fallback Intent',  
'text': 'where is the railway station ?'}},  
{'act': 'Default Welcome Intent',  
'text': 'Hi! My name is MinoTour :) What do you want to do?'}},  
({'intents': 'find.accommodation', 'text': 'show me an hotel with car park'},  
{'act': 'find.event', 'text': 'What is the datetime?'}},
```

not suitable for our case

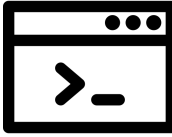
```
({'intents': [{'act': 'inform', 'slots': [['food', 'italian']]}],  
{'act': 'inform', 'slots': [['area', 'west']]}],  
'text': 'im looking for a restaurant in the west part of town serving italian food'},  
{'act': 'request_pricerange',  
'text': 'Would you like something in the cheap, moderate, or expensive price range?'}},  
({'intents': [{'act': 'inform', 'slots': [['this', 'dontcare']]}],  
'text': 'doesnt matter'},  
{'act': 'select_pricerange',  
'text': 'Sorry would you like something in the cheap price range or you dont care.'}},  
({'intents': [{'act': 'inform', 'slots': [['this', 'dontcare']]}],  
'text': 'i dont care'},  
{'act': 'api_call',  
'text': 'api_call area="west" food="italian" pricerange="dontcare"'}},
```



## ② Integrate new data in DeepPavlov



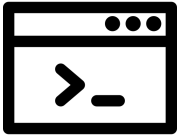
## ③ Interaction with the go\_bot



- After moving from Windows to Linux, we managed to interact with the go\_bot :

```
x::Hi, I want cheap food
2019-06-13 10:45:08.788 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'pricerange': 'cheap'}, got 23 results.
>> The lucky star serves chinese food.
x::is there anything with french food please ?
2019-06-13 10:45:21.908 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'pricerange': 'cheap', 'food': 'french'}, got 0 results.
>> #name serves french food.
x::something expensive ?
2019-06-13 10:45:36.689 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'pricerange': 'expensive', 'food': 'french'}, got 2 results.
>> Cote serves french food in the expensive price range.
x::in what part of town is it ?
2019-06-13 10:45:47.940 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'pricerange': 'expensive', 'food': 'french'}, got 2 results.
>> Restaurant two two is a nice place in the north of town serving tasty french food.
x::ok, give me the adress
>> Sure, restaurant two two is on 22 chesterton road chesterton.
x::and the phone number too
>> The phone number of restaurant two two is 01223 351880.
x::thank you !
>> You are welcome!
```

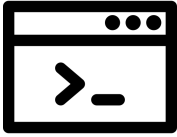
### ③ Interaction with the go\_bot



- We tried to interact with the go\_bot with new added data
- Add the new restaurant in the SQLite database (**previously seen food origin**)

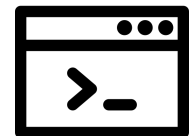
addr	pricerange	food	phone	postcode	area	name
36 saint andrews street	expensive	japanese	01223 462354	c.b 2, 3 a.r	centre	wagamama
Boulevard Dugommier	cheap	japanese	0678965346	06500	north	sushi shop

## ③ Interaction with the go\_bot



```
x::Hi I am looking for a japanese restaurant
2019-06-16 16:21:31.668 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'food': 'japanese'}, got 2 results.
>> Wagamama serves japanese food.
x::I would to eat in a cheap one
2019-06-16 16:22:52.146 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'food': 'japanese', 'pricerange': 'cheap'}, got 1 results.
>> Sushi shop serves japanese food.
x::Great! In which part of the town it is ?
2019-06-16 16:23:13.3 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'food': 'japanese', 'pricerange': 'cheap'}, got 1 results.
>> Sushi shop is a nice place in the north of town serving tasty japanese food.
x::Cool! can I have the address please ?
>> Sure, sushi shop is on Boulevard Dugommier.
x::and the phone number please
>> The phone number of sushi shop is 0678965346.
x::Thank you very much
>> You are welcome!
```

### ③ Interaction with the go\_bot

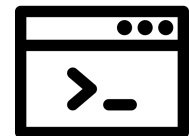


- Add the new restaurant in the SQLite database with **new food origin (argentinian)** that **didn't exist** before in the DSTC2 dataset and the SQLite database

