E-COMMERCE CHATBOT

USING LSTM

MURALI KRISHNAN S
REG.NO: 422521104305
r.s.murali0628@gmail.com
University College Of Engineering Villlupuram

AGENDA

01 PROBLEM STATEMENT
02 PROPOSED SYSTEM/SOLUTION
03 SYSTEM RECQUIRMENTS
04 ALGORITHM AND DEPLOYMENT
05 WHO ARE THE END USERS
06 RESULT
07 CONCLUSION
08 REFERENCES

PROBLEM STATEMENT

Addressing the challenges of eCommerce customer support, we've introduced a cutting-edge automated chatbot powered by deep learning methodologies. This innovation revolutionizes customer service by delivering instantaneous, uniform, and budget-friendly support solutions. Our objective is to elevate user satisfaction while optimizing the efficiency of our support operations.

PROPOSED SYSTEM/SOLUTION

DEEP LEARNING MODEL

Utilize LSTM networks to process and understand the sequential nature of natural language data effectively. This involves capturing long-term dependencies insequences, facilitating the model's understanding and classification of user queries.

CHATBOT ARCHITECTURE DEDIGN

Design a chatbot architecture using LSTM networks for natural language processing, complemented by text cleaning, tokenization, and intent classification to deliver precise and relevant responses.

PROPOSED SYSTEM/SOLUTION

TEXT PROCESSING

Implement cleaning and organizing of raw text data to ensure clean input for the model. This involves converting text to lowercase, lemmatizing words, and removing stopwords to prepare the text data for model input.

HYPER PARAMETER TUNING

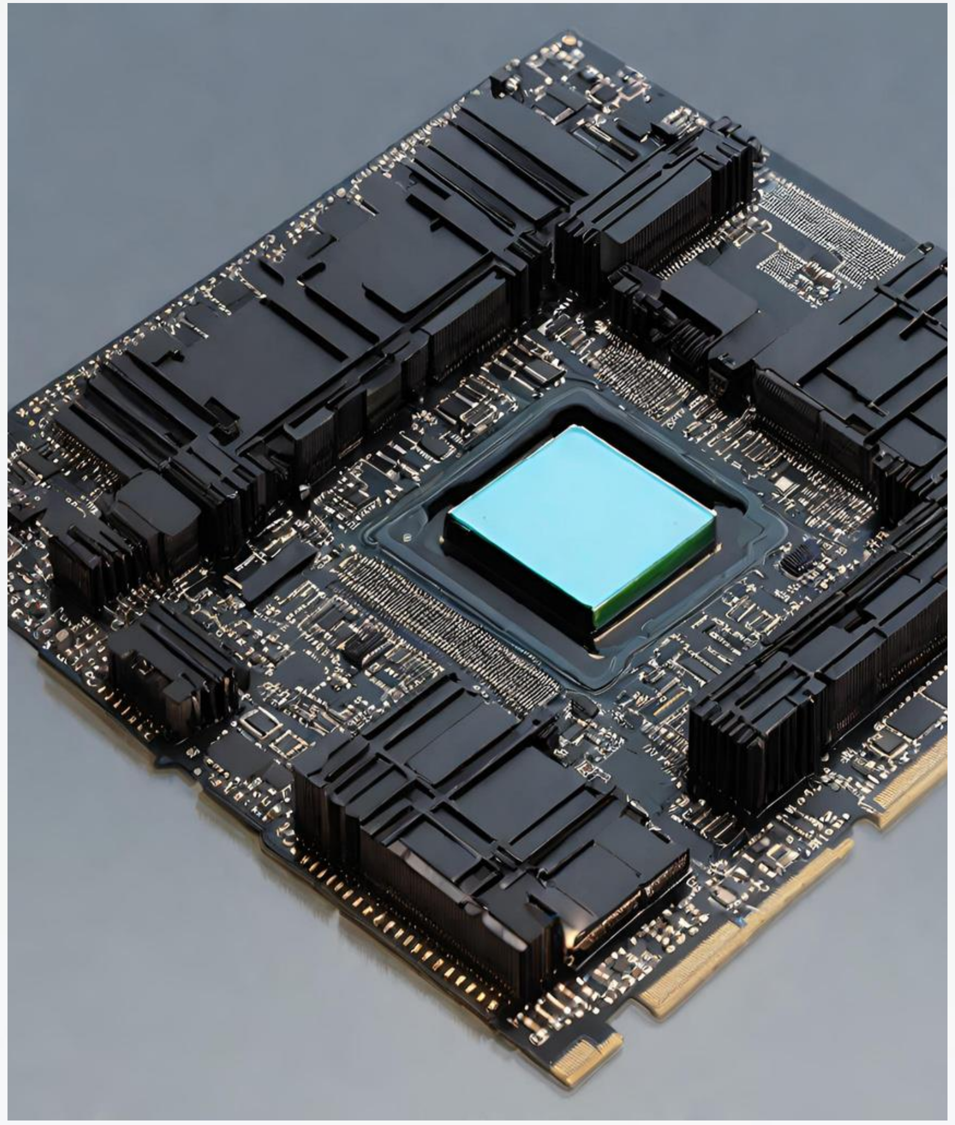
Utilize Keras Tuner to automate the optimization of model parameters to improve performance and accuracy. This involves systematically exploring the hyperparameter space to find the optimal configuration for the model.

