**AIParrot**

**Project overview**

AIParrot is an intelligent conversational AI that uses machine leaving to generate responses to a given question. For a live demo, you can visit the website link().

**Technology stack**

As AIParrot we use Python with Tensorflow and Keras as a deep learning framework of choice for machine learning tasks. For web interface and API, we use Python, Flask, and Bootstrap

**Get started**

To get started with AIParrot only pre-requirement is Python after installing python you can run “pip install requirements.txt“ to install all dependencies AIParrot needs to run.

**General understanding of project**

Where to find code

AIParrot is an open-source machine learning project and code is hosted on GitHub you can check out the code via the link(https://github.com/YigitGunduc/parrot)

Dir structure

AIParrot contains lots of nested files in this section I will explain some of the major files

**AI**: AI folder contains two folders one named seq2seq and the one called “seq2seqwithembedding” seq2seq file contain code for bare-bones se2seq model and the “seq2seqwithembedding” folder contains a seq2seq model with embedding layers and GloVe vectors

**Docs**: contain documentation of AIParrot

**Platform**: platform folder contains the web-app and Twitter support folder. Actually, these folders are pretty self-explanatory web-app folder contains both the web app and the API and the twitter-support folder contain twitter API

**Seq2seq Models**

Seq2Seq model takes a sequence in this case sentences and outputs another sequence. The encoder captures the context of the input sequence in the form of a hidden state vector and sends it to the decoder, which then produces the output sequence. In our case, we have two of them. The first one is bare-bones seq2seq and the other one uses embedding and GloVe vectors to increase the learning rate. For more information about GloVe vectors, you can check out Stanford’s page.

(link: https://nlp.stanford.edu/projects/glove/)

**Generate response from a pre-trained model**

There are a couple of ways to generate a response from AIParrot I will explain all of them in the section below

Web app

Webapp is the easiest way to ask a question to the AIParrot

API

If you want to integrate AIParrot with your project API is just for you there is an easy to use API for example request you can check the “platform/ExampleApiRequest.py” file or the code below

import requests

response = requests.get("URL")

#raw response

print(response.json())

#cleaned up response

print(response.json()["response"])

**Train your own model**

To train your own model navigate to the desired folder(“seq2seqwithembedding” or seq2seq) there is no right model but I highly encourage you to train seq2seqwithembedding if you have resources. After you choose your model you can run the train.py file and it will generate model files for you. It takes the data batch by batch and all the batches contain 10000 lines our dataset has 320000 lines so it will generate around 30 “.h5” extension files after training has done you better to go through end eval at leas each fifth tenth, fifteenth … the last model to find which batch fits your needs best

**Eval your model**

Firstly give the exact path to your own model to the eval.py after that you can run eval.py to generate responses from it. Eval.py will pick questions from the dataset if you want to give it specific questions and see how it behaves. You can run the web app in your localhost and give it your own model to see how it behaves.

**Contributions**

Contributions are what make the open-source community such an amazing place to learn, inspire, and create. Any contributions you make are greatly appreciated.

1. Fork the Project
2. Create your Feature Branch (git checkout -b feature/AmazingFeature)
3. Commit your changes (git commit -m 'Add some AmazingFeature')
4. Push to the Branch (git push origin feature/AmazingFeature)
5. Open a Pull Request

**License**

This project is licensed under the MIT License (see the LICENSE file for details).