Project Report

UNSW SYDNEY

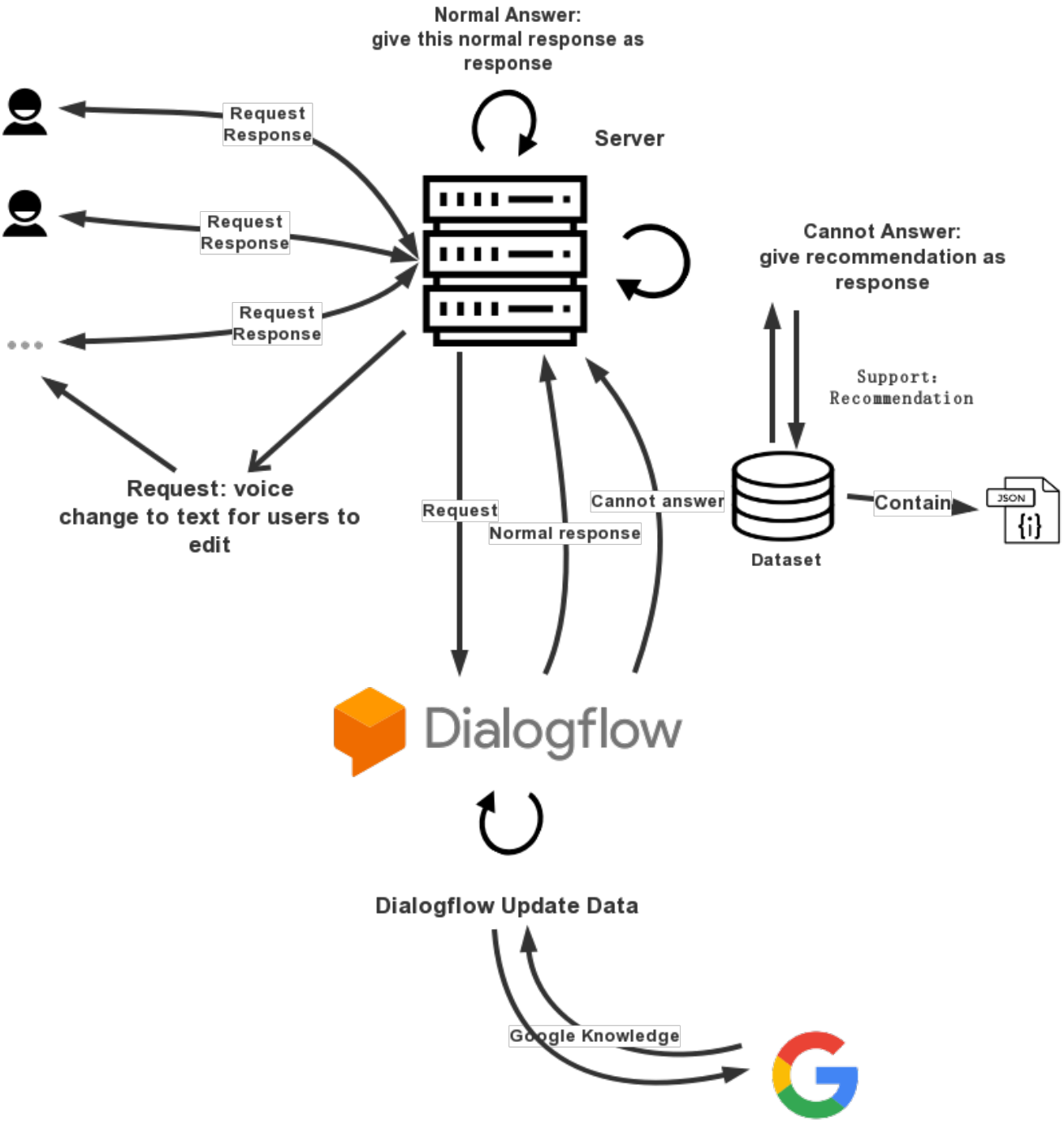
Chat Bot
X_O_BOT

Group member:
Man YI
Xiao LI
Mofei WANG

Abstract: A chat bot web server, get answer from your questions.

