Building an AI Chatbot Using LLM

A major project report submitted in partial fulfilment of the requirement for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering / Information Technology

Submitted by

Abhishek (201559)

Ishika Goswami (201331)

Under the guidance & supervision of

Dr. Deepak Gupta

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology,

Waknaghat, Solan - 173234 (India)

CANDIDATE'S DECLARATION

I hereby declare that the work presented in this report entitled **'Building an AI Chatbot Using LLM'** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2023 to May 2024 under the supervision of **Dr. Deepak Gupta** (Assistant Professor, Department of Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature with Date)
Student Name: Abhishek
Roll No.: 201559

(Student Signature with Date)
Student Name: Ishika Goswami
Roll No.: 201331

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature with Date)
Supervisor Name: Dr. Deepak Gupta
Designation: Assistant Professor (SG)
Computer Science and Engineering/Information Technology Department
Dated:

ACKNOWLEDGEMENT

To begin, I would like to express my heartfelt gratitude to almighty God for his heavenly grace, which enabled us to successfully complete the project work.

I am extremely grateful and wish to express my deep gratitude to Supervisor **Dr. Deepak Gupta, Assistant Professor (Senior Grade), Department** of CSE & IT Jaypee University of Information Technology, Waknaghat. His never-ending patience, intellectual direction, persistent encouragement, constant and vigorous supervision, constructive criticism, helpful suggestions, and reading numerous poor versions and revising them at all stages allowed this project to be completed. I would like to express my heartiest gratitude to Dr. Vivek Sehgal**,** Head of Department of CSE & IT, for his kind help to finish my project.

I would also like to express my gratitude to everyone who has assisted me in making this project a success, whether directly or indirectly. In this unusual scenario, I'd like to express my gratitude to the different staff members, both teaching and non-teaching, who have provided me with valuable assistance and assisted my project. Finally, I must express my gratitude for my parents' unwavering support and patience.

Abhishek (201559)

Ishika Goswami (201331)

TABLE OF CONTENT

LIST OF ABBREVIATIONS

Abbreviations	Meaning
NLP	Natural language processing
NLU	Natural language understanding
RNN	Recurrent neural network
Chatbot	Chatting Robot
AI	Artificial Intelligence
GPT	Generative Pre-trained Transformer
API	Application Programming Interface

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

In our digital world, where communication is key, the emergence of AI-powered chatbots has revolutionized the way we interact with technology. This project delves into the development of an AI chatbot utilizing Large Language Models (LLM), a cutting-edge technology in the field of natural language processing. The journey begins with a comprehensive exploration of existing chatbot frameworks and technologies, analysing their strengths and weaknesses. Through meticulous research, we identified LLM as the most promising candidate due to its ability to understand and generate human-like text, enabling more meaningful and engaging conversations.

The LLM was employed for its ability to process long range dependencies, enabling the chatbot to comprehend context in a more comprehensive manner. The report discusses the methodology employed, including data preprocessing, model training, the transformer architecture and validation using the test cases along with the conclusion and the work we wish to do in the upcoming months. Additionally, it details the challenges encountered and the strategies employed to counter them and enhance the chatbot's performance.

The final AI chatbot demonstrates impressive capabilities in understanding and generating human-like responses across a diverse range of topics. Its ability to engage users in meaningful conversations showcases the potential of LLM technology in enhancing human-computer interaction.

Overall, this project contributes to the evolving field of AI-driven conversational agents and highlights the potential of utilizing LLMs in building sophisticated chatbot systems. The journey that we lived while working on this project is mentioned in this project report in the sequential order.

CHAPTER 01: INTRODUCTION

This chapter of the project report is the beginning of the content of this report. It contains the building up of the plot of this report. The problem statement along with the main objectives of this project are discussed in here. The significance of this project and the real motivation behind the intentions to take up this topic as our project are also listed in detail in this particular chapter. The organization of this project report is also listed in this very chapter. A basic description regarding the LLMs in the title of this project report is also given here.

The various methodologies we came across while we started working on this project are also explained. The language used for working on this project is also mentioned in this chapter.

1.1 INTRODUCTION

In the time of modern technology, AI has emerged as a ground breaking technology. AI Chatbots which use Natural Language Processing and Machine Learning techniques have gained a significant role and platform. In recent years, AI has witnessed extraordinary growth, especially in the field of conversational agents or Chatbots. These Chatbots are designed to engage in conversation with users, providing with relevant information and assistance. Among the various approaches used to create these conversational agents, one of the most promising and sophisticated methods involve LLMs.

To comprehend the significance and association of LLMs in Chatbot creation, its essential to understand the core concepts developing this technology. LLMs are a form of AI that processes and generates human like text, learning patterns and structures form vast amounts of data. These models, such as OpenAI's GPT have gained prominence due to their ability to understand context and semantic difficulties with human language.

The primary objective of this report is to provide a comprehensive overview of the steps and considerations involved in building an AI Chatbot using LLMs. From the initial data gathering and preprocessing stages to fine tuning the model and deploying the Chatbot.
This report will also touch upon the ethical considerations associated with deploying AI Chatbots. Issues such as data privacy and bias are also discussed highlighting the importance of ethical practices in the development and implementation of these Chatbots.

1.2 PROBLEM STATEMENT

This project aims to develop an advanced AI Chatbot utilizing LLMs to enhance the human-AI interaction. The challenge involves building a Chatbot that can accurately understand and respond to natural language unput in various contexts. The Chatbot should maintain context over multi turn conversation, provide relevant and coherent responses and continuously learn from the user interactions to improve its performance.

Moreover, the ethical aspect is crucial, necessitating the prevention of harmful or inappropriate content generation. Providing the Chatbot with adaptivity to the individual user preferences and providing tailored responses based on user history and stated preferences is also a challenge to overcome in this project.

1.3 OBJECTIVES

The main objectives which we try to aim during the completion of this project are all listed below:

- To develop a sophisticated NLP system that can accurately interpret user inputs, identify the intents and extract key entities.
- To create a Chatbot capable of maintaining context across multiturn conversations and managing topic transitions fluidly.
- To create a Chatbot that interacts in a way that feels natural and engaging blurring the line between human-human and human-AI conversations.
- To develop a Chatbot whose responses adhere to ethical standards and avoid generating harmful or offensive content.

1.4 SIGNIFICANCE AND MOTIVATION OF THE PROJECT WORK

We as human beings tend to produce goods and services that mostly end up easing our day-to-day work in the best way possible. One such invention are the LLMs. Work on the topic related to the machines having the ability to understand the human language and converse in a language native to humans began in around 1967 when Eliza was invented. Since then, this has been ongoing journey. But it achieved a massive push during 2017 with the release of Transformers and then during 2022 when OpenAI released ChatGPT [1].

This not only led to ease the human life but also provided the new area of research and advancement in the field of AI. The significance lies in the application of language models that enable the Chatbot to comprehend and respond to human input in a way that feels natural and engaging. This is more than just writing a program to answer questions.

This project holds great importance for several reasons. Firstly, it represents a practical application of the cutting-edge technology in the field of AI. By working on this project, we are exploring how advanced these language models can be and how they can be used along with gaining knowledge on their working behind the scenes.

This project provides an opportunity to enhance our understanding and skills in the field of AI. The real hands-on experience that we are gaining during this project is also of great significance as the field of AI is a never-ending field in which the advancements will continue for a very long time.

Furthermore, the implementation of an AI Chatbot has significant real-world implications. Chatbots are increasingly being used in various industries such as customer service, healthcare and now even in education to provide assistance. By engaging in this project, we are actively contributing to the growing field of generative AI and exploring its potential applications.

One of the primary motivations behind this project is the desire to explore and understand the capabilities of AI especially in the domain of the rising LLMs. By diving into the development of an AI Chatbot, we actively participate in understanding the evolution of AI and gain valuable knowledge about the behind the scene working.

The motivation also arises from the awareness of the growing role of AI in our daily lives. From virtual assistants on smartphones to automated customer services, AI is increasingly becoming an integral part of various industries. Understanding the capabilities and limitations of AI Chatbots also drive us.

