A

MINI PROJECT REPORT

ON

OPEN AI CHATBOT

Submitted in partial fulfillment of the requirements

For the award of Degree of

BACHELOR OF ENGINEERING

IN

CSE (AI ML)

Submitted By

K. ROHITH NAYAK	**245321748031**
P. GOUTHAM VARMA	**245321748304**
T. ABHILASH	**245321748307**

Under the guidance

Of

Mr. G. SURENDHER

Assistant Professor

Department of CSE(AIML)

NEIL GOGTE INSTITUTE OF TECHNOLOGY

Kachavanisingaram, Hyderabad, Telangana 500058.

2024

NEIL GOGTE INSTITUTE OF TECHNOLOGY

A Unit of Keshav Memorial Technical Education (KMTES)

Approved by AICTE, New Delhi & Affiliated to Osmania University, Hyderabad

CERTIFICATE

This is to certify that the Mini project work entitled “**OPEN AI CHATBOT”** *is a bonafide work carried out by* **K. ROHITH NAYAK (245321748031), P. GOUTHAM VARMA (245321748304), T. ABHILASH (245321748307)** of III-year V semester **Bachelor of Engineering** *in CSE(***AIML)** *by Osmania University, Hyderabad during the academic year* **2023-2024** *is a record of bonafide work carried out by them. The results embodied in this report have not been submitted to any other University or Institution for the award of any degree*

Internal Guide

Mr. G. Surendher

Assistant Professor

Head of Department

Dr. T. Prem Chander

Associate Professor

External

NEIL GOGTE INSTITUTE OF TECHNOLOGY
A Unit of Keshav Memorial Technical Education (KMTES)
Approved by AICTE, New Delhi & Affiliated to Osmania University, Hyderabad

DECLARATION

I hereby declare that the Mini Project Report entitled, “**OPEN AI CHATBOT**” submitted for the B.E degree is entirely my work and all ideas and references have been duly acknowledged. It does not contain any work for the award of any other degree.

Date:

K. ROHITH NAYAK	**245321748031**
P. GOUTHAM VARMA	**245321748304**
T. ABHILASH	**245321748307**

ACKNOWLEDGEMENT

We are happy to express our deep sense of gratitude to the principal of the college **Dr. R. Shyam Sunder, Professor**, Neil Gogte Institute of Technology, for having provided us with adequate facilities to pursue our project.

We would like to thank**, Dr. T. Prem Chander**, **Head of the Department,** CSE(**AIML**), Neil Gogte Institute of Technology, for having provided the freedom to use all the facilities available in the department, especially the laboratories and the library.

We would also like to thank my internal guide **Mr. G. Surendher, Assistant Professor,** CSE**(AIML),** Neil Gogte Institute of Technology, for his technical guidance & constant encouragement.

We sincerely thank my seniors and all the teaching and non-teaching staff of the Department of Artificial Intelligence & Machine Learning for their timely suggestions, healthy criticism and motivation during this work.

Finally, We express my immense gratitude with pleasure to the other individuals who have either directly or indirectly contributed to our need at the right time for the development and success of this work.

ABSTRACT

OpenAI's chatbot, powered by the GPT-3.5 architecture, is a state-of-the-art natural language processing model designed for a wide range of applications. Leveraging deep learning techniques, the chatbot excels in generating human-like responses and engaging in meaningful conversations across various domains. With a knowledge cutoff in early 2022, the model has been trained on diverse datasets, enabling it to comprehend and generate contextually relevant text in response to user prompts.

This powerful chatbot exhibits versatility in tasks such as text completion, question answering, language translation, and even creative writing. Its ability to understand nuanced queries and produce coherent and contextually appropriate responses sets it apart in the realm of conversational AI. OpenAI's commitment to fine-tuning and optimizing the model ensures continuous advancements, making it an invaluable tool for developers, businesses, and researchers seeking cutting-edge natural language understanding and generation capabilities.

Despite its impressive capabilities, it's important to note that the chatbot has limitations, such as occasional generation of inaccurate or biased content. OpenAI encourages responsible use of the technology and emphasizes ongoing efforts to address these challenges, reflecting a commitment to ethical and safe deployment of advanced AI systems in real-world applications.

TABLE OF CONTENTS

List of Figures

CHAPTER – 1

INTRODUCTION

1.1 PROBLEM STATEMENT

The project aims to develop an advanced OpenAI-based chatbot that tackles key challenges in conversational AI. These challenges include improving Natural Language Understanding (NLU), ensuring context retention, personalizing interactions, incorporating multi-modal capabilities, addressing ethical concerns and biases, optimizing scalability and performance, refining the user interface and experience, and enabling seamless integration with external systems. The objective is to create an efficient, user-friendly chatbot that offers a more human-like and engaging conversational experience while adhering to ethical standards and catering to diverse user needs.

1.2 MOTIVATION

AI chatbots are motivated by their potential to enhance customer engagement, streamline communication, and provide efficient and cost-effective solutions across various industries. These chatbots offer benefits such as 24/7 availability, cost savings through automation, scalability, time efficiency, and data-driven insights. Additionally, their adaptability to different sectors and early adoption can provide a competitive advantage, positioning businesses as innovative and customer-focused. Overall, the motivation for AI chatbots lies in their ability to improve user experience, increase operational efficiency, and contribute to a positive brand image.