TOKENIZATION AND PADDING

Implement tokenization and padding techniques to convert text data into numerical sequences and ensure a consistent input length for the model. This involves tokenizing textual data to convert words into numerical form and applying padding to ensure a uniform sequence length.

SYSTEM APPROACH

1.HARDWARE

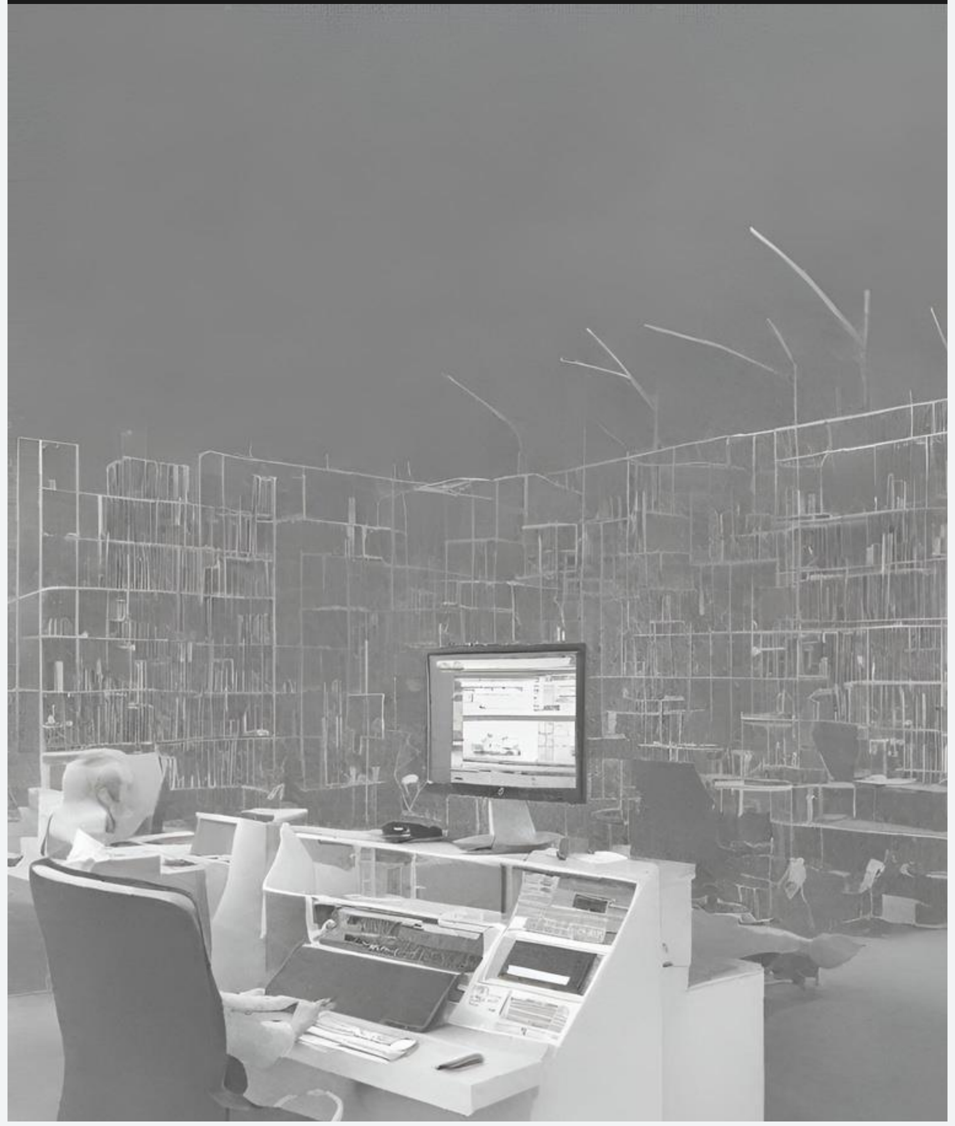


SYSTEM RECQUIRMENTS

- CPU: A multicore processor to handle computational tasks efficiently,
- supporting the intensive processing demands of deep learning algorithms.
- RAM: Minimum of 8GB RAM to ensure smooth performance during model
- training and inference, accommodating the memory requirements of large
- datasets and complex computations.
- Internet Connection: Stable and high-speed internet connection to facilitate
- data retrieval, model updates, and seamless interaction with external services
- and APIs.

SYSTEM APPROACH

2.SOFTWARE



SYSTEM RECQUIRMENTS

- Python: Programming language used for developing the chatbot application.
- TensorFlow/Keras: Deep learning libraries used for building and training the
- LSTM model.
- NLTK: Natural Language Toolkit used for text preprocessing and tokenization.
- scikit-learn: Library for various machine learning tasks like data preprocessing
- and model evaluation.
- Jupyter Notebook: Interactive computing environment used for prototyping,
- data analysis, and model development.
- Optional: Django for web-based interface (if applicable).

ALGORITHM AND DEPLOYMENT

01

DATA PREPARATION

- Acquired the dataset from Kaggle to serve as the foundation for training and evaluating the chatbot model .
- Gathered the requisite data from Kaggle to establish the groundwork for both training and testing phases of the chatbot model.

02

TEXT PROCESSING

- Employed NLTK for text processing to clean and organize raw text data.
- Eliminated punctuation and stop words to ensure the input was devoid of irrelevant elements.
- Utilized lemmatization to reduce words to their base forms for enhanced analysis.

03

DEEP LEARNING MODEL:NETWORK

- Implemented LSTM networks to handle natural language data processing and comprehension.
- Configured the model architecture with two LSTM layers, each comprising 110 units.
- Employed the Adam optimizer, setting the learning rate to 0.01 for efficient model training.

ALGORITHM AND DEPLOYMENT

04

HYPERPARAMETER TUNING MODELS:TUNER

- Employed Keras Tuner for automating model parameter optimization.
- Varied the number of units in LSTM layers from 50 to 150.
- Investigated the range of dense layers in the model from 1 to 20.
- Fine-tuned learning rates across 0.01, 0.001, and 0.0001 for enhanced performance.

05

TOKENIZATION AND PADDING

- Utilized Keras Tokenizer to tokenize the text data, transforming words into numerical sequences.
- Implemented padding to maintain a consistent sequence length of 200 for model input.
- Configured the vocabulary size to 2000 to encompass the most prevalent words in the dataset.
- Incorporated an out-of-vocabulary token (OOV) mechanism to manage words not present in the vocabulary during tokenization.