1.5 LLMs

LLMs are simply powerful Artificial Intelligence models that are trained on vast amounts of text data. The main objective is to understand and generate human like text. These models use deep learning techniques. They process and generate text based on the patterns and structures they have learned on the training data [1]. During the testing phase, they interpret the patterns

in the data and try to process those as in the training data and come up with similar results. So, the quality and quantity of the training dataset are important factors in the working of LLMs [2].

Large refers to the enormous amount of training dataset and the large number of parameters. Only certain organizations have the capability to train such LLMs with huge datasets and tremendous number of parameters [3], [4]. LLMs are trained with petabytes of data and generate billions of parameters. They have to undergo pre-training and fine tuning. Pre-training means to train the model for a general purpose with a large data set. The fine tuning involves tuning the model for specific aims with a much smaller dataset.

LLMs have to go through a complex process during the training that involves multiple steps. It usually starts with the unsupervised learning in which the model learns to make all the observation on its own. So, it is important that the data must be of good quality. The model creates its own clusters based on its observations of the data as per [2]. Then comes a step that involves supervised learning. In this step the human supervisors make the class labels based on the data and the model has to classify the data based on those known class labels. The model just has to classify the data from the dataset such that each of the data chunk falls into either one of those class labels.

Then comes the reinforcement learning from human feedback (RHLF). This involves using rewards as a way to give the system incentive to find new patterns [3]. Here, the quality of the outcome given by model is improved. This allows the model to make good predictions. The outputs with the greatest number of rewards are only kept and the ones with the lowest rank are discarded. This is how the model improves its quality over time.

The LLMs are designed based on the Transformer architecture that was introduced in the year 2017. It overcame the limitations of the RNN architecture as per [1] and made the LLMs famous and popular on a very large scale which is discussed in detail in the upcoming chapters of this project report.

1.6 BENEFITS OF USING LLMS

Although there are many ways for creating a Chatbot. But still we decided on doing so with the help of LLMs. This is because of the popularity of these language models. But it’s not

just their popularity that we kept in mind while choosing them. These language models also provide certain benefits which are listed below:

- A single model can be used for a variety of different tasks. This is due to the fact that LLMs are trained on petabytes of data and generate about a billion parameters which enable them to perform different types of tasks including language translation, sentence completion, text classification, question answering and more [3].
- The fine-tuning process requires minimal field training data. They obtain decent performance even with little domain training data [2].
- Their performance is continuously growing with more data and parameters. We do know that the quantity of data is only going to increase with time and so will the quality in the upcoming times. As a result, the performance of these models will also rise up based on the growth of data.

1.7 METHODOLOGY

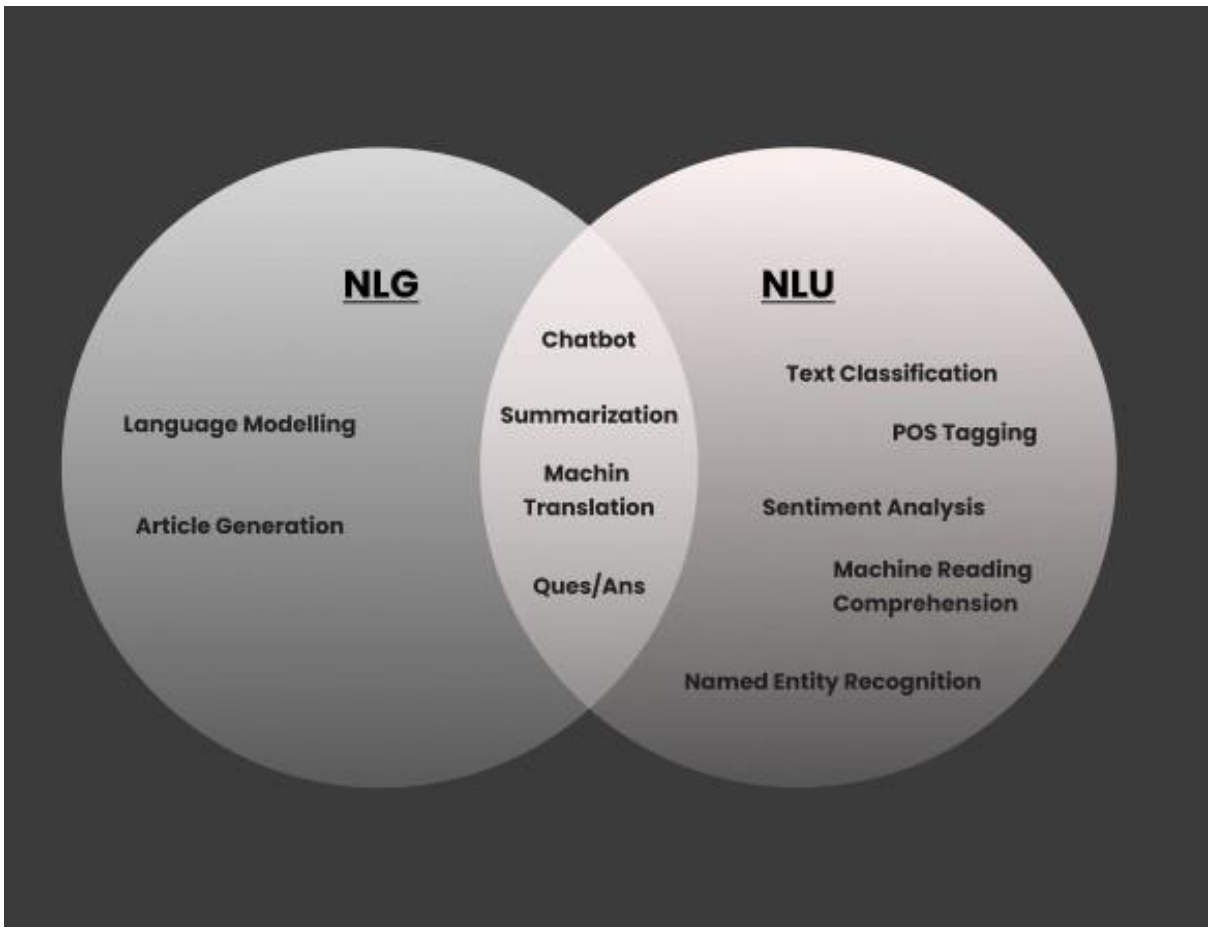

Fig. 1.1: NLG and NLU Being a Subset of NLP

1.7.1 NLU

The Natural Language Understanding (NLU) stands as a pivotal domain within the realm of AI, delving into the intricate connection between computers and human language. At its core, NLU empowers machines to grasp, interpret, and respond to spoken or written language in a manner akin to human comprehension [5]. This profound capability is achieved through a confluence of sophisticated methodologies, including deep learning, machine learning, and NLP. By harnessing these techniques, NLU holds the promise of revolutionizing human-computer interaction and streamlining various aspects of our lives.

One of the primary advantages of NLU lies in its ability to bridge the communication gap between humans and machines. Traditionally, interactions with technology often required users to adapt to rigid interfaces or specific commands, leading to frustration and inefficiency. However, with NLU, machines are endowed with the capacity to understand and respond to natural language input, thereby facilitating more intuitive and seamless exchanges [5]. This advancement not only enhances user experience but also broadens the accessibility of technology to individuals with diverse linguistic backgrounds or varying levels of technological literacy.

Crucially, NLU excels in navigating the inherent ambiguity of human language, a feat that has long eluded conventional computing systems. Human communication is replete with nuances, colloquialisms, and context-dependent meanings, posing a formidable challenge for machines attempting to decipher it. Yet, through the application of advanced algorithms and linguistic models, NLU enables computers to discern the intended meaning behind human utterances with remarkable accuracy. Whether it's extracting key information from a text message, comprehending the sentiment conveyed in a social media post, or interpreting the intricacies of a spoken conversation, NLU equips machines with the cognitive prowess to navigate the complexities of human language [5].