```
x::Hi I want argentinian food
2019-06-16 16:30:41.886 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'food': 'indian'}, got 22 results.
>> Cocum serves indian food.
x:I said argentinian
>> Cocum serves indian food.
x:a tuscan restaurant ?
2019-06-16 16:31:18.907 INFO in 'deeppavlov.models.go_bot.network'['network'] at line 472: Made api_call with {'food': 'tuscan'}, got 0 results.
>> I am sorry but there is no tuscan restaurant that matches your request.
```

- The **Tuscan** restaurant **didn't exist** in the SQLite database, but **was mentioned** in the DSTC2 dataset !

### ③ Interaction with the go\_bot



- After adding a dialogue about the argentinian restaurant in the DSTC 2 dataset, we had some technical errors :

```
deepavlov.core.common.errors.ConfigError: 'obs_size' parameter must be equal to saved model parameter value '611', but is equal to '612'
```

- In the beginning we thought it's about the length of the training set, so we deleted one random dialogue, but it was not the case :

```
File "/home/yasser/anaconda3/lib/python3.6/site-packages/tensorflow/python/client/session.py", line 1076, in _run_str(subfeed_t.get_shape()))
ValueError: Cannot feed value of shape (1, 1, 612) for Tensor 'features:0', which has shape '(?, ?, 611)'
```

- The problem is purely technical, can be solved !

---

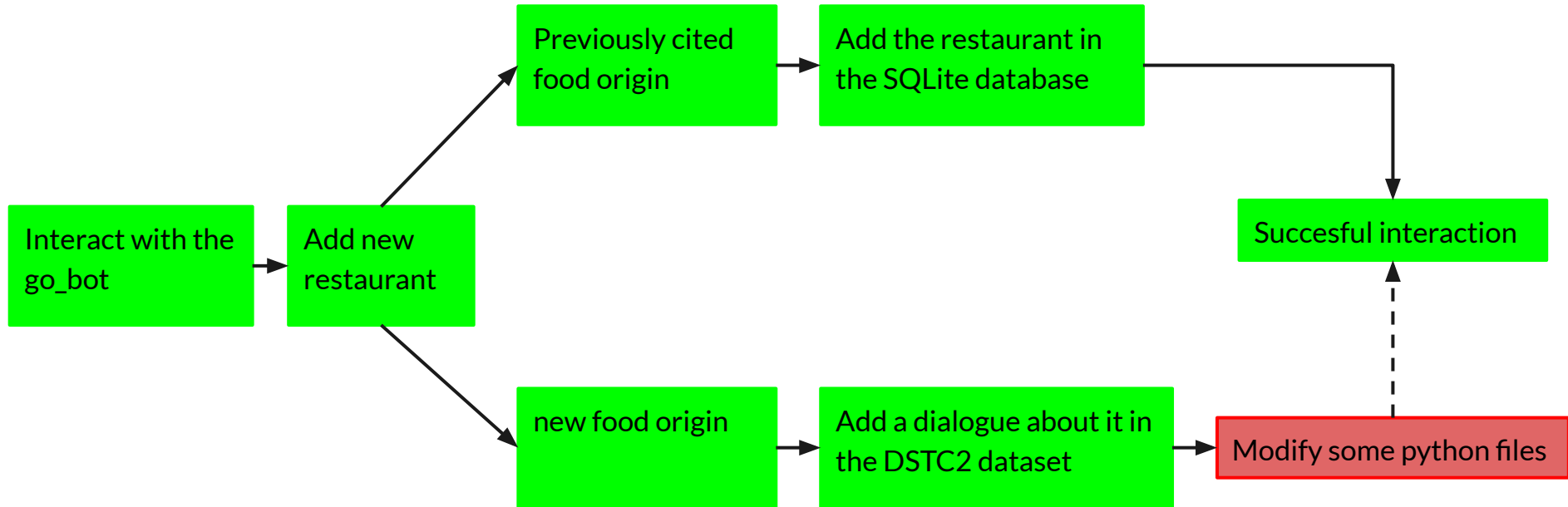
# Demonstration

---

# Future Work : RoadMap



# Road Map :



# Conclusion :

Interesting

Challenging

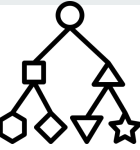
Inspiring

---

# Thank you !

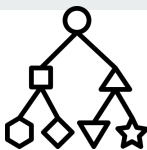


# Intent classification



- The core idea is to train the NLU of the chatbot to recognize the intent of each sentence.

hi	- greet
hello	- greet
bye	- end_conversation
goodbye	- end_conversation
see you	- end_conversation
Suggest me some mexican restaurants	- restaurant_search
im looking for restaurants	- restaurant_search



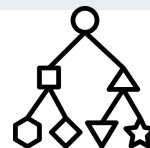
# Intent classification : SNIPS dataset

- Aims to train a chatbot for intent classification.
- Contains 15 000 rows (all different).

text	intent
Can you add a track to my Spain Top 50 playlist	AddToPlaylist
what is the weather forecast for Maryland	GetWeather
Give the Hollow Man three out of 6 stars	RateBook