1.3 SCOPE

The scope of AI chatbots is broad and encompasses diverse industries and applications. From revolutionizing customer service in e-commerce to providing healthcare support, assisting in education, streamlining financial services, and enhancing internal business processes, AI chatbots play a pivotal role in improving efficiency and user experience. Their adaptability and versatility make them valuable tools in areas such as government services, travel, entertainment, language translation, and event management. As technology continues to advance, the scope of AI chatbots is expected to expand, offering innovative solutions and transforming the way businesses and organizations interact with users.

1.4 OUTLINE

AI chatbots represent a transformative technology with a broad scope of applications across industries. Comprising key components such as Natural Language Processing and Machine Learning, these chatbots excel in understanding user inputs, retaining context, and offering personalized, efficient interactions. Challenges include addressing biases, ensuring privacy, and navigating technical integration issues.

CHAPTER – 2

LITERATURE SURVEY

EXISTING SYSTEM:

Existing AI chatbot systems include Google's Dialogflow, Microsoft's Azure Bot Service, IBM Watson Assistant, Facebook Messenger Bot, Rasa, Amazon Lex, and OpenAI's ChatGPT. These systems offer a range of features such as natural language processing, multi-platform support, integration with cloud services, and open-source customization. Organizations and developers choose from these systems based on specific requirements, preferences, and desired levels of customization. The dynamic nature of the AI landscape means that new systems may have emerged since the last update in January 2022.

PROPOSED SYSTEM:

The proposed system for an AI chatbot aims to advance existing capabilities by incorporating enhanced Natural Language Processing (NLP), improved contextual understanding, and personalized user interactions. The system emphasizes multi-modal capabilities, ethical AI practices, scalability, and performance optimization. User experience is prioritized through refined UI/UX design, while integration with external systems and continuous learning mechanisms ensures comprehensive and up-to-date information. Security and privacy measures, a feedback loop for continuous improvement, and robust documentation contribute to the overall reliability and effectiveness of the AI chatbot system. This proposed system aligns with evolving user needs and technological advancements, emphasizing adaptability and responsiveness.

- Input Text
- LLM
- Output Text

CHAPTER - 3

SOFTWARE REQUIREMENTS SPECIFICATION

1. Purpose of the Document :

The purpose of this Software Requirements Specification (SRS) is to outline the functional and non-functional requirements for the development of an AI chatbot. This document serves as a comprehensive guide for stakeholders, including developers, testers, and project managers.

- ### Scope of the Project :

The AI chatbot will be designed to provide conversational interactions with users across various platforms and industries. It will incorporate natural language understanding, context retention, and personalization features to enhance user experience.

2. System Overview

- ### System Description :

The AI chatbot system will leverage advanced natural language processing and machine learning techniques to understand user inputs, retain context, and generate contextually relevant responses. It will be adaptable to various industries, providing personalized and efficient interactions.

- ### System Architecture :

The system architecture will consist of the following components:

- Natural Language Processing (NLP) module

- Machine Learning algorithms

- Context Management

- User Interface (UI) layer

- Integration with external systems

3. Functional Requirements:

- **User Interaction:**

- The chatbot shall be able to initiate and respond to user queries in a conversational manner.

- The chatbot shall support multi-modal interactions, including text, images, audio, and video.

- The chatbot shall retain context across multiple user interactions**.**

- **Personalization:**

- The chatbot shall analyze user behavior to provide personalized responses.

- Users shall have the option to customize the chatbot's behavior based on individual preferenc

- **Integration:**

- The chatbot shall integrate seamlessly with external systems, databases, and APIs to retrieve relevant information.

- Integration with third-party platforms (e.g., messaging apps) shall be supported.

- **Security and Privacy:**

- User data and interactions shall be encrypted to ensure security and privacy.

- The chatbot shall comply with data protection regulations and ethical AI practices.

4. Non-functional Requirements:

- **Performance:**

- The chatbot shall handle a minimum of 1000 concurrent users with response times under 1 second.

- **Scalability:**

- The system shall scale horizontally to accommodate increasing user loads.

- **Reliability:**

- The chatbot shall have a system uptime of at least 99.5%.

- **Usability:**

- The UI shall be intuitive and user-friendly, ensuring a positive user experience.

- **Maintainability:**

- The system shall allow for easy updates and maintenance without disrupting user interactions.

5. Constraints:

- **Technological Constraints:**

- The chatbot shall be developed using programming languages and frameworks compatible with the targeted platforms.

- **Regulatory Constraints:**

- The chatbot shall adhere to data protection laws and regulations in all regions of operation.

6.Documentation:

- **User Documentation:**

- A comprehensive user guide shall be provided for end-users.

- **Developer Documentation:**

- Technical documentation shall be provided for developers, including API specifications and integration guidelines.

7. Conclusion:

- This Software Requirements Specification provides a detailed outline of the functional and non-functional requirements for the development of an AI chatbot. It serves as a foundation for the design, development, and testing phases of the project, ensuring alignment with user expectations and industry standards..