Overview

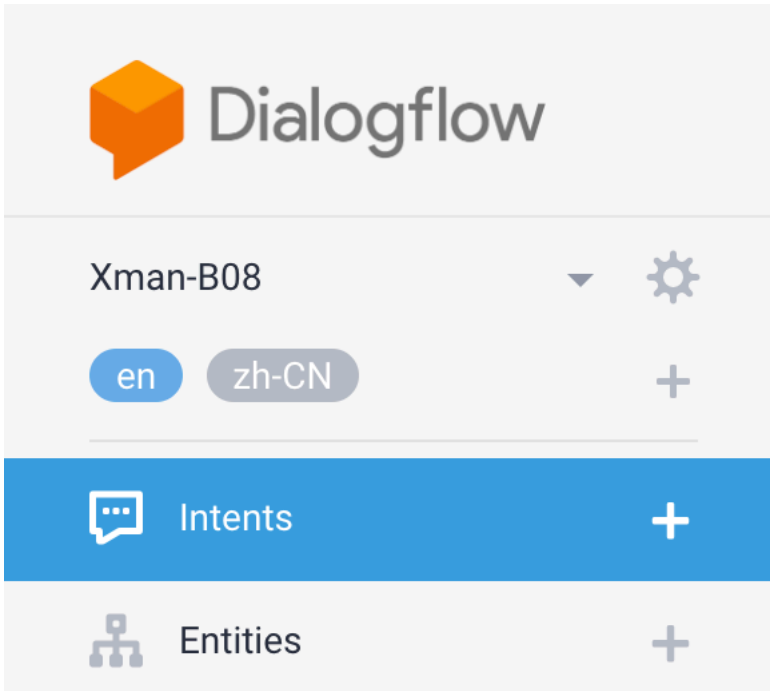
Architecture / Design



Data collection & integration:

1. Data are saved in API (dialogflow.com).[1] Because there is no legal data available, our group are inserted by out group member.
Dialogflow has a very clear data management, contains entities and intents management in that webpage.

2. Dialogflow has a ‘client access token’ for user to connect API from users application. [2]



API KEYS (V1)

Client access token	38396cfaee2644e783c84d6667e829b0
Developer access token	2e3000e397d44b44aa93b0822a3465f5

Web Site Server:

1. Based on python3 Django web server, which is easy to manage. Major python files: settings.py, urls.py, admin.py, apps.py, models.py, views.py.
2. Web server also has some support function, web pages in templates, script files in static, data (in json file) in dataset folder.

Information extractor and Analysis:

For give recommendation for users, export dataset from dialogflow.
Intents and entires are saved in dataset for providing recommendation for users.

Descriptions of functionalities

Functionalities list with descriptions:

1. Basic functionalities:

Basic functionalities this application provided:

- Chat with x_o_bot
  - Text request
- Receive answer from x_o_bot
  - Text answer
- Search in google

2. Additional functionalities:

Extra functions including:
When asking questions:

Voice input is implement

When receiving answer form x_o_bot:

Website answer: could jump to target website
Bad answer: give recommendation for users

Implementation Challenges

1. User demand analysis:

This challenge is to analyze the users' demand, which is necessary to improve application qualities. Although Dialogflow has a function of collection the data input by the user, divided these questions into different kinds if hard. Reasons of the bad answer receiving: Spell mistake, questions cannot be answer and empty input.

To solve these problems, we implemented a algorithm to provide users a recommendation for using.

2. Data Maintenance:

This challenge is when data maintenance, we found a problem is the more the question Dialogflow has, the more difficult the right answer could be received.

To solve these problems, we did not improve the questions more complexity, more long. We do a lot on entities in intents, more entities could make system knowledge library recognize questions more accurate.

Entities have more synonym:

3. Date Analysis:

For analyzing the data for recommendation, export from dialogflow which contains multiple json files about entities and intents. How to get a suitable recommendation for users is a challenge problem.

To solved this problem, using Levenshtein distance[3] algorithm to calculate questions and questions in dataset, compare these questions and get each pair questions' similarity. Then rank these similarities and get suitable recommendation. [4]

$$\operatorname{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \operatorname{lev}_{a,b}(i-1,j) + 1 \\ \operatorname{lev}_{a,b}(i,j-1) + 1 \\ \operatorname{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

4. User Interface:

User interface challenge is how to make user easier to separate the user words and x_o_bot words and make browser windows more convenient for user focus the current windows when types questions.

To solve these problems, we wrote css files to separate chat areas one is white, the other one is grey. Additionally, we set the input area always in the bottom of the window.

```
<div id="scrolldIV" style="overflow: scroll; height: 100%; width: 100%; border: 1px solid #999;">
```

```
var div = document.getElementById('scrolldIV');
div.scrollTop = div.scrollHeight;
```

5. Voice Implement

Voice implement challenge could be solve by the SpeechRecognition [5] package in python library.

