

Conversational Interfaces for Chat Bot and Artificial Intelligence using Python

Idowu Adeleke



If you've passed your 25th birthday,
you were there when the aliens
landed



Idowu Adeleke

pythonista

data science student



@Idowu_adelek

Session Objectives


- Exposure to conversational interfaces
- Have you convinced
- Examples
- Personal goals
 - ❑ Memorable presentation
 - ❑ Learn from the audience

Quiet thought :

Many years back people on earth had **epic challenges** they looked forward to solve



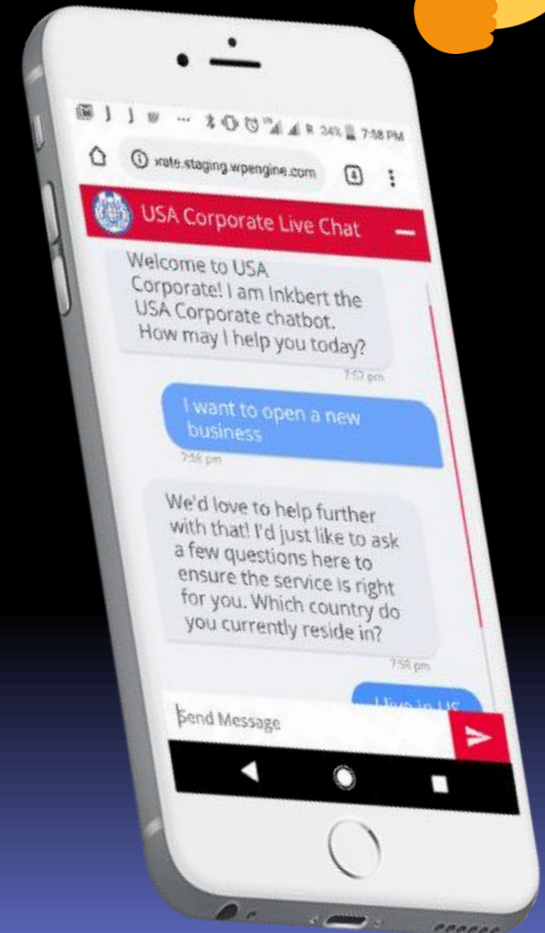
What is my generation's epic challenge????



**check out
some Data**

What are conversational interfaces?

- Next evolution in the user interface
- Teaching computers to communicate like people



Case Study:

Build a conversational interface that will inform users about the weather condition anywhere

Introducing Frank



How to create your own



- Language Parsing tool- Parse it
- Knowledge Base- Get answer
- Context Management – Store context
- User experience and server- Say it



Ah ha !



Every “business oriented ” bot creator needs:

- Parser
- Knowledge Base
- Context manager
- UI

Why not use an open source library for best in each class?

Some Python library



- RASA NLU
- RASA core
- Chatterbot
- Tensor layer
- Will
- QQbot
- PyMessenger
- Flask-Assistant
- fbmq
- rivescript
- Red-Discord Bot
- Dragonfire
- Legobot
- ehforwarderbot
- programy
- chat
- permabots
- bothu
- chatbotAI
- vexbot

Building Frank with the RASA Library

```
from rasa_nlu.converters import load_data
from rasa_nlu.config import RasaNLUConfig
from rasa_nlu.model import Trainer
from rasa_nlu.model import Metadata, Interpreter

def train_nlu(data, config, model_dir):
    training_data = load_data(data)
    trainer = Trainer(RasaNLUConfig(config))
    trainer.train(training_data)
    model_directory = trainer.persist(model_dir, fixed_model_name = 'weathernlu')

def run_nlu():
    interpreter = Interpreter.load('./models/nlu/default/weathernlu', RasaNLUConfig('config_spacy.json'))
    print(interpreter.parse(u"I am planning my holiday to Barcelona. I wonder what is the weather out there."))

if __name__ == '__main__':
    #train_nlu('./data/data.json', 'config_spacy.json', './models/nlu')
    run_nlu()
```

! weather_domain.yml x

```
1 slots:
2   location:
3     type: text
4
5
6 intents:
7   - greet
8   - goodbye
9   - inform
10
11
12 entities:
13   - location
14
15 templates:
16   utter_greet:
17     - 'Hello! How can I help?'
18   utter_goodbye:
19     - 'Talk to you later.'
20     - 'Bye bye :('
21   utter_ask_location:
22     - 'In what location?'
23
24
25 actions:
26   - utter_greet
27   - utter_goodbye
28   - utter_ask_location
29   - actions.ActionWeather
30
```



```
25/25 [=====] - 0s - loss: 0.9914 - acc: 0.8000
Epoch 197/200
25/25 [=====] - 0s - loss: 1.1293 - acc: 0.6400
Epoch 198/200
25/25 [=====] - 0s - loss: 0.9757 - acc: 0.7600
Epoch 199/200
25/25 [=====] - 0s - loss: 1.0444 - acc: 0.8000
Epoch 200/200
25/25 [=====] - 0s - loss: 1.0216 - acc: 0.6800
INFO:rasa_core.policies.keras_policy:Done fitting keras policy model
Bot loaded. Type a message and press enter :
Hello
-----
Chat history:

    bot did:      None

    bot did:      action_listen

    user said:    Hello

                whose intent is:      greet

we currently have slots: location: None
-----
The bot wants to [utter_greet] due to the intent. Is this correct?

1.      Yes
2.      No, intent is right but the action is wrong
3.      The intent is wrong
0.      Export current conversations as stories and quit
```