CHAPTER – 4

SYSTEM DESIGN

Use case Diagrams:

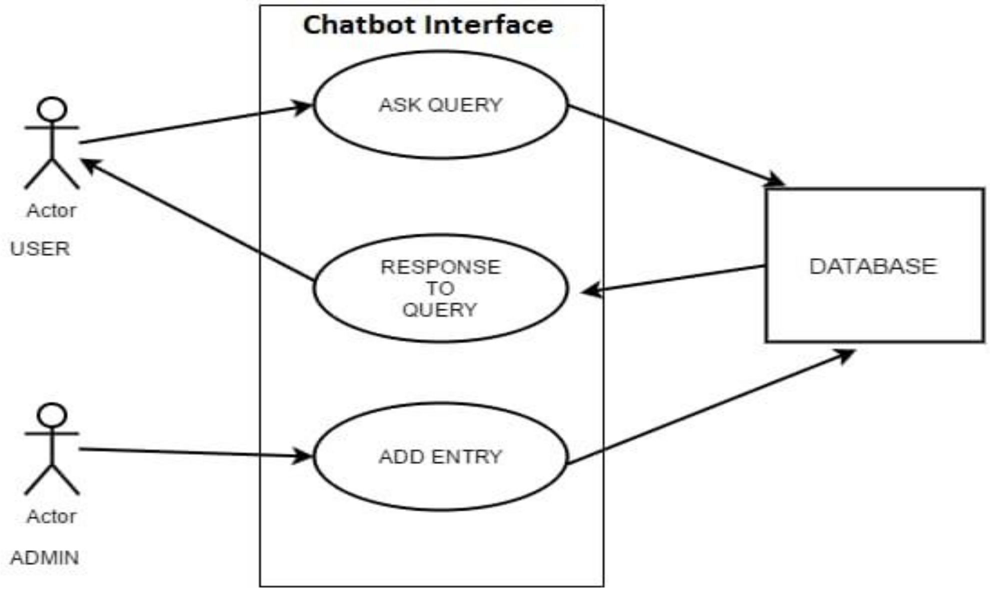


Fig 4.1: Use case diagram for User

- A use case diagram for an OpenAI chatbot represents the interactions between users and the system.
- In the context of an OpenAI chatbot, actors might include end-users and administrators, while use cases could encompass actions like initiating a conversation, receiving responses, and managing chatbot settings.
- The diagram illustrates the high-level functionalities and how users interact with the chatbot, providing a visual representation of the system's key features and user interactions.

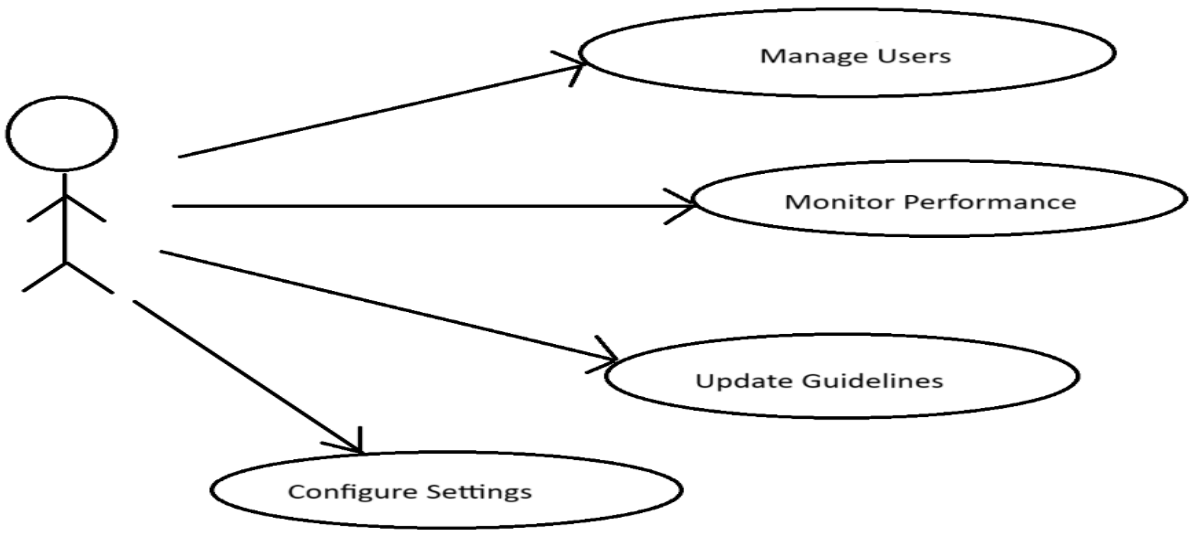


Fig 4.2: Use case diagram for Admin

- In the use case diagram for an OpenAI chatbot's admin interface, the primary actor is the administrator.
- Key use cases may involve managing user access, monitoring chatbot performance, updating conversation guidelines, and settings.
- The diagram visualizes how administrators interact with the system to oversee and control various aspects of the chatbot's operation, emphasizing functionalities relevant to administrative tasks and responsibilities.

Class Diagram:

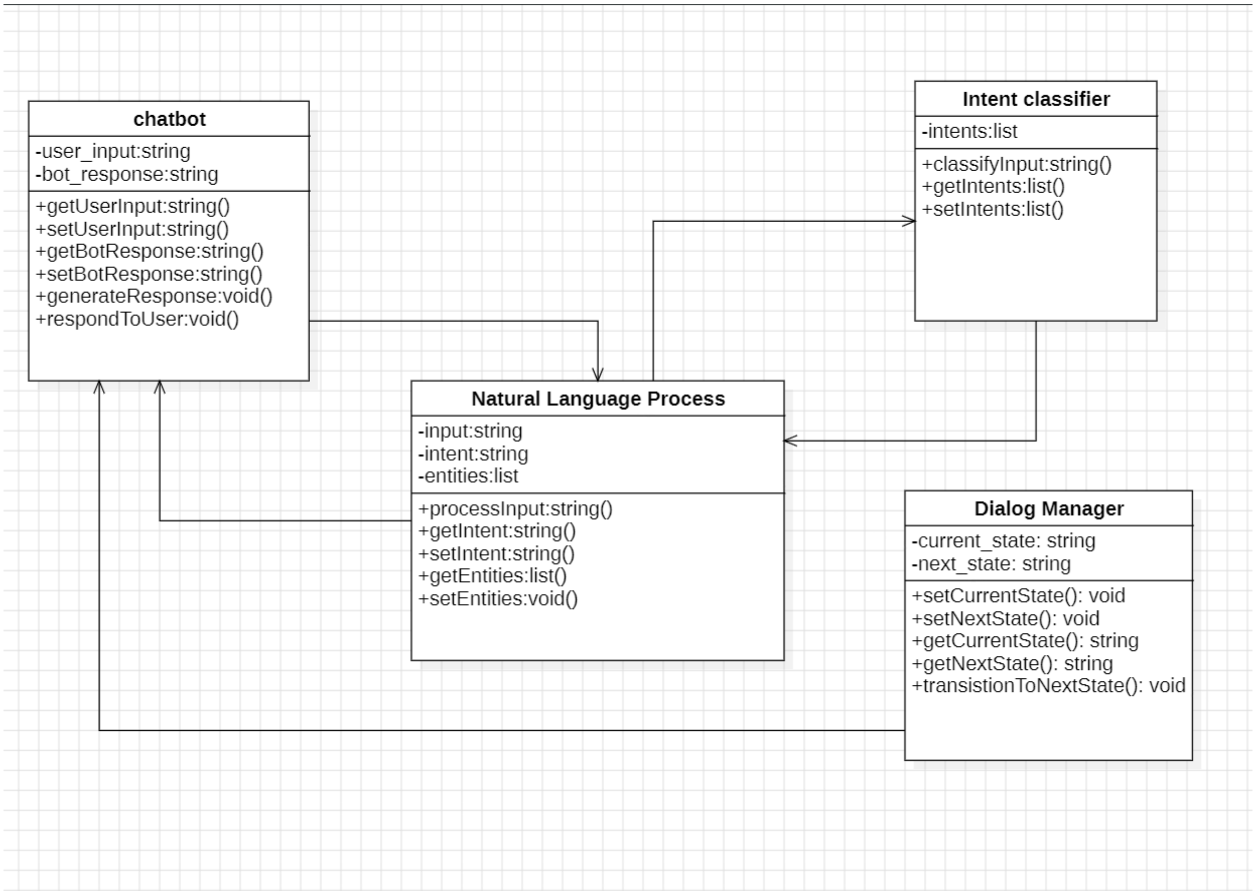


Fig.4.3 : Class diagram for AI ChatBot.

- A class diagram for an OpenAI chatbot provides a static view of the system's structure, focusing on the classes, their attributes, and relationships.
- Common classes might include NLP, Chatbot, Intent Classifier, andDialog Manager.
- The User class could have attributes like user_input and preferences, while the Chatbot class might contain methods for generating responses.
- Associations between classes illustrate relationships.
- The class diagram offers a blueprint for understanding the essential entities, their properties, and how they relate within the OpenAI chatbot system.

Sequence Diagram:

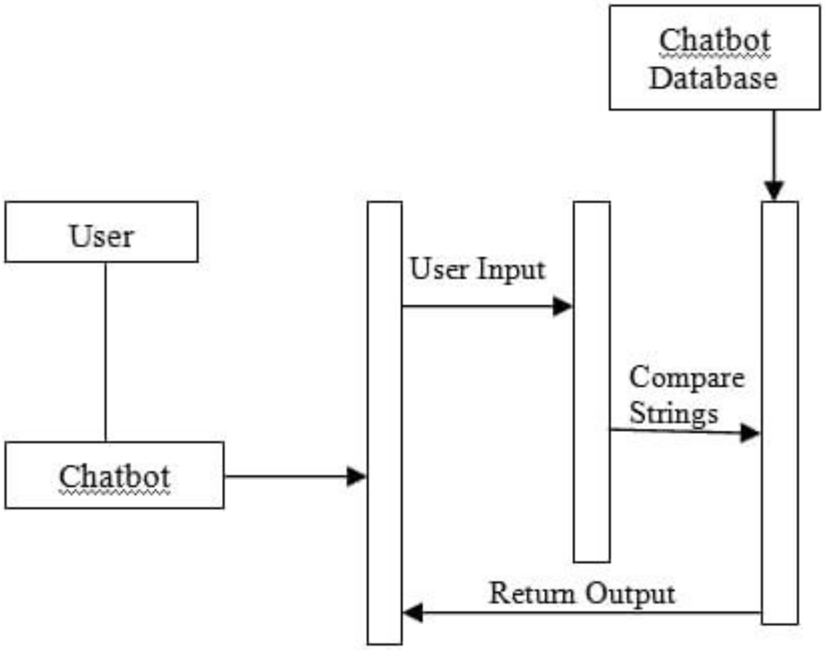


Fig4.3 : Sequence Diagam for AI ChatBot

- A sequence diagram for an OpenAI chatbot illustrates the dynamic interactions and message flow between different components over time.
- The diagram captures the initiation of a conversation, user input, processing by the chatbot, and the delivery of responses.
- It highlights the sequential nature of communication, showcasing the collaboration between system components.