A problem in this project process is the recognize of voice perform not very well because of multiple questions, so we take the recognized text out and could be edited by users.

User Documentation / Manual

X_o_bot is a web client to chat based on Dialogflow.

Requirements And Start

Starting server needs some dependency packages, in project a requirements.txt will list all the required packages in it.

python3 -m pip install -r requirements.txt

Type command above to install.

python3 manage.py runserver

Type command above to start server.

For allow other devices to have access can also type command and in settings.py to add your host into ALLOWED_HOSTS.

python3 manage.py runserver 0.0.0.0:8000

Configuring Dialogflow

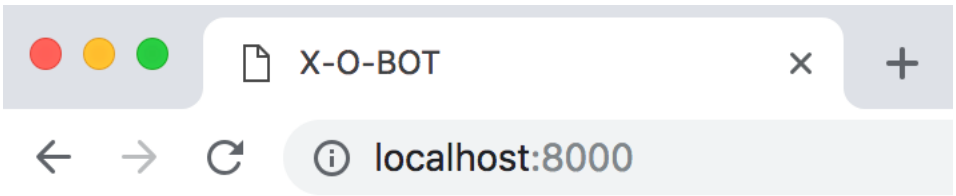
For Dialogflow settings in settings.py, we build a

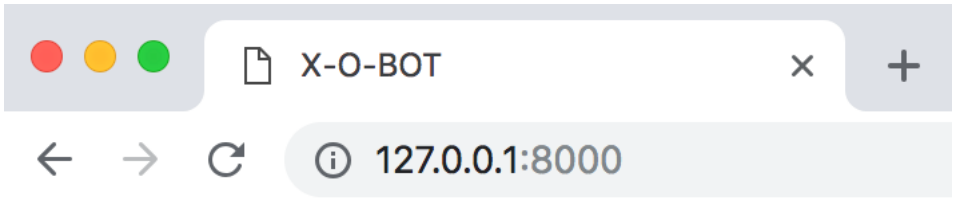
```
# DIALOGFLOW client_access_token
DIALOGFLOW = {
    'client_access_token': '38396cfaee2644e783c84d6667e829b0',
}
```

The client_access_token could be change to users' dataset, project could adapt other dataset. [mentioned in Data collection & integration]

Web Page

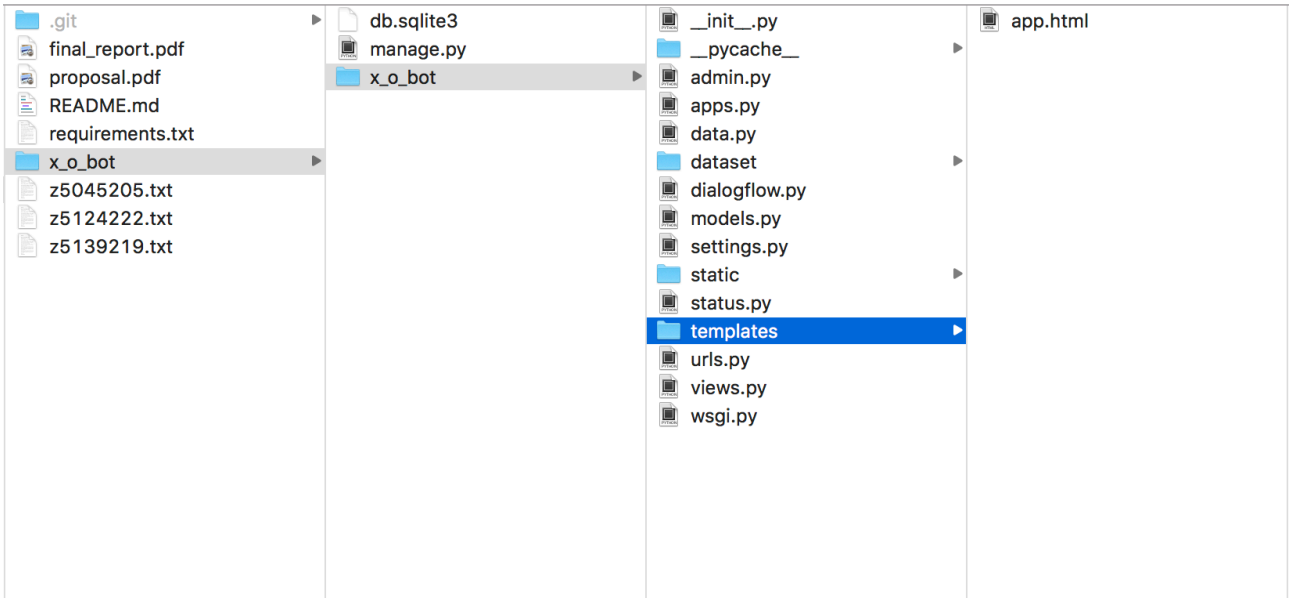
Url:





Or you can modify the ALLOW_HOSTS for adapt the url you require.