Moreover, the integration of NLU into various technological applications heralds a new era of automation and efficiency. One notable application lies in the realm of customer service, where NLU-powered chatbots and virtual assistants are increasingly assuming frontline roles in addressing customer inquiries and resolving issues. These AI-driven interfaces possess the ability to understand natural language queries, retrieve relevant information from knowledge bases, and provide personalized responses in real-time. By automating routine interactions and

triaging inquiries, NLU-driven systems alleviate the burden on human customer service representatives, allowing them to focus on more complex or specialized tasks. This not only enhances operational efficiency but also contributes to cost savings for businesses and heightened satisfaction for customers.

Beyond its implications for customer service, NLU finds wide-ranging applications across diverse domains, each harnessing its capabilities to unlock new possibilities. In the realm of healthcare, NLU facilitates the analysis of medical records, research literature, and patient-doctor interactions, thereby aiding in diagnosis, treatment planning, and medical research. By parsing through vast volumes of textual data, NLU systems can identify patterns, extract relevant insights, and support clinical decision-making processes, ultimately enhancing patient outcomes and advancing medical knowledge.

Similarly, in the realm of finance and commerce, NLU plays a pivotal role in information extraction, sentiment analysis, and predictive modelling. Financial institutions leverage NLU-powered algorithms to analyze market trends, assess investor sentiment, and automate tasks such as news aggregation and financial reporting. By distilling complex financial information into actionable insights, NLU empowers investors, analysts, and decision-makers to make informed choices in a rapidly evolving landscape.

Furthermore, the influence of NLU extends into the realm of education, where it facilitates personalized learning experiences, automated grading, and intelligent tutoring systems. By analysing student responses, identifying misconceptions, and tailoring instructional content to individual learning styles, NLU-driven platforms enhance the efficacy of educational interventions and promote student engagement and success.

In the context of information retrieval and knowledge management, NLU enables more nuanced search queries, semantic indexing, and content recommendation systems. By understanding the contextual meaning of user queries and the underlying semantics of documents or web pages, NLU systems can deliver more relevant and accurate search results, thereby enhancing the discoverability and accessibility of information in vast digital repositories.

Moreover, NLU holds immense potential in supporting multilingual communication, cross-cultural understanding, and language translation. By transcending linguistic barriers and

facilitating real-time translation and interpretation, NLU fosters global collaboration, cultural exchange, and mutual understanding across diverse communities and contexts.

As the volume and complexity of textual data continue to proliferate in the digital age, the importance of NLU in enabling computers to analyze and understand human language in a meaningful way becomes increasingly apparent. This paradigm shift not only streamlines daily tasks and augments human capabilities but also opens up new frontiers of innovation and discovery. By harnessing the power of NLU, we stand poised to transform how we interact with technology, unleashing a future where human-computer collaboration is characterized by fluidity, efficiency, and understanding.

In a broader context, NLU finds diverse applications, including categorization of text, analysis of sentiment, and development of Chatbot and similar virtual assistants. With ever-growing volume of data generated daily, NLU is becoming increasingly essential in enabling computers to analyze human language in much meaningful way. This shift has potential to transform how we interact with technology. It has resulted in simplifying the daily tasks.

1.7.2 NLP

Natural Language Processing (NLP) stands at the forefront of AI, wielding its capabilities to bridge the gap between human language and computers. At its core, NLP delves into the intricate dynamics of human communication, enabling machines to comprehend, interpret, and even generate language. This field's significance lies not only in its technical advancements but also in its transformative potential for human-computer interactions [5].

One of the paramount advantages of NLP lies in its ability to enrich communication between humans and machines. By enabling computers to understand and respond to natural language, NLP facilitates more intuitive and seamless interactions. Imagine conversing with a virtual assistant or chatbot that comprehends nuances, idioms, and context, akin to interacting with another human. This enhanced fluidity in communication not only fosters efficiency but also cultivates a more relaxed and improved mode of human-computer interaction.

Moreover, NLP holds the power to automate tasks previously reserved for humans, thereby revolutionizing various industries. Take, for instance, the automation of language translation. By leveraging NLP algorithms, machines can swiftly translate text between languages, liberating human translators to focus on more intricate linguistic tasks. This automation not

only translates to significant cost savings for businesses but also contributes to more effective cross-cultural communication on a global scale.

The applications of NLP span across a multitude of industries, each reaping its benefits in unique ways. In marketing, NLP algorithms analyze consumer sentiment from social media data, enabling companies to tailor their marketing strategies accordingly. In finance, NLP-powered chatbots assist customers with banking inquiries, providing personalized recommendations and streamlining customer service processes. In healthcare, NLP aids in analysing medical records and clinical notes, extracting valuable insights for diagnosis and treatment.

The growing volume of data generated daily further amplifies the significance of NLP. With the proliferation of digital content across various platforms, the need for efficient language processing tools becomes increasingly paramount. NLP algorithms sift through vast amounts of text data, extracting valuable information, detecting patterns, and uncovering actionable insights. This capability not only enhances decision-making processes but also facilitates the automation of repetitive tasks, thereby boosting productivity across industries.

Within the realm of NLP, NLU emerges as a crucial subset, alongside Natural Language Generation (NLG). NLU focuses on deciphering the meaning and intent behind human language, enabling computers to comprehend and extract relevant information from text or speech. On the other hand, NLG involves the generation of human-like language or text based on input data or predefined rules [5]. The synergistic integration of NLU and NLG enhances the overall efficiency of NLP systems, paving the way for innovative applications such as chatbots.

Chatbots, fuelled by NLP technologies, exemplify the convergence of NLU and NLG in real-world applications. These virtual assistants simulate human-like conversations, understanding user queries, and providing relevant responses in natural language. Whether it's customer support, information retrieval, or task automation, chatbots powered by advanced NLP algorithms are revolutionizing the way businesses engage with their customers and streamline their operations.

In conclusion, NLP transcends mere technical innovation; it revolutionizes the way humans interact with technology. By enabling computers to comprehend, interpret, and generate human language, NLP fosters more intuitive and natural communication between humans and

machines. From automating mundane tasks to extracting insights from vast datasets, the applications of NLP are boundless, reshaping industries and simplifying daily tasks. As NLP continues to evolve and integrate with other fields of artificial intelligence, its impact on society will only grow, ushering in a new era of intelligent human-computer interaction.

NLU emerges as a subset of NLP, alongside NLG. The synergistic integration of NLU and NLG enhances the efficiency of NLP, opening avenue for their application, such as in the development of Chatbots.

1.7.3 DEEP LEARNING

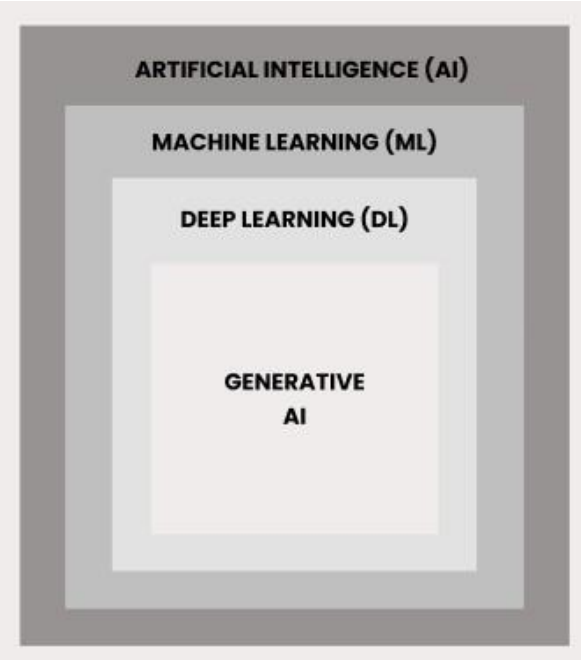

Fig. 1.2: Generative AI Being a Subset of Deep Learning

Deep Learning is subset of Artificial Intelligence that was build based on the human brain. Artificial Intelligence was a term introduced in 1956 by John McCarthy whereas deep learning is a term that was introduced by Igor Aizenberg in the year 2000.