CHAPTER – 5

IMPLEMENTATION

5.1 SAMPLE CODE

# In[1]:

```
!pip install langchain

!pip install openai

!pip install gradio

!pip install huggingface_hub
```

# In[2]:

```
import os

import gradio as gr

from langchain.chat_models import ChatOpenAI

from langchain import LLMChain, PromptTemplate

from langchain.memory import ConversationBufferMemory
```

# In[3]:

```
OPENAI_API_KEY="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"

os.environ["OPENAI_API_KEY"] = OPENAI_API_KEY
```

# In[4]:

```
template = """You are a helpful assistant to answer user queries.

{chat_history}
```

```
User: {user_message}

Chatbot:"""



prompt = PromptTemplate(

    input_variables=["chat_history", "user_message"], template=template

)



memory = ConversationBufferMemory(memory_key="chat_history")
```

# In[5]:

```
# from langchain.llms import HuggingFacePipeline

# hf = HuggingFacePipeline.from_model_id(

#     model_id="gpt2",

#     task="text-generation",)
```

# In[6]:

```
llm_chain = LLMChain(

    llm=ChatOpenAI(temperature='0.5', model_name="gpt-3.5-turbo"),

    prompt=prompt,

    verbose=True,

    memory=memory,

)
```

# In[7]:

```
def get_text_response(user_message,history):

    response = llm_chain.predict(user_message = user_message)

    return response
```

# In[8]:

```
demo = gr.ChatInterface(get_text_response, examples=["How are you doing?","What are your interests?","Which places do you like to visit?"])
```

# In[9]:

```
if __name__ == "__main__":

    demo.launch() #To create a public link, set `share=True` in `launch()`. To enable errors and logs, set `debug=True` in `launch()`.
```

# In[10]:

```
from huggingface_hub import notebook_login



notebook_login()
```

# In[11]:

```
from huggingface_hub import HfApi

api = HfApi()
```

# In[12]:

```
HUGGING_FACE_REPO_ID = "gouthamvarma2347/ARGcommandoss"
```

# In[13]:

```
%mkdir /content/ChatBotWithOpenAI

!wget -P /content/ChatBotWithOpenAI/ https://s3.ap-south-1.amazonaws.com/cdn1.ccbp.in/GenAI-Workshop/ChatBotWithOpenAIAndLangChain/app.py

!wget -P /content/ChatBotWithOpenAI/ https://s3.ap-south-1.amazonaws.com/cdn1.ccbp.in/GenAI-Workshop/ChatBotWithOpenAIAndLangChain/requirements.txt
```

# In[14]:

```
%cd /content/ChatBotWithOpenAI


api.upload_file(
    path_or_fileobj="./requirements.txt",
    path_in_repo="requirements.txt",
    repo_id=HUGGING_FACE_REPO_ID,
    repo_type="space")


api.upload_file(
    path_or_fileobj="./app.py",
    path_in_repo="app.py",
    repo_id=HUGGING_FACE_REPO_ID,
    repo_type="space")
```

# In[15]:

generate

CHAPTER - 6

RESULT ANALYSIS

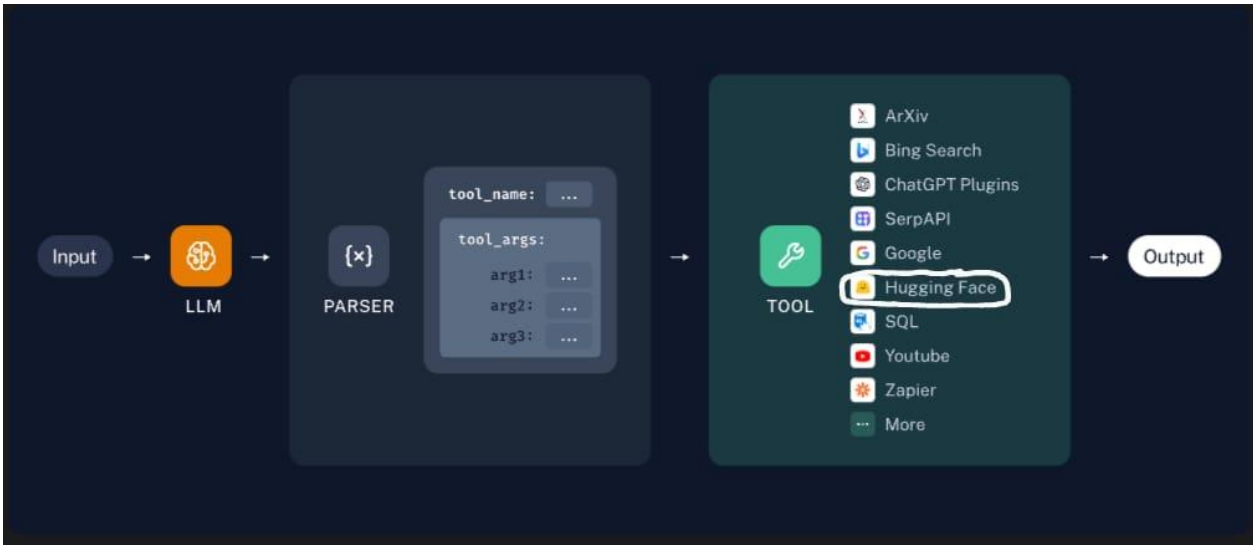


Fig 6.1: Large Language Models

- Large Language Models (LLMs) are algorithms designed for natural language processing (NLP) tasks.
- They use deep learning techniques, specifically models like GPT-3 (Generative Pre-trained Transformer 3), to understand and generate human-like text.