Django Web Server

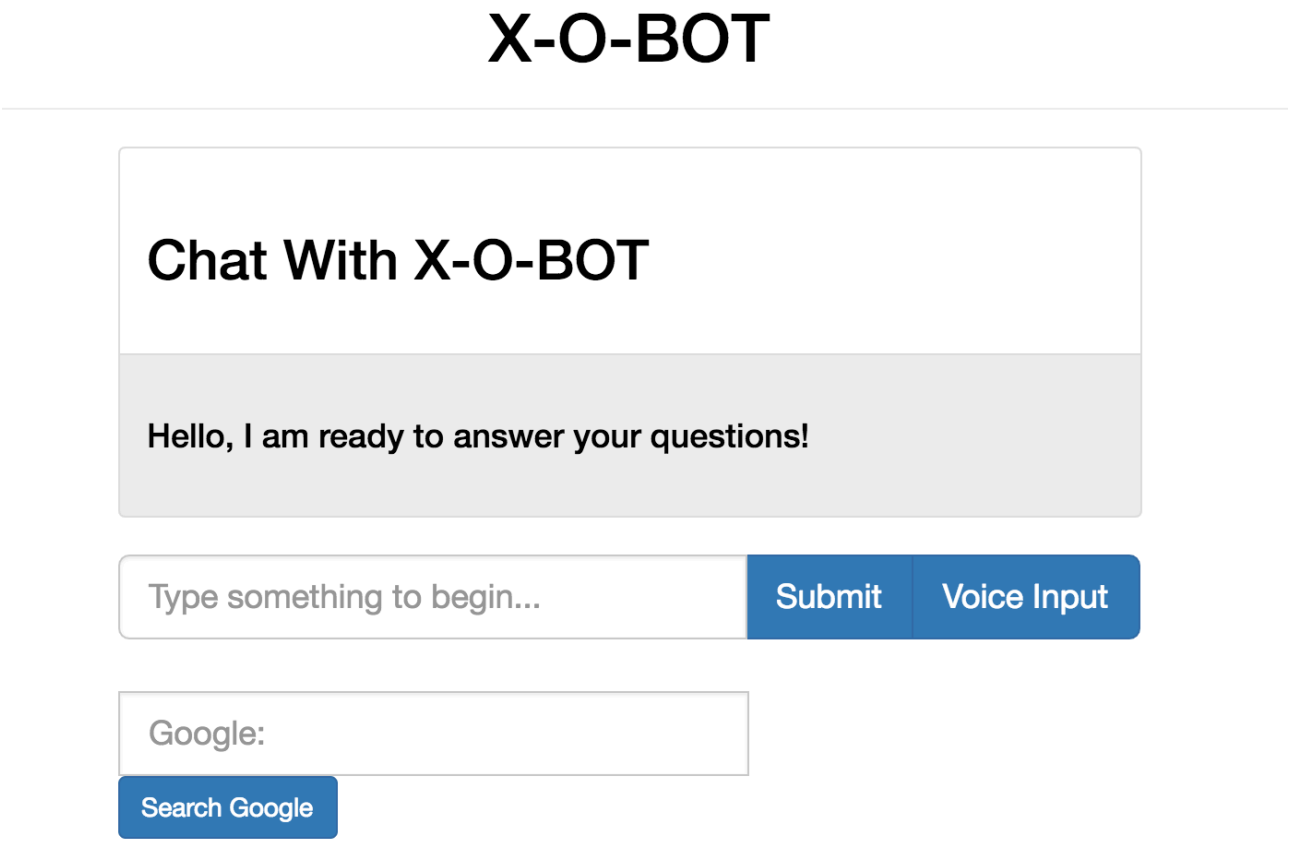


- mange.py: main program for start web server
- settings.py: main setting values saved in
- admin.py, model.py: for database management, but using data in json files in dataset folder only.
- Views.py, urls.py: main function saved in views.py and urls.py saved valid url in this web server.
- dialogflow.py, status.py: saved functions in connect with Dialogflow.
- data.py: handle data in dataset and provided recommendation

In templates folder, saved the html files and static saved css files and js files for app.html. In dataset folder, saved json files are in unified specification which export from Dialogflow.