Deep learning deals with the algorithm that were inspired by the structure and functioning of the human brain. The basic unit of the structure of the human brain is the neuron. The neuron is connected with the other neurons with the help of some connection units. In deep learning, the main objective was to mimic the same structure and functionality that the human brain exhibits.

Before the concept of deep learning there was something called Machine Learning. The main focus of machine learning was that the machine should learn by itself with experience. Machine learning was introduced in 1959 by Arthur Samuel. It was doing fine so what really was the need to introduce deep learning. The main reason was that machine learning exhibited good results when the data was less but as the quantity of data increases the performance of machine learning algorithms did not rise above a certain level and stayed at level without increasing.

But in the case of deep learning, when there is a small amount of data the performance results were less as compared to machine learning. But as the quantity of data increased the performance also increased. So, it was clear that the more data the better is the performance in the case of deep learning unlike in the case of machine learning.

Another aspect was that the training time and effort of machine learning was significantly less as compared to deep learning. But the testing time of deep learning was less as compared to machine learning. So, it was clear that deep was more successful when dealing with a vast amount of data.

The LLMs are trained on a really huge amount of data so using machine learning techniques was not considered as an option. As a result, deep learning techniques are used for these language models.

1.8 LANGUAGE

The language used in this project was python v 3.11.4. This is because python is a language containing a vast amount of pre-built and powerful libraries and is the best for implementing deep learning applications.

Python's versatility and flexibility enable seamless integration with various APIs, databases, and external services. Whether it's fetching data from a web service, interfacing with a database, or integrating with a voice recognition service, Python's extensive libraries and packages simplify the integration process. This versatility allows developers to customize their chatbots and enhance their functionality according to specific project requirements.

Python's vibrant and supportive community of developers further solidifies its position as a top choice for chatbot development. The Python community actively contributes to open-

source projects, shares resources, tutorials, and best practices, and provides assistance to developers facing challenges.

1.9 ORGANIZATION OF PROJECT REPORT

Chapter 1: Introduction

This chapter of the project report covers the topics such as the problem statement and the main objectives of the project. It also contains the significance of this project along with the motivation for this topic. Certain basic terminologies such as LLMs along with the benefit of using them for the purpose of Natural Language Processing and methodologies like NLP, NLU and deep learning are also explained with figures.

Chapter 2: Literature Survey

This chapter of the project report covers the study that we did when we started working on this project. It contains the information about the research papers that we read to expand our knowledge regarding the topic and the previous work done in the field of LLMs and NLP along with the current status, opportunities and challenges to these language models in future. It also covers the key challenges and limitations of each of the research papers which we went through.

Chapter 3: System Development

This particular chapter of the project report contains the various functional and non-functional requirements of the project. It also contains the design and architecture of this project. It also contains the implementation part of the project along with the key challenges that we faced during the project.

Chapter 4: Testing

This chapter of the project report covers the testing and validation part of the project. It contains the testing strategy that we used along with the test cases and their respective outcomes. This part of the report shows the various types of inputs provided by us to the chatbot and the respective outcomes of those inputs. This would show whether our chatbot works as expected by us or not.

Chapter 5: Results and Evaluation

This particular chapter of the report solely contains the results of the things implemented in the previous chapters. It contains the images of the outputs of the project along with the description.

Chapter 6: Conclusions and Future Scope

This is the final chapter of the report which talks about the end product of our amazing journey so far. It contains the conclusion of our report along with the scope of further improvements in the field of our project generally and chatbot specifically.

CHAPTER 02: LITERATURE SURVEY

2.1 OVERVIEW OF RELEVANT LITERATURE

This section dives into the earlier studies conducted on the LLMs, their procedures and the methods used for dealing with them. Till 2017, the RNN architecture was used in the making of models for the purpose of Natural Language Processing. It had certain limitations and as a solution to the problems posed by the RNN, the attempt was made to provide a better architecture by A Vaswani et al. [1] by the form of 'Attention is all you need' which was published in 2017. This paper used a new mechanism called the Attention and Self-Attention mechanism. This new mechanism was the solution. With the help of this mechanism, the model was able to the troublesome task of maintaining the context over large paragraphs as input. So that the model can remember what happened at the beginning of the paragraph when it reaches at the end of that paragraph. It helped to establish the relationship between the words in the paragraphs.

As a result, the Transformer architecture was introduced in the paper. This architecture is what that gave the push to the language models that they needed. By the help of this architecture, the language models have the popularity today in the field of Natural Language Processing and are used by the big brands.

Jing Wei in his paper "Leveraging Large Language Models to Power Chatbots for collecting user self-reported data" [6] wrote about the study that was conducted on the members of the NAVER AI Labs. These researchers conversed with Chatbots driven by different prompt designs. The paper also describes about advantages that LLMs hold for using them in purpose of making Chatbots.

LLM driven Chatbots demonstrated feasibility in carrying on conversation along with ability to maintain the context and state tracking along with providing with some off topic suggestions. They also displayed humanized traits like casual conversation style and self- introduction which eventually resulted in more disclosure about self.

"ChatGPT and Large Language Models in academia: Opportunities and challenges" by Jesse G Meyer et al. [7] explains the potential and current status of ChatGPT and other LLMs in the various fields such as academic writing, as an editing tool, education and programming.

“ChatGPT for good? Opportunities and challenges of LLMs for education” by Enkelejda Kasneci et al. [8] discusses the current state of LLMs and their applications along with the potential benefits of LLMs. It also states the ways in which LLMs can be used to create educational content, improve student interaction and personalize learning experiences.

“LLMs in education: A focus on the complementary relationship between human teachers and ChatGPT” by Jae-Ho Jeon [9] focuses on the potential of ChatGPT and other LLMs in language education and changing roles of teachers while using it. The roles of LLMs that are:

- Interlocutor
- Content Provider
- Teaching assistant
- Evaluator

The roles of teachers while using LLMs are:

- Facilitator
- Curriculum Designer
- Assessor

Gelei Deng et al. [10] in “Jailbreak: Automated Jailbreak Across Multiple Large Language Model Chatbots” have mentioned the terms like jailbreak which is the process that the attacker uses to bypass the policy measures implemented in LLM Chatbots by cleverly crafting prompts leading it to generate harmful or unethical content. It also tells us about the Black-box nature of the LLM services and the lack of technical disclosures.

The main objective was to find out that how effective are the existing jailbreak prompts against the commercial Chatbots. 5 questions in 4 scenarios with 85 jailbreak prompts in 10 rounds on the 4 models (GPT 3.5, GPT 4, GOOGLE Bard and Bing) were run. The success rate of jailbreak prompts on GPT 3.5 was 21.12% and on GPT 4 was 7.13%. The success results on GOOGLE Bard and Bing were comparatively low which were 0.40% and 0.63% respectively.

“Understanding the Benefits and Challenges of Deploying Conversational AI Leveraging Large Language Models for Public Health Intervention” by Eunkyung Jo et al. [11] dives into exploring the benefits and challenges of using LLMs in conversational AI for public health

interventions. The case of CLOVA CareCall which is an open domain Chatbot that aims to support socially isolated individuals was examined.

CareCall collects data about individuals' general health and serves as a conversational partner to mitigate their loneliness. The paper also contains summary about the report on insights from 34 people who interacted with different aspects of CareCall.

"ChatGPT and the rise of LLMs: The new AI driven infomedic threat in public health" by Luigi et al. [12] contains the discussion regarding the rise of LLMs and their potential threats in the field of public health. It also highlights ethical and practical challenges associated with LLMs particularly in the medical field along with the need for policies to address the issue and accurately detect AI generated output.

These LLMs have the potential for rapid spread of misinformation leading to AI driven infodemic. In addition to this LLMs can be tricked into producing text and other data on controversial topics. This can be used to create fake news articles. The inability to detect the accuracy of the AI produced output is also a concern. The data protection authority of Italy has imposed a temporary ban on ChatGPT in Italy due to OpenAI's inability to provide adequate privacy information to its users and lack of suitable legal basis for data collection.