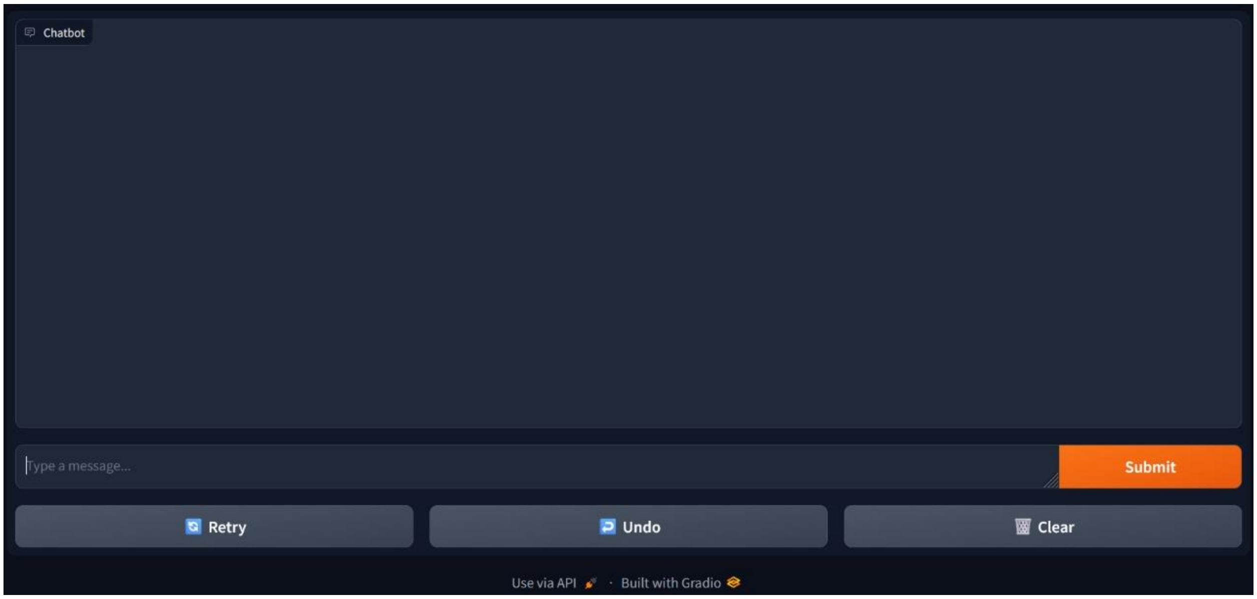
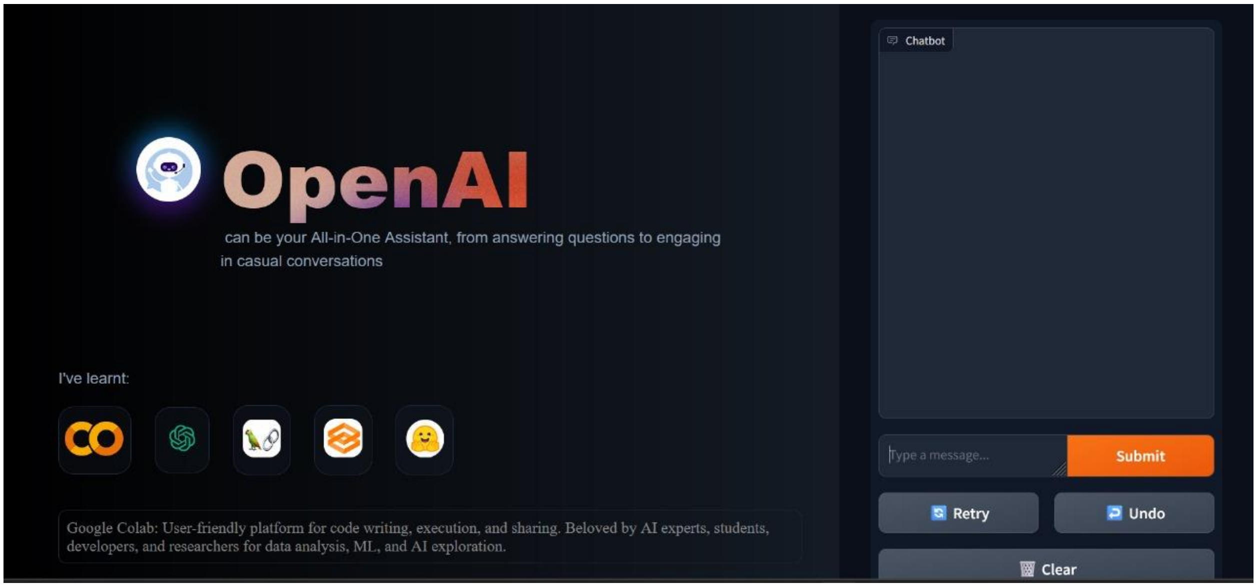


Fig 6.2: CHATBOT INTERFACE

- It is our **Chatbot Interface** which is designed by using **Gradio**
- Gradio is an open-source Python library that simplifies the process of creating user interfaces for machine learning models.
- Gradio is designed to be user-friendly and allows developers, even those without extensive front-end experience, to quickly build interactive interfaces for their machine learning models.