- Contains 7 different intents : *“SearchCreativeWork”*, *“GetWeather”*, *“BookRestaurant”*, *“PlayMusic”*, *“AddToPlaylist”*, *“RateBook”*, *“SearchScreeningEvent”*

# Intent classification : Work with new data



- Minotour chat logs :

A	B	C	D	E	F	G	H	I	J	K	L	M	N
id	date	timestamp	platform	channel	user	intent	confidence	lang	raw_message	clean_message	timestamp	response	rating
cc7b5e	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	7110469a-1l	Default Welcome Int	1		"Hi"	"Hi"	1543916	"Hi! My name is Minotour I'm here to advice you on ev	0
ce22fee	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	7110469a-1l	find.event	1	en	"show events in Nice fo	"show events in Nice for tomorrow"	1543916	<result_card>	0
eee4e1	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	abandon	1	en	"reboot"	"reboot"	1543916	"Of course let's see how I can help you"	0
b75269	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	1	en	"show me events"	"show me events"	1543916	"What is the city?"	0
1572eff	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	0.87140	en	"nice"	"nice"	1543916	"What is the datetime?"	0
a87b7e	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	1	en	"this month"	"this month"	1543916	<result_card>	0
d192ba	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	0.83409	en	"can you show me thea	"can you show me theater plays ? "	1543916	"What is the city?"	0
5b7325	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	0.87140	en	"Nice"	"Nice"	1543916	"What is the datetime?"	0
598dc8	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.event	1	en	"this month"	"this month"	1543916	<result_card>	0
b5bce	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.generic	0.51999	en	"can you show me thea	"can you show me theater-plays ? "	1543916	"are you interested in events activities, sightseeings, k	0
e65b84	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	abandon	1	en	"reboot"	"reboot"	1543916	"Ok maybe I understood something wrong! Let's start a	0
191b95	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	abandon	1	en	"reboot"	"reboot"	1543917	"Sure what do you want to do now?"	0
7479fa	2018-12-0	1543916	webchat	<a href="https://mir">https://mir</a>	b51d1308-0	find.activity	1	en	"i would like to see thea	"i would like to see theater"	1543917	"Where do you want to go?"	0

- After modifying the dataset : 4159 rows ! (15 000 rows in the SNIPS dataset)

# Intent classification : Results