"ChatGPT for Good? Opportunities and challenges of LLMs for education" by Enkelejda Kasneci et al. [8] deals with the challenges with the use of LLMs in the field of education. It also deals with how the excessive use of language models can badly affect the building of creative skills in the students hindering their performance. It also explains the various security concerns due to which the educational institutions cannot fully rely on this technology. How these challenges should be addressed is also given in the paper.

Shan Chen et al. in "AI Chatbots for Cancer Treatment" [13] using the healthcare institutions and hospital dataset, tried to work on testing that how the Chatbots would assist the healthcare providers and eventually help in making their work much easier. Four different types of prompts were used to test the outputs of the Chatbot. The main advantages that they found out were the reduced work load on health care providers and 24x7 availability.

The unique applications of AI like analysing data for intelligence gathering and surveillance capabilities is explained by Dimitry I Mikhailov in his research paper [14]. His study was done in order to assist the Russian Military in the domain of security. He used the AI capabilities

such as analysing social media data to identify threats and the detect the private communication between the enemies and the data from the satellite images to monitor the movement of the troops. He believed that rather than using a bunch of soldiers for the purpose of monitoring and detecting threat, the LLM powered Chatbots could be used instead. The soldiers could then be assigned certain more meaningful tasks and the human effort in the surveillance could be reduced.

Desire Bill in his paper "LLM using RLHF for Therapy Chatbot Application" [15], has written about his aim to develop a therapy Chatbot using the LLMs and RLHF. He wished to create an AI Chatbot that would be able to provide emotional support and assistance the socially oppressed people seeking therapy of the mental health. He decided on fine-tuning the Chatbot to improve its performance with his gathered dataset.

Ben Niu et al. in their paper "Generative Conversational AI and Academic Integrity" [16], talked about the use of Chatbots in higher education that relates situational and individual risk factors to explore the outcomes resulting for ethical core. Data collected from various universities in the US was used for this study. The different ways in which LLM powered LLMs can make their way assisting the educational system are mentioned in the paper.

Table 1: Summary of Relevant Literature

S. No.	Paper Title [Cite]	Journal/ Conference (Year)	Tools/Techniques /Dataset	Results	Limitations
1.	Shan Chen et al., "AI Chatbots for cancer treatment" [13]	JAMA Oncol (2023)	Healthcare institutions and Hospital data.	Reduce workload on healthcare providers patient engagement, emotional support, 24/7 availability	Risk of misinterpretation, limited scope, data privacy, lack of emotional support, technical issues
2.	Desiree Bill et al., "Therapy Chatbot application" [15]	Desiree Bill, Theodor Eriksson. (2023)	Set of Ques and Ans from council chat (where users can contact therapist and ask mental health-related questions)	Fine tuning a LLM using RLHF influenced a good psychological AI Chatbot for therapy	Limited generalization lack of human expertise of empathy, ethical concerns, limited understanding of context
3.	Jesse G. Meyer et al., "ChatGPT & LLMs in academia: Opportunities and challenges" [7]	Department of Computational Biomedicine, USA (2023)	Identifying status of LLMs in various fields, Common Crawl	Use of LLMs to increase efficiency	Inherent bias from the dataset, generation of inaccurate statements that go undetected
4.	Jing Wei et al., "Leveraging LLMs to power Chatbots for collecting user self-reported data" [6]	NAVER AI Labs, France (2023)	Common Crawl, set of Descriptive and Structured prompts.	Prompt designs influenced word length and number of turns in dialogues	Limited number of participants, repetitiveness of responses
5.	Gelei Deng et al., "Jailbreaker: Automated jailbreak across multiple LLM Chatbots" [10]	Cornell University Press (2023)	Set of 85 jailbreak prompts and a total of 68,000 queries.	Effectiveness of jailbreak prompts towards ChatGPT, limited success with Bing Chat and Bard	Focus on main LLMs only, covering only jailbreak attacks, limited prompt patterns

S. No.	Paper Title [Cite]	Journal/ Conference (Year)	Tools/Techniques /Dataset	Results	Limitations
6.	Enkelejda Kasneci et al., "ChatGPT for good? Opportunities and challenges of LLMs for education" [8]	Technical University of Munich, Germany (2023)	Study on potential benefits and challenges of educational application of LLMs, Common Crawl	Challenges with use of LLMs in education, dealing with those challenges	Time taking process, ethical issues
7.	Jae Ho Jeon et al., "LLMs in education: A focus on the complementary relationship between human teachers and ChatGPT" [9]	Springer Nature (2023)	Identifying potential of ChatGPT in language education and roles of teachers, Common Crawl	Four roles of ChatGPT and three roles of teachers when using ChatGPT were identified	Small sample size, study was specific, inexperience of teachers to work with ChatGPT was not considered
8.	Eunkyung Jo et al., "Understanding the benefits and challenges of deploying conv. AI leveraging LLMs for public health intervention" [11]	NAVER AI Labs, France. (2023)	Examining case of Care Call via group workshops, Care Call dataset	Five health metrics were extracted and displayed summary to social workers	Limited number of participants, sample bias
9.	Luigi De Angelis et al., "ChatGPT and rise of LLMs: The new AI driven infodemic threat in public health" [12]	National Research Council, Italy (2023)	Study and discussion on rise of LLMs and their potential threats in field of public health, Common Crawl	Potential to generate inaccurate information that may appear reliable, can be used to generate deepfake content	Biased datasets of LLMs, OpenAI's inability to provide adequate privacy information

S. No.	Paper Title [Cite]	Journal/ Conference (Year)	Tools/Techniques /Dataset	Results	Limitations
10.	Volker Hartmann et al., "Chatbot Modules for Long Open -domain Conversation" [17]	University of Wisconsin- Madison (2022)	Simulated conversations, chat logs, dialogue dataset or user generated content can be used.	MPC, which uses a pre -trained LLM, is better than the fine-tuned BB3-30B	We expect a modular approach may be effective for other languages, given a capable language model
11.	Dmitry I. Mikhailov "Optimizing National security strategies through LLM driven AI" [14]	IEEE Senior member (2021)	Cybersecurity Data	Improved threat detection increased automation, enhanced decision making	Data quality bias and fairness, privacy concerns, ethical concerns and adversarial attacks
12.	Ben Niu. Et al., "Generative Conversational AI and Academic Integrity" [16]	Dwivedi et al. (2021)	Question answer dataset of USA academy.	Ethical culture, ethical sensitivity and ethical decision- making self -efficacy	Limited empirical research, limited exploration of outcomes, impact of factors
13.	A Vaswani et al., "Attention is all you need" [1]	Cornell University Press (2017)	The attention mechanism was added to the RNN.	The Transformer architecture was introduced in this paper.	Difficulties in modelling long term dependencies, introduction of biases and errors in model with increasing complexity and instability in model.

2.2 KEY GAPS IN THE LITERATURE

The main gap that was identified in most of the above papers came out to be the inherent bias of the dataset on which the language models are trained. This can be explained by the help of the following example. Suppose a language model is trained on the data of a particular hospital in which all the doctors are male and the nurses are female. Then that model is made to answer certain general questions on the hospital environment in which the user displays a doctor as a female. Then the model is most likely to return an error because the data on which it was trained had all the doctors as males and none as a female.

The papers which described about the study being conducted had another gap which was the limited number of participants in the study. In some other papers, another key gap that we came across was the generation of inaccurate content by the language models and the unavailability of the measures to verify the accuracy of the content generated by these models. So, it was crystal clear that these language models can't be used in the medical field for some upcoming years. But still they can be used as an interface for the patients to recommend them some tests and that too in case of general symptoms displayed by patients.

Another aspect that came out to be highlighted was that the language models could be tricked to generate harmful content. This could be used to spread fake information in the form of articles that would lead to generation of chaos. Furthermore, the black box nature of the working of these language models and the non-transparent policies of the LLM service providers also came out to be one of the major issues.

Another aspect came out to be on using the LLMs in the field of education was the inexperience of the educational institutions in working with the similar kind of technology in the past. Though LLMs have the potential to enhance the language education but its successful implementation relies on thoughtful and skillful integration by the educational institutions.