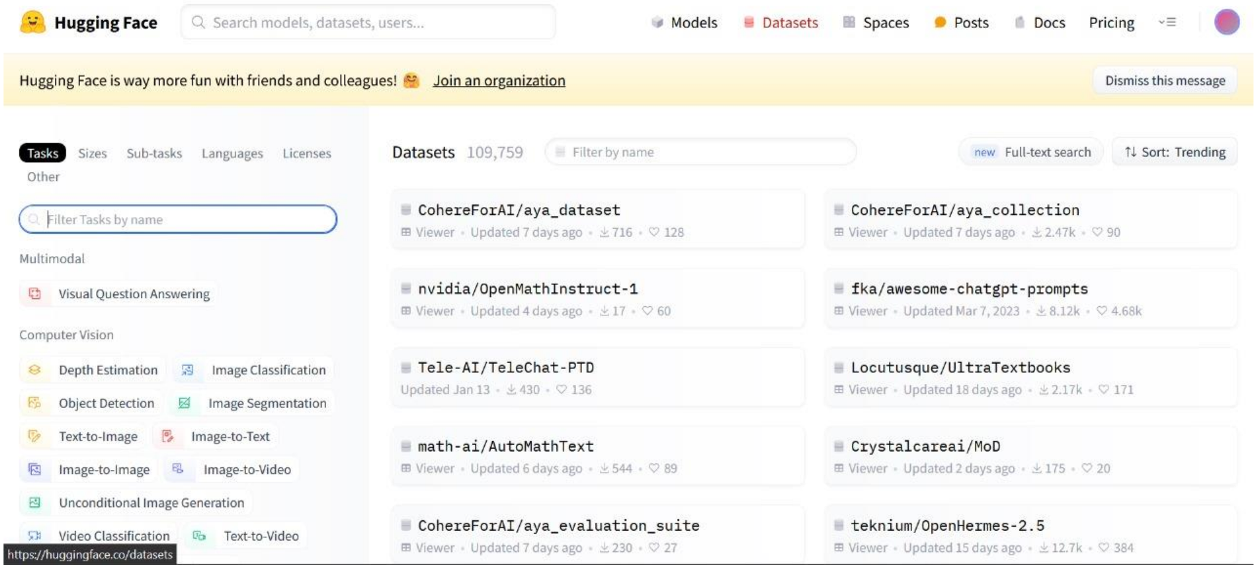


can be your All-in-One Assistant, from answering questions to engaging in casual conversations

Google Colab: User-friendly platform for code writing, execution, and sharing. Beloved by AI experts, students, developers, and researchers for data analysis, ML, and AI exploration.

Fig 6.3: OVERALL INTERFACE

- It is an Interface of our MiniProject which is developed by using HTML, CSS and JavaScript



Hugging Face is way more fun with friends and colleagues! 🤗 Join an organization

Datasets 109,759

CohereForAI/aya_dataset
Viewer • Updated 7 days ago • 716 • 128

CohereForAI/aya_collection
Viewer • Updated 7 days ago • 2.47k • 90

nvidia/OpenMathInstruct-1
Viewer • Updated 4 days ago • 17 • 60

fka/awesome-chatgpt-prompts
Viewer • Updated Mar 7, 2023 • 8.12k • 4.68k

Tele-AI/TeleChat-PTD
Updated Jan 13 • 430 • 136

Locutusque/UltraTextbooks
Viewer • Updated 18 days ago • 2.17k • 171

math-ai/AutoMathText
Viewer • Updated 6 days ago • 544 • 89

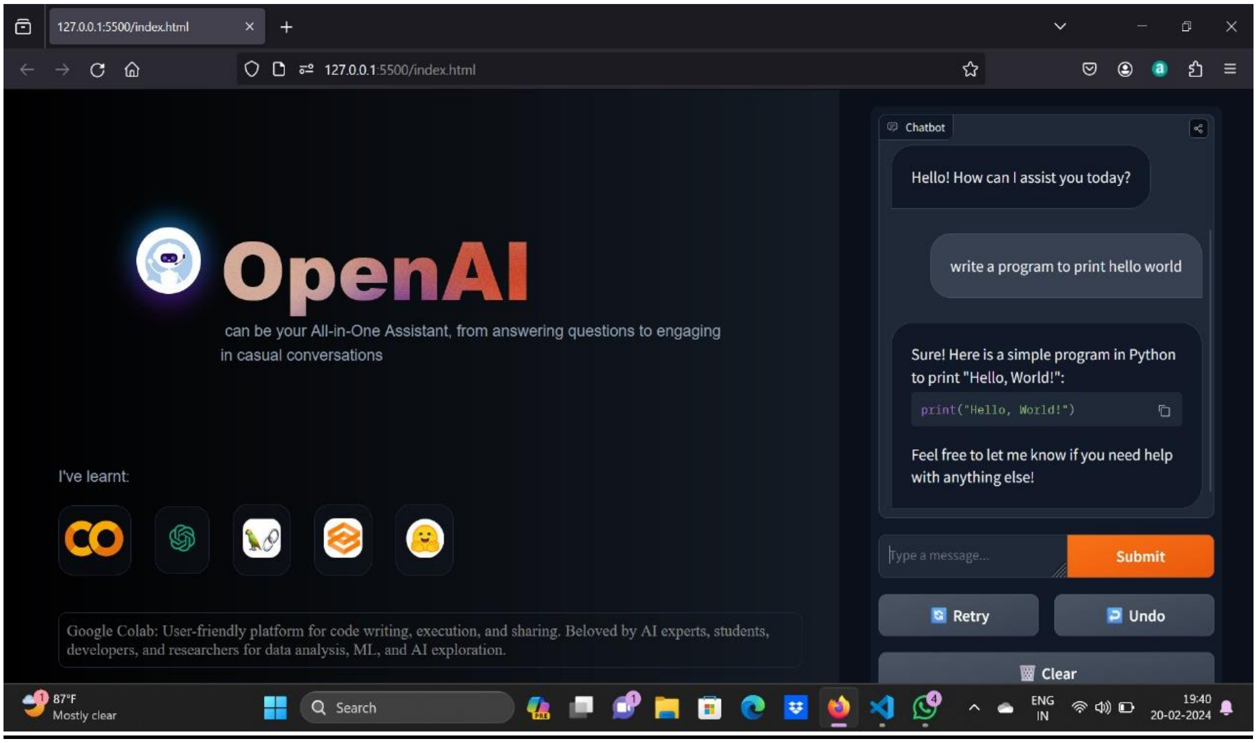
Crystalcareai/MoD
Viewer • Updated 2 days ago • 175 • 20