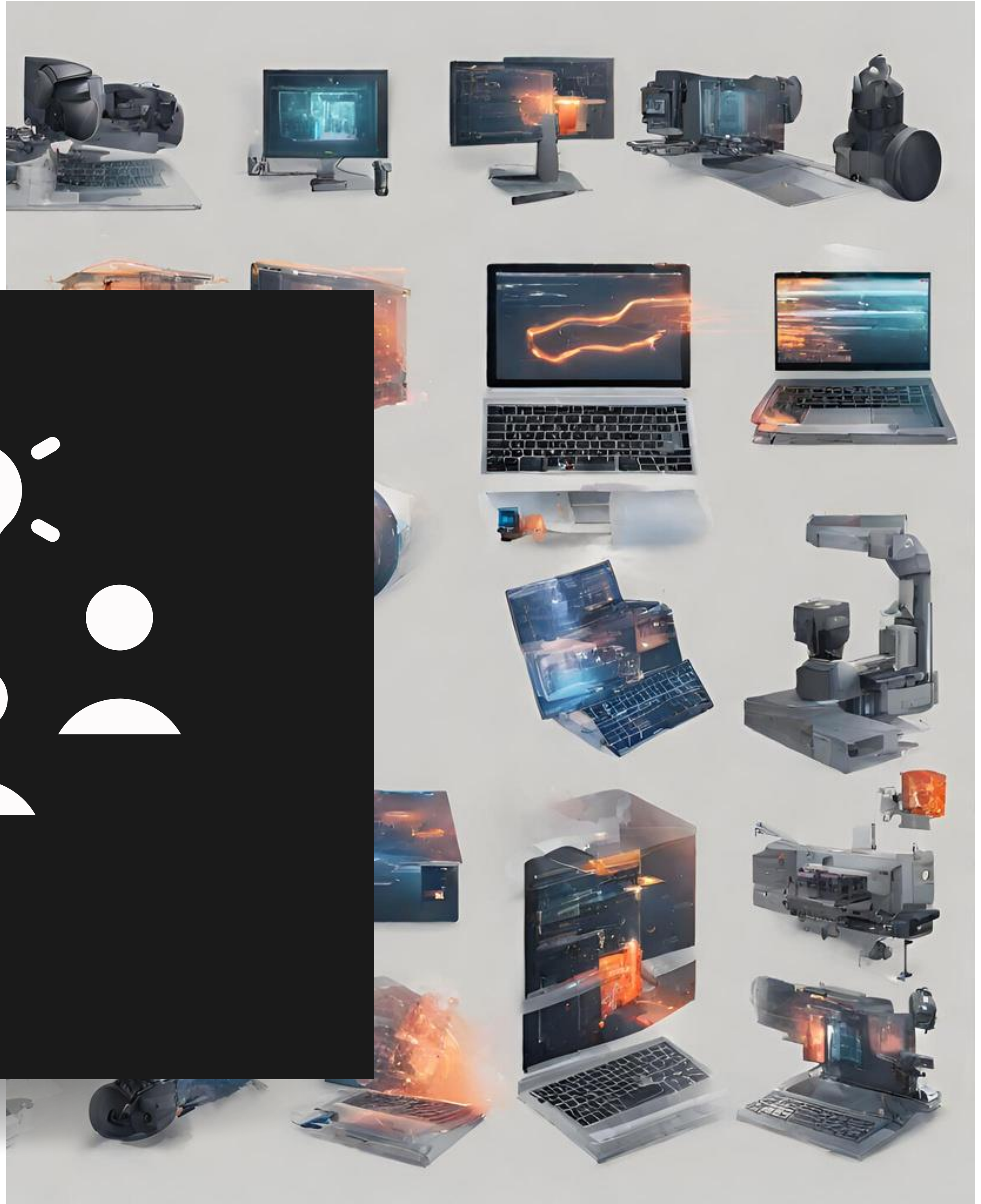
DEPLOYMENT

1. JUPYTER NOTEBOOK

- The chatbot code is hosted on GitHub by implementing the main code directly on a Jupyter Notebook. This approach allows for easy viewing and execution of the code.