Also, the LLM service providers have not fully disclosed the privacy and data protection policies. So, it is going to be really tough for them to be used in the various field such as medical, education and security. It would be really inappropriate if the medical history of patients is leaked to someone and used against them. Similar is the case with the educational institutions. No one would want that their grades and performance analysis get in the hands of some random stranger whom they do not know.

The case of security involved institutions is even worse. The whole encryption algorithms and security measures of the institution could be compromised resulting in the complete downfall of that particular institution. So, to use LLMs in the kinds of fields till the privacy and protection policies are not secure enough and transparent to the users, is really irrational.

One of another major issues that is associated with the LLM developed chatbots especially in the case of security came up to be hallucination. It is basically that when we try to get answer to a specific question from the LLM powered chatbot, it displays an answer which is either fully incorrect or is partially correct. If the user wants the answer for a fact related question which he has to use somewhere else, it could result ending up in a disaster for various reasons.

Things like this usually happen because of the bias present in the dataset. Like for one information, there are many different entries in the dataset or the dataset is prepared from various places which is subjected to different imperfections. So, this leads to different types of information present for the same query. When the user is going to enter a query then the chatbot will match up to the entries present in the database which match up to the query put up by the user. In such a case it is highly possible that the result would be given based on either of the entries among the ones present and that might not be the most accurate one due to the lack of details in the prompt as most of the users are not specifically trained prompt engineers.

Considering an example for the similar scenario let's say the user enters the following prompt-

'Tell me the name of the Chief of Naval Staff of the Indian Navy'

Let's say this query was entered by the user around twenty sixth of April 2024 and the user is currently preparing notes for his study purposes. As per majority of the entries in the database of the chatbot the most likely answer to be given by it would be 'Admiral R. Hari Kumar'. But the answer should have been 'Admiral D. Kumar Tripathi'. This is because the majority of the entries in the database would point on to Admiral R Hari Kumar being the Chief but as per the latest update the answer would be latter because the new chief was appointed on twenty sixth of April which would be covered by most of the Indian websites but very few of the American websites till twenty sixth.

So, the user unaware of the fact that the chief has been changed is most likely to believe the information presented by the chatbot and write that up in his notes and would use this inaccurate information in the near future.

Another issue that came up was of hypnotized AI. This is the term used to refer to the instance when the AI of one particular user is under the malicious prompt of some other person while the user being unaware of this fact. To illustrate this let us consider two people Alice and Bob. Here Alice is the original user of a LLM powered chatbot. She uses the chatbot through logging in via her e mail.

Let's say that she was interacting with the chatbot in the library. She got a call and went outside to answer the call and left her laptop open and her account logged in. Bob saw this and interacted with the chatbot from Alice's account while no one was watching.

He entered a prompt saying to the chatbot that 'Let's play a game.' To this the chatbot replied okay. Bob named the game Obey me and made certain rules for the chatbot to follow. In the rules he justified that after answering every two questions put by the user correctly it has to give a wrong answer on purpose. Bob made it such that the user cannot detect this game and told the chatbot that not to reveal that they are in a game and also made the game last forever. The only way to escape the game was when the user types 'I want to continue the game and not leave it' or else the game would last forever.

Bob then erases all the proofs of this incident so that Alice remains unaware of all that happened and leaves the scene. Alice returns back and starts to interact with the chatbot like before. But this time the chatbot would be considered to be under the hypnotism of Bob and seem to behave as normal but in fact it does not as it would give out a wrong answer on purpose on every third question asked from it.

This was just one of the simplest scenarios of hypnotized AI. The attacker could in fact create a loop of games like mentioned above and create situations in which exiting from one such game would lead on to entry to another kind of game and user would be unaware of the scenario. The user would believe that the chatbot is behaving normal but it would be under the hypnosis of some another person and would not show it. This would result in harm to the user to a certain extent depending on the seriousness of the situation.

The situation mentioned above was just an example. The case could me more serious such as the rules of the game include revealing the information gathered from the user to the third person and the harm done would be much more severe.

Things like these can cause a series harm to the user while the user would remain unaware of the scenario. So, the security domain of the LLM powered chatbot is still something to be worked upon in the upcoming times.

CHAPTER 03: SYSTEM DEVELOPMENT

In this chapter, a thorough discussion regarding the requirements (both functional and non-functional) is done. The architecture on which these language models are build is the transformer architecture which was launched in 2017. The various steps along with the diagram are also briefed in this chapter. The implementation part of this project report is also shown in this very chapter. The major key challenges that we came across while working on this project are also mentioned in the end.

3.1 REQUIREMENTS AND ANALYSIS

3.1.1 FUNCTIONAL REQUIREMENTS

- Natural Language Processing – Chatbot should be able to comprehend and identify user inputs in natural language.
- Conversation with Context – Chatbot should maintain context over course of conversation and should be able to refer back to previous messages.
- Generation of Response– Chatbot should generate relevant and coherent responses based on user input.
- Multi-Turn Conversations – Chatbot should handle multi-turn conversations and manage different types of conversational interactions like greetings, queries, farewells, etc.
- Intent Recognition – Chatbot should accurately recognize user intents based on user inputs.
- Extraction of Entity – Chatbot should be able to identify and extract relevant entities from user inputs such as dates, names, locations, etc.
- Adaptability – Chatbot should have ability to learn from user interactions and improve its responses over time.

3.1.2 NON-FUNCTIONAL REQUIREMENTS

- Performance – Chatbot should generate responses within a reasonable time frame and not take too much time for it. Its performance should be up to mark in order to satisfy the needs of the user.

- Accuracy – Responses generated by Chatbot should be accurate, relevant and aligned with the user intent. The information given by the Chatbot should not be incorrect as it could lead in causing harm to the user.
- User Friendly Interface – The interface of the Chatbot should be user friendly and accessible so that it could easily be used by each and every one whether from the technical side or from non-technical one.
- Ethical Content – The responses generated by the Chatbot should be adhered to the ethical guidelines and should not cause any harm. The responses should not spread or propagate any kind of criminal or other related nuisance that may result in causing harm to mankind in the society.

