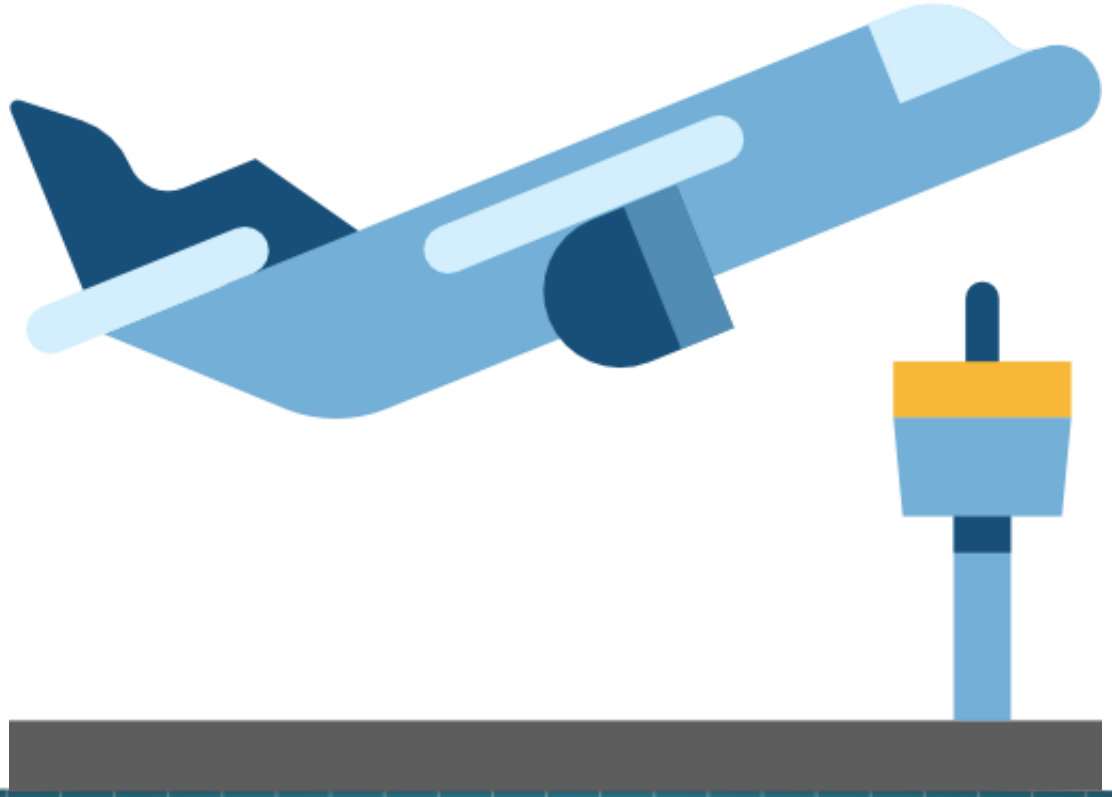
Build a Chatbot with Rasa (Natural Language Processing)

Presentation by Mr Namgyal BRISSON

Let’s take off

- RASA Documentation
- Chatito: Training & Testing Data Generation
- Let’s dig in with an Example on Github



Minimal Airline Companies Database

- **The main idea is to imitate in a minimalist way a database of airline companies.**
- **Insertion of a destination city with the associated airline ticket prices.**
- **Following cities will be inserted:**
  - **Taiwan**
  - **Thaïlande**
  - **California**
  - **Paris**
  - **Mauritius**
  - **Mumbai**
  - **Toulouse**
  - **Lyon**
  - **Berlin**
  - **Rome**
  - **Madrid**
  - **Johannesburg**
  - **Antananarivo**
  - **New Delhi**
  - **New-York**
  - **Atlanta**
  - **San Francisco**
  - **Port Louis**

Create Sample Flights

PK	DESTINATION	PRICE (€)	COMPANY
1	taiwan	500	air taiwan
2	thailand	700	air asia
3	california	600	air america
4	paris	1200	air france
5	paris	990	air italia
6	paris	1080	lufthansa
7	mauritius	1190	air france
8	mauritius	1280	air mauritius
9	india	599	air india
…	…	…	…

SQL CREATE script

```
create table if not exists flights
    (
        pk INTEGER not null
            constraint flights_pk
                primary key autoincrement,
        destination TEXT not null,
        price INTEGER not null,
        company TEXT not null
    );

create unique index if not exists
flights_pk_uindex on flights (pk);
```

RASA Training Sets

```
import ./common.chatito

# Ways to request a restaurant within a location (using probability operator)
# NOTE: 60% of the examples should come from the first sentence, and 40% from the second

%[findFlight]('training': '100', 'testing': '100')
    *[60%] ~[hi?] ~[please?] ~[find?] ~[flight] ~[from] @[departure] ~[departure?] ~[to] @[destination] ~[destination?] ~[thanks?]
    *[40%] ~[flights] ~[destination to] @[destination] ~[destination?]

@[departure]
    ~[new york]
    ~[san francisco]
    ~[atlanta]
    ~[port louis]
    ~[paris]
    ~[thaiwan]
    ~[india]

@[destination]
    ~[new york]
    ~[san francisco]
    ~[atlanta]
    ~[port louis]
    ~[paris]
    ~[thaiwan]
    ~[india]

~[find]
    find
    look for
    search for
    help me find

~[destination to]
    destination to
    in the area of
    near by
    go to
    land to

~[flights]
    flight
    plane
```

Create the Bot

- Create database using SQLite (sqlite3)
  - Insert flights rows from fixture
- Training NLP parser with Chatito training data set
- Reply with database records matching NLP parser entities
- Testing with samples

To go further

- Create a multi-domain bot
  - Create a training dataset for each domain you want to process (flight, restaurant, hotels, events, ...)
  - Aggregate them all into a Chatbot
  - Parralelize User’s question to each bot by implementing a multi-processes application
  - Eventually, reply to the User for each domain from which you get replies

Time to Land...

Thanks for attending :)