Main page area:

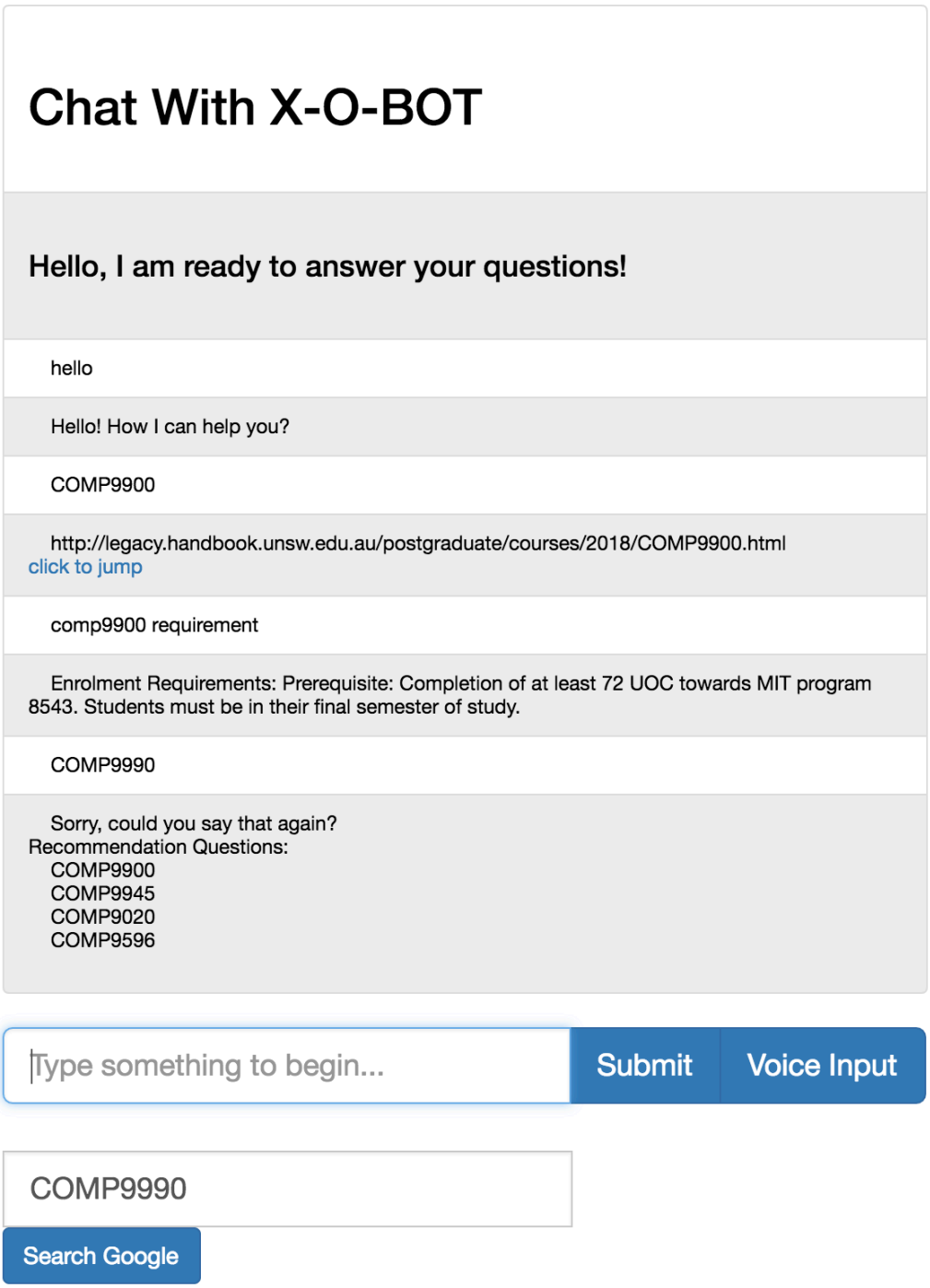
Type questions in the blank area to start conversation.