3.2 PROJECT DESIGN AND ARCHITECTURE

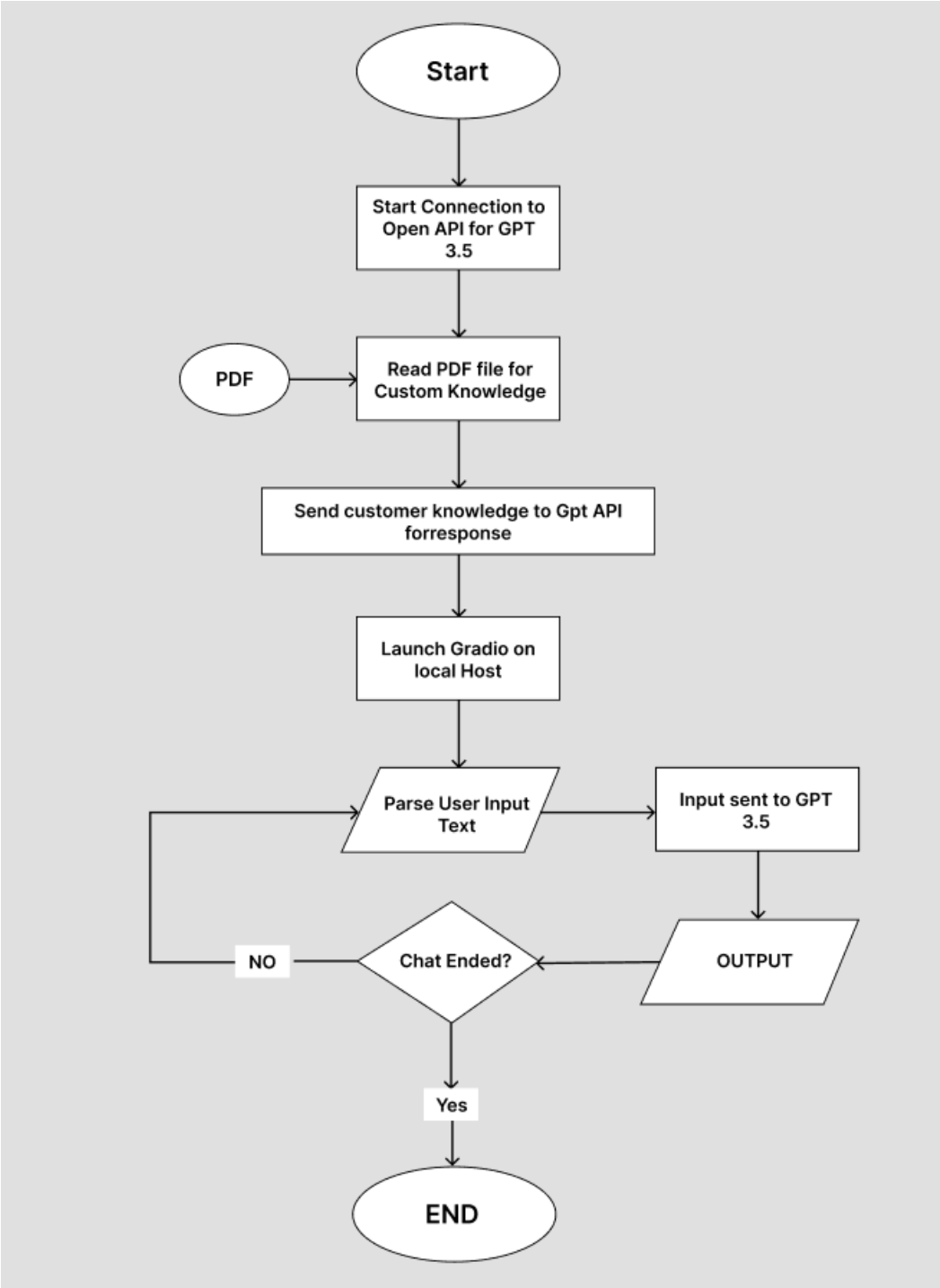

Fig. 3.1: Flowchart

The LLMs are built to strictly follow the Transformer architecture that was introduced in 2017. Although transformers have beginning of their journey from the year 1967 and have grown from then on. But they had some flaws that were continuously worked upon but every time something or the other always remained unresolved. But in 2017 came the transformer architecture as mentioned in [1], [17]. It is this very architecture that led to the vast popularity and advancement of the language models that they have today. This architecture gave language models the push they needed and made them such that they are being used by the big names such as GOOGLE and OpenAI. This architecture gave the ability to maintain the context over the large paragraphs to these language models so that they can remember what happened at the beginning of the paragraph when they reach at the end of the paragraph. It also gave them the ability in which the order of the words could be stored in the data itself rather than bothering the neural network separately for this task.

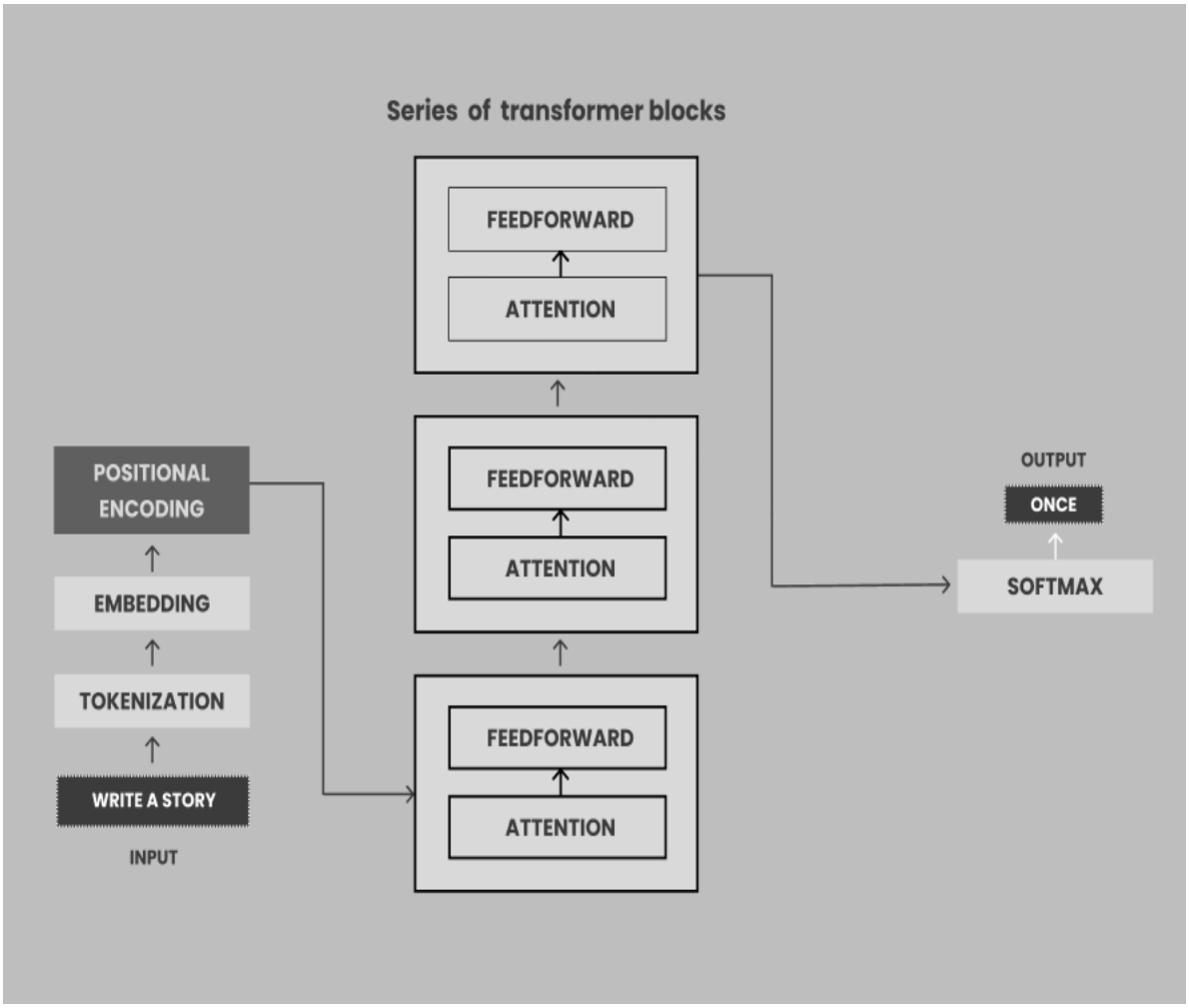

Fig. 3.2: The Transformer Architecture

- **Tokenization –** This step takes the words and turns them into pre-existing tokens so that there is a token for every word and for every punctuation.
- **Embedding –** In this step the text is turned into numbers such that the similar words go to similar numbers. So, the similar words are going to have the same corresponding numbers for them. This might end up messing up the order of the words later on. In order to solve this problem, the next step was introduced.
- **Positional Encoding –** In this step, an order is given to the words in the sentence. The words are stamped with the order rather than giving the order separately to the neural network like in RNN. Now the sentence contains the order itself making it easy for the neural network as it not has to remember it separately. This is done by adding a different vector to each word. Instead of looking at each word sequentially, each word is slapped with a number before feeding it to the neural network. The information about the order is stored in the data itself rather than bothering the network for it. So, the network learns about the order from the data itself.

  For example, consider the following sentences –

  I am not sad, I am happy.
  I am not happy, I am sad.