📄 stories.md ✕

```
1  ## Generated Story 6361943907618605498
2  * greet
3    - utter_greet
4  * inform
5    - utter_ask_location
6  * inform
7    - utter_ask_location
8  * inform{"location": "Italy"}
9    - slot{"location": "Italy"}
10   - action_weather
11   - slot{"location": "Italy"}
12 * inform
13   - utter_ask_location
14 * inform{"location": "sri"}
15   - slot{"location": "sri"}
16   - action_weather
17   - slot{"location": "sri"}
18 * inform
19   - action_weather
20   - slot{"location": "sri"}
21 * inform{"location": "Berlin"}
22   - slot{"location": "Berlin"}
23   - action_weather
24   - slot{"location": "Berlin"}
25 * inform{"location": "SriLanka"}
26   - slot{"location": "SriLanka"}
27   - action_weather
28 * inform
29   - action_weather
30 * inform
31   - action_weather
32 * inform{"location": "Berlin"}
33   - slot{"location": "Berlin"}
```

```
Epoch 290/300
17/17 [=====] - 0s - loss: 0.2260 - acc: 1.0000 - val_loss: 1.1463 - val_acc: 0.4000
Epoch 291/300
17/17 [=====] - 0s - loss: 0.2085 - acc: 1.0000 - val_loss: 1.1456 - val_acc: 0.4000
Epoch 292/300
17/17 [=====] - 0s - loss: 0.2713 - acc: 0.8824 - val_loss: 1.1695 - val_acc: 0.4000
Epoch 293/300
17/17 [=====] - 0s - loss: 0.2658 - acc: 1.0000 - val_loss: 1.1342 - val_acc: 0.4000
Epoch 294/300
17/17 [=====] - 0s - loss: 0.1956 - acc: 1.0000 - val_loss: 1.1274 - val_acc: 0.4000
Epoch 295/300
17/17 [=====] - 0s - loss: 0.2549 - acc: 1.0000 - val_loss: 1.1084 - val_acc: 0.6000
Epoch 296/300
17/17 [=====] - 0s - loss: 0.3035 - acc: 0.9412 - val_loss: 1.0882 - val_acc: 0.6000
Epoch 297/300
17/17 [=====] - 0s - loss: 0.2846 - acc: 0.9412 - val_loss: 1.0858 - val_acc: 0.6000
Epoch 298/300
17/17 [=====] - 0s - loss: 0.2904 - acc: 0.9412 - val_loss: 1.0571 - val_acc: 0.6000
Epoch 299/300
17/17 [=====] - 0s - loss: 0.2050 - acc: 1.0000 - val_loss: 1.0549 - val_acc: 0.6000
Epoch 300/300
17/17 [=====] - 0s - loss: 0.2746 - acc: 0.9412 - val_loss: 1.0542 - val_acc: 0.6000
Bot loaded. Type a message and press enter :
Hello
Hello! How can I help?
Tell me the weather
In what location?
In Sweden
It is currently light snow in Stockholm at the moment. The temperature is -4.8 degrees, the humidity is 93% and the wind speed is 11.9 mph.
Goodbye
Talk to you later.
```

```
import logging

from builtins import str
from flask import Blueprint, request, jsonify

from rasa_core.channels.channel import UserMessage, OutputChannel
from rasa_core.channels.rest import HttpInputComponent

logger = logging.getLogger(__name__)

class SlackBot(OutputChannel):

class SlackInput(HttpInputComponent):
    def __init__(self, slack_dev_token, slack_verification_token, slack_client, debug_mode):
        self.slack_dev_token = slack_dev_token
        self.debug_mode = debug_mode
        self.slack_client = slack_client
        self.slack_verification_token = slack_verification_token

    def blueprint(self, on_new_message):
```




☆ frank

Messages About



Q Search



Today



frank APP 11:06 PM

Hello! How can I help?



Justina Petraityte 11:06 PM

What's the weahter



frank APP 11:06 PM

In what location?



Justina Petraityte 11:06 PM

In Vilnius



frank APP 11:06 PM

It is currently Light snow in Vilnius at the moment. The temperature is -8.0 degrees, the humidity is 79% and the wind speed is 11.9 mph.



Justina Petraityte 11:06 PM

Great, goodbye



frank APP 11:06 PM

Bye bye :(



Message @frank

I



What is our epic challenge???



**“Welcome to the new
world, where digital
aliens are man’s new
best friend”**

-Idowu Adeleke

Thank You !



@Idowu_adelek



idowuadeleke