2. DJANGO WEB APPLICATION

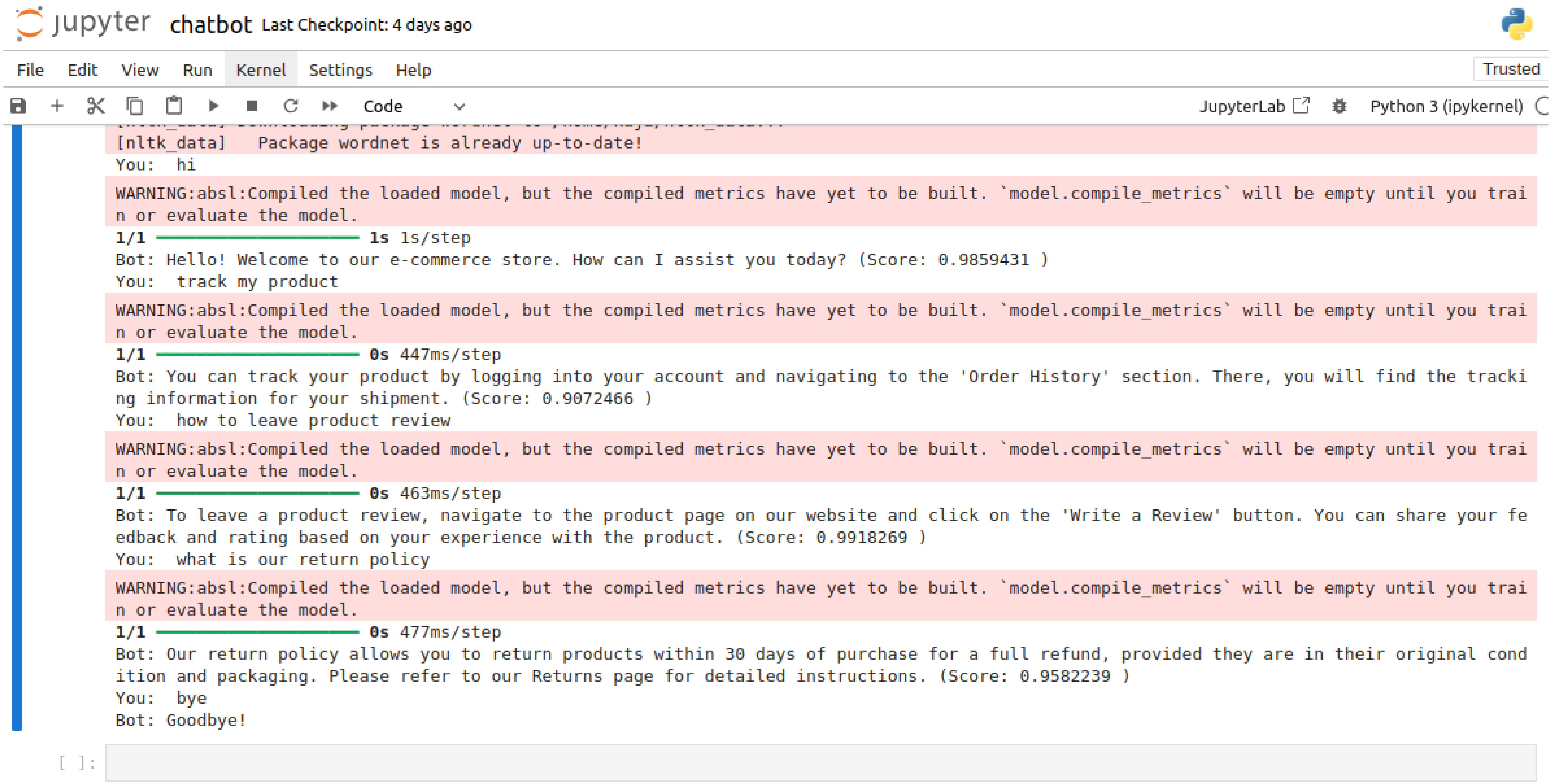
- The chatbot is presented as a user-friendly web-based interface using the Django framework and hosted on GitHub. This method offers a seamless user experience and makes the chatbot accessible via a web browser.



WHO ARE THE END USERS?

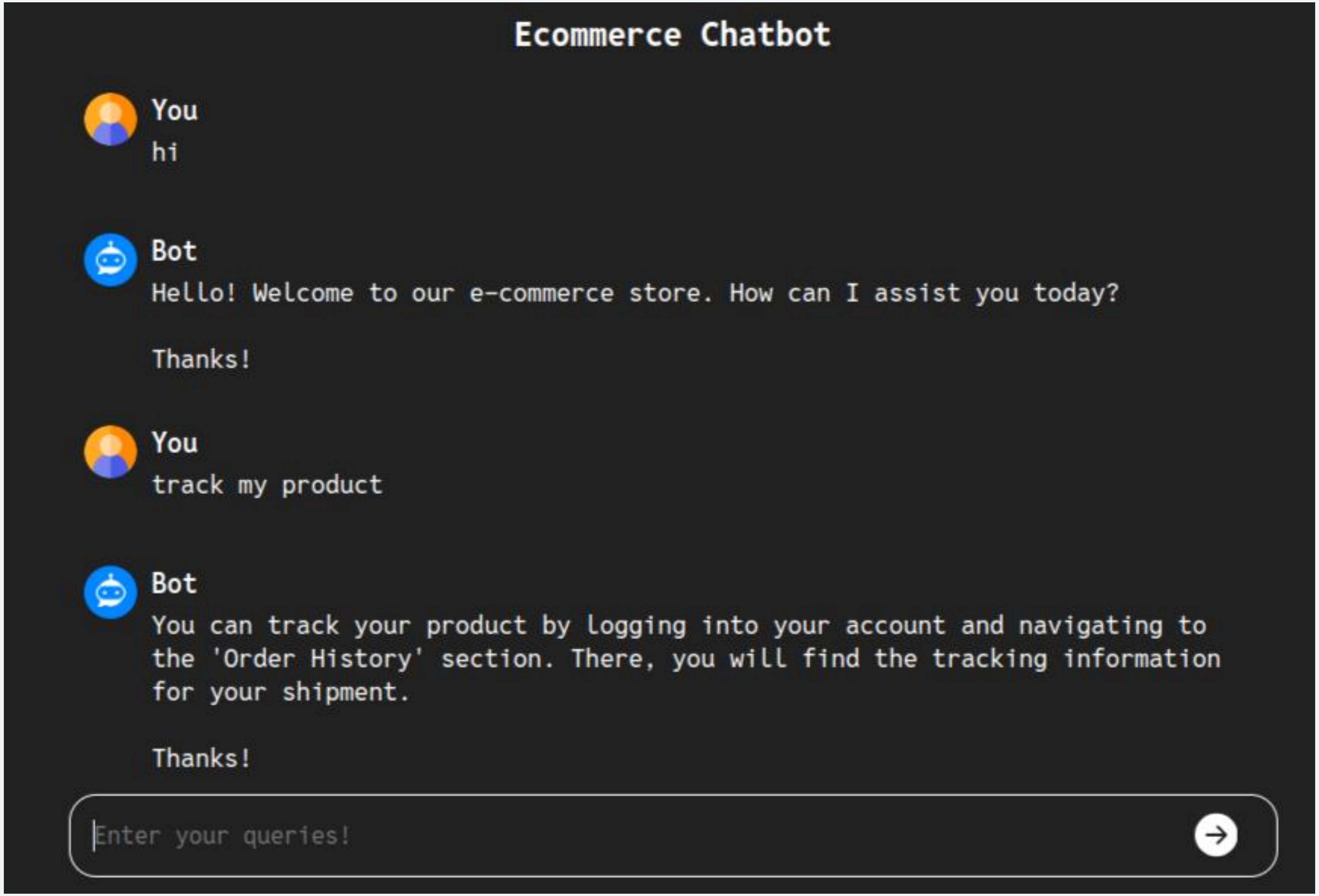
- **Online Shoppers:** Individuals seeking product information, recommendations, or assistance during their shopping journey.
- **Customer Support Teams:** Team members tasked with addressing customer inquiries and delivering prompt, precise responses.
- **Website Visitors:** Potential customers browsing the eCommerce platform, possibly in need of guidance or information.
- **Technical Support:** IT professionals or developers responsible for maintaining and optimizing the chatbot's functionality and performance.

