**Text Based Chatbot**

**Source Code Details:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Serial No. | File Name | Type of Language | Type of File (or File Extension) | Functionality | Dependency |
| 1 | DSAI\_Model\_Training.ipynb | Python | .ipynb  Implemented using OOP concepts | Used to train the model and save it as DSAI\_Chatbot\_Model.h5. | Input:  DSAI\_intents.json,  DSAI\_words.pkl,  DSAI\_classes.pkl  Output: DSAI\_chatbot\_model.h5 |
| 2 | DSAI\_Web\_Chatbot.py | Python | .py | Used to create text based chatbot in Web Interface | Input: DSAI\_Chatbot\_Model.h5, DSAI\_Intents.json, DSAI\_Words.pkl, DSAI\_Classes.pkl, DSAI\_Index.html  Dependency:  DSAI\_Utility.py |
| 3 | DSAI\_Desktop\_Chatbot.py | Python | .py | Used to create text based chatbot as Desktop application | Input: DSAI\_Chatbot\_Model.h5, DSAI\_Intents.json, DSAI\_Words.pkl, DSAI\_Classes.pkl  Dependency:  DSAI\_Utility.py |
| 4 | DSAI\_Utility.py | Python | .py | Contains helper function to be invoked in DSAI\_Web\_Chatbot.py and DSAI\_Desktop\_Chatbot.py |  |
| 5 | jquery.scrollbar.js | JavaScript | .js | Used to define interaction behavior of scrollbar in web interface | - |
|  | DSAI\_index.html | Html | .html | Used to define the web interaction page | - |

**Run Instructions:**

|  |  |  |  |
| --- | --- | --- | --- |
| Step Number | Run Command | Run Dependency | Run Instruction |
| 1 | Run DSAI\_Model\_Training.ipynb file |  | This creates pickle files and trained model |
| 2 | Run ‘python DSAI\_Chatbot\_Flask.py’ |  | This runs text based chatbot in web interface |
| 3 | Run ‘python DSAI\_Chatgui.py’ |  | This runs text based chatbot as desktop application |

**Source Code Steps:**

1. DSAI\_Model\_Training.ipynb
   1. Import dataset DSAI\_Intents.json
   2. Store classes and words in pickle format
   3. Prepare training data and testing data
   4. Create model
   5. Compile model
   6. Train and save the model as DSAI\_chatbot\_model.h5
2. DSAI\_Web\_Chatbot.py
   1. Get user message
   2. Predict class of user message by invoking DSAI\_Utility.py
   3. Get response for the predicted class by invoking DSAI\_Utility.py
   4. Display the chatbot response
3. DSAI\_Desktop\_Chatbot.py
   1. Create GUI for Desktop application
   2. Get user message
   3. Predict class of user message by invoking DSAI\_Utility.py
   4. Get response for the predicted class by invoking DSAI\_Utility.py
   5. Display the chatbot response
4. jquery.scrollbar.js

JavaScript code for customizable scrollbar effects

1. DSAI\_index.html

HTML design code for chatbot