The area in bottle of the page, it could display the question you want. If you think recommendation is not suitable, you can also search in Google.

Voice input button is for voice input, you can use your microphone to recognize your voice.

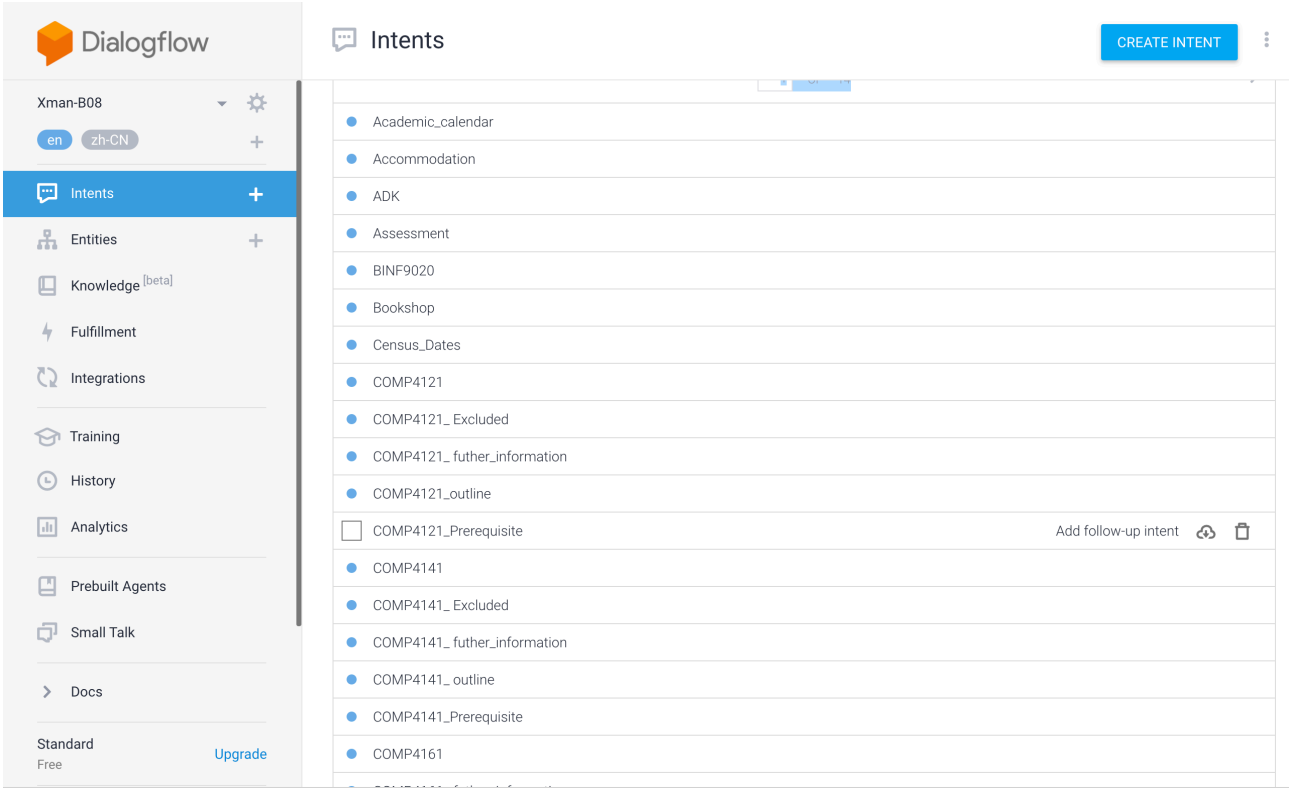
X-O-BOT



Grey area is chat bot's words and white is users says.

1. Normal answer is text.
2. Website answer has a 'click to jump' button to jump to target website.
3. Recommendation will list questions may be you want.

Dialogflow



Manage questions and answers, manage entities (key words), this project also can be connected to other Dialogflow dataset.

Conclusion

Advantage:
1. Easy to use and have a simple webpage to understand.
2. Dataset contains questions and answers in Dialogflow is easy to manage.

Improvement:
1. Recommendation could become more genetic for providing question
2. Dialogflow could change depend on what area are this project used in.

References

[1] http://www.dialogflow.com/
[2] https://dialogflow.com/docs/reference/agent
[3] https://en.wikipedia.org/wiki/Levenshtein_distance
[4] https://www.cuelogic.com/blog/the-levenshtein-algorithm
[5] https://pypi.org/project/SpeechRecognition/