RESULT: EXECUTED ON JUPYTER



```
[nltk_data]   Package wordnet is already up-to-date!
You:  hi
WARNING:absl:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until you trai
n or evaluate the model.
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step
Bot: Hello! Welcome to our e-commerce store. How can I assist you today? (Score: 0.9859431 )
You:  track my product
WARNING:absl:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until you trai
n or evaluate the model.
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 447ms/step
Bot: You can track your product by logging into your account and navigating to the 'Order History' section. There, you will find the tracki
ng information for your shipment. (Score: 0.9072466 )
You:  how to leave product review
WARNING:absl:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until you trai
n or evaluate the model.
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 463ms/step
Bot: To leave a product review, navigate to the product page on our website and click on the 'Write a Review' button. You can share your fe
edback and rating based on your experience with the product. (Score: 0.9918269 )
You:  what is our return policy
WARNING:absl:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until you trai
n or evaluate the model.
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 477ms/step
Bot: Our return policy allows you to return products within 30 days of purchase for a full refund, provided they are in their original cond
ition and packaging. Please refer to our Returns page for detailed instructions. (Score: 0.9582239 )
You:  bye
Bot: Goodbye!
```

RESULT: IMPLEMENTED ON DJANGO



Ecommerce Chatbot

You

hi

Bot

Hello! Welcome to our e-commerce store. How can I assist you today?

Thanks!

You

track my product

Bot

You can track your product by logging into your account and navigating to the 'Order History' section. There, you will find the tracking information for your shipment.

Thanks!

CONCLUSION

-Our eCommerce chatbot leverages advanced AI to deliver personalized customer support, enhancing the shopping experience and providing a competitive edge in the digital marketplace.

-As AI technologies continue to advance, our chatbot is poised to evolve and offer even more innovative solutions to meet the changing demands of the eCommerce industry.

REFERENCES

1.NumPy : https://numpy.org/

2.TensorFlow : https://www.tensorflow.org/

3.Keras : https://keras.io/

4.scikit-learn (sklearn) : https://scikit-learn.org/stable/

5.NLTK (Natural Language Toolkit) : https://www.nltk.org/

6.Keras Tuner : https://keras.io/keras_tuner/

7.Pickle (Python's standard library) : https://docs.python.org/3/library/pickle.html

8.Django : https://www.djangoproject.com/

Thank you!