CohereForAI/aya_evaluation_suite
Viewer • Updated 7 days ago • 230 • 27

teknium/OpenHermes-2.5
Viewer • Updated 15 days ago • 12.7k • 384

Fig 6.4: DATASETS BY HUGGING FACE

- The **Hugging Face Hub** is a remarkable platform that hosts an extensive collection of machine learning resources.
- The Hugging Face Hub is designed to facilitate the sharing and discovery of pre-trained models, datasets, and other resources related to NLP and machine learning.
- **Models**: The Hub houses over **350,000 models**, including state-of-the-art models for natural language processing (NLP).
- **Datasets**: With **75,000 datasets**, the Hub offers a wide variety of data across different domains and modalities.



OpenAI

can be your All-in-One Assistant, from answering questions to engaging in casual conversations

I've learnt:

Google Colab: User-friendly platform for code writing, execution, and sharing. Beloved by AI experts, students, developers, and researchers for data analysis, ML, and AI exploration.

Hello! How can I assist you today?

write a program to print hello world

Sure! Here is a simple program in Python to print "Hello, World!":

```
print("Hello, World!")
```

Feel free to let me know if you need help with anything else!

Figure 6.5: GENERATED OUTPUT

- It is an image showing about output is given by out chatbot.

CHAPTER – 7

CONCLUSION AND FUTURE SCOPE

OpenAI's chatbot technology, exemplified by models like GPT-3, has achieved remarkable advancements in Natural Language Processing (NLP). This technology has demonstrated its versatility across various applications, including content generation, language translation, and code completion. However, ethical considerations, such as the potential for misuse, have prompted OpenAI to actively implement safety measures and ethical guidelines.

The **Future** of OpenAI's chatbot technology entails ongoing research and development efforts to further improve language models. Anticipated developments include increased customization and fine-tuning options, expanded integration into diverse industries like customer service and healthcare, enhanced accessibility through APIs, and continued efforts to address ethical concerns for responsible and safe usage. Monitoring OpenAI's updates is crucial for staying informed about the evolving landscape of chatbot technology.

BIBLIOGRAPHY

General Information:

- OpenAI Website: https://openai.com/
- Introducing ChatGPT Blog Post: https://openai.com/chatgpt

Technical Papers:

- Attention Is All You Need by Vaswani et al. (2017): https://arxiv.org/pdf/1706.03762 (This paper introduced the Transformer architecture, which is the foundation of GPT-3.5)
- Learning to Summarize with Human Feedback by Stiennon et al. (2020): https://arxiv.org/abs/2009.01325
- Scaling Laws for Reward Model Overoptimization by Gao et al. (2022): https://arxiv.org/abs/2210.10760

Additional Resources:

- OpenAI API Reference: https://openai.com/blog/openai-api
- OpenAI Blog: https://openai.com/blog
- OpenAI Community Forum: https://community.openai.com/

Disclaimer:

This bibliography is not exhaustive and is intended to provide a starting point for further research. Please note that the information contained in the listed resources may not be entirely up-to-date due to the rapid development of the field.

APPENDIX A: TOOLS AND TECHNOLOGIES

- Building an OpenAI chatbot involves leveraging a variety of tools and technologies to enhance development and deployment. Below are key elements commonly used in the creation of chatbots using OpenAI's technology:

- OpenAI GPT Models:
  - GPT-3: The latest version of the Generative Pre-trained Transformer by OpenAI, providing powerful natural language processing capabilities.

- Programming Languages:
  - Python: A widely used programming language for its simplicity and extensive libraries. It is often used to integrate OpenAI's models into applications.

- Development Frameworks:
  - OpenAI API: Access to OpenAI's models is typically facilitated through their API, allowing developers to interact with GPT models seamlessly.

- Front-end Technologies:
  - HTML, CSS, JavaScript: Essential for developing user interfaces and enabling real-time interactions with the chatbot on websites.

- Natural Language Processing (NLP) Libraries:
  - spaCy, NLTK: Libraries that facilitate language processing tasks, such as tokenization, entity recognition, and sentiment analysis.

- **Large language model:** play a crucial role in the development of sophisticated chatbots, providing them with the ability to understand and generate human-like text.