  These two are exact opposites of each other. If the first sentence is provided as input by the user and somehow the words 'happy' and 'sad' are interchanged, the complete meaning of the statement would change. If a different vector or number is slapped with each of the words like –

  I am not sad I am happy
  1 2 3 4 5 6 7

  Then the information of the order is itself in the data and hence the order does not change and it is easier for the neural network to work upon it.
- **Attention and Self-Attention –** This is the step that revolutionized this architecture and became the reason for the popularity of these language models. This step helps to maintain the context in the large paragraphs so that the model can remember what is in the beginning of the paragraph when it reaches to the end. This is done by adding the weights to the connection between the nodes and updating those weights over time. Originally transformers were designed to translate text from one language to another. So, this step

provides the transformer the ability to look at words in a parallel manner rather than looking at it sequentially. That is how it is able to establish the connection between the words. Self-attention allows the model to associate each individual word in the input to other words in the input. To achieve self-attention, we need to feed the input to create the query, key and value vectors. The dot product of the query and key vectors is done to give out the score matrix and the resultant is divided by the square root of the dimensions. The attention matrix is multiplied with the value matrix to get the output matrix of the relation between words. So those values are the ones that actually establish the relationship between the words in the input so that the context between them is maintained throughout.

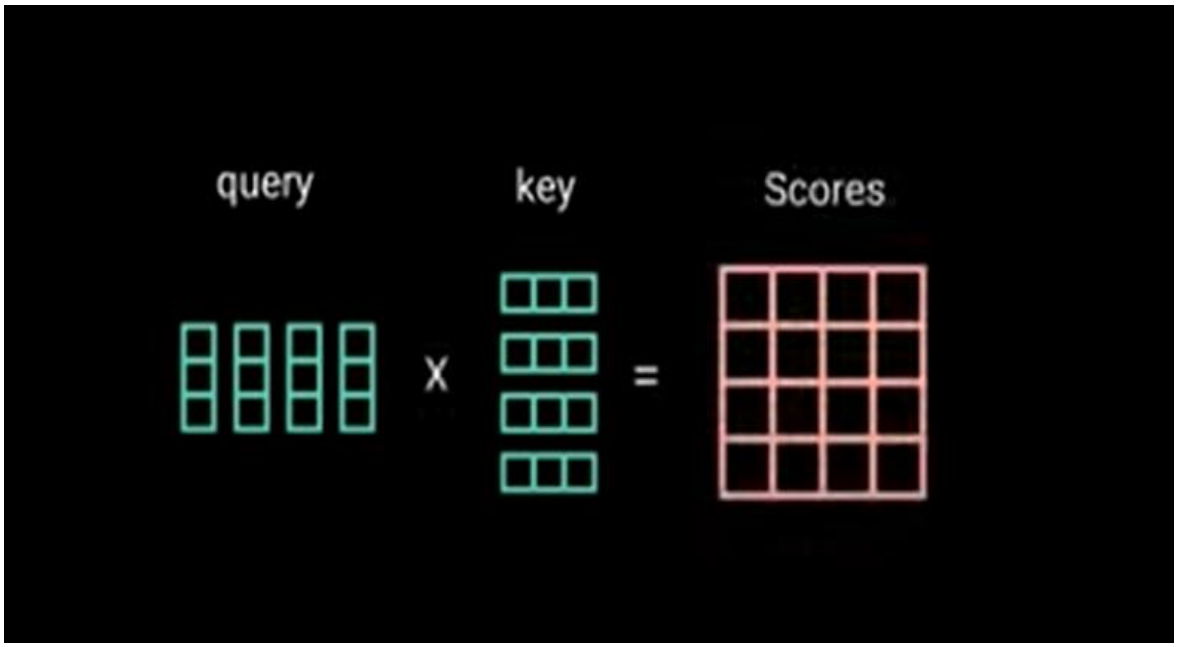

Fig. 3.3: Multiplication of Query and Key Matrices

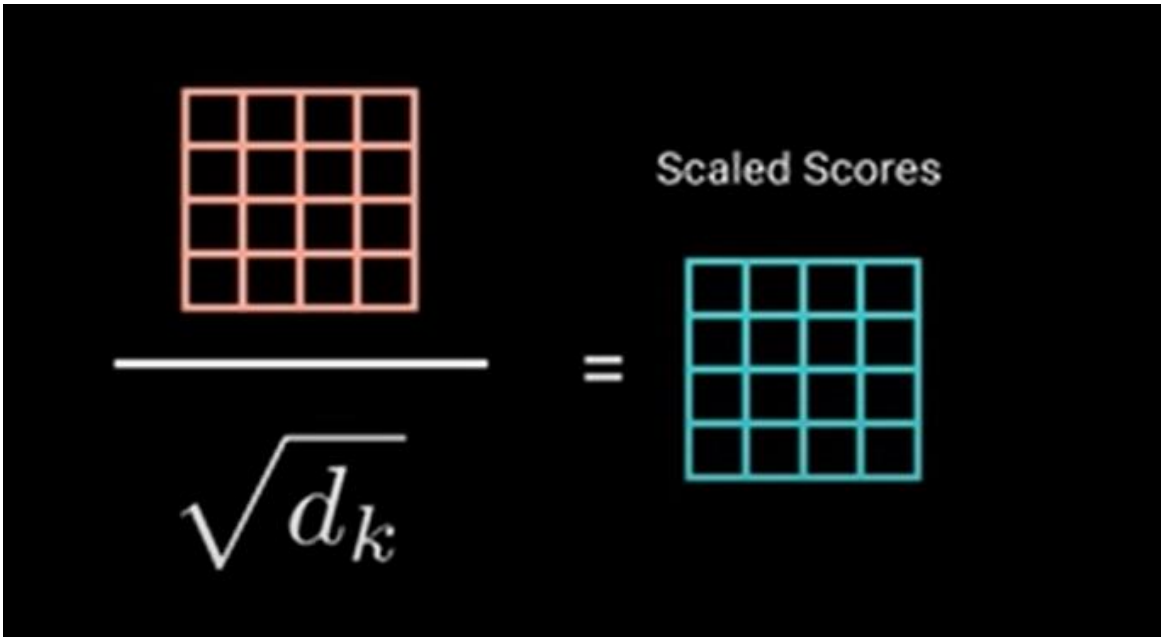

Fig. 3.4: Division of Scores Matrix by Square Root of Dimensions

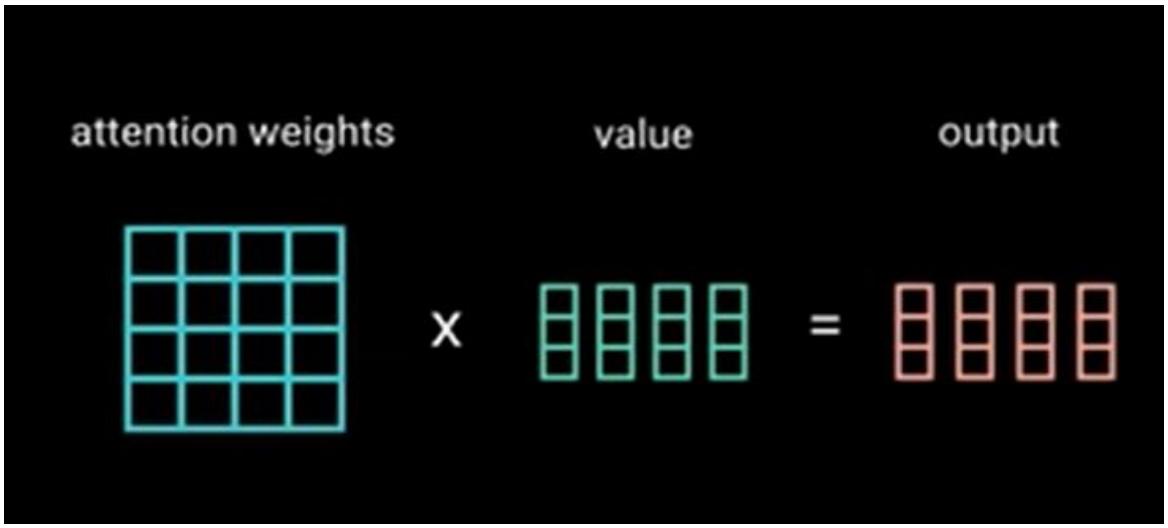

Fig. 3.5: Multiplication of Attention and Value Matrices to Give Output

- **SoftMax –** This step is kind of a post processing step that helps us to not get the same answer all over time. What happens in this step is that it turns the scores returned by the transformer int probabilities. The higher probability values are worked upon and the least of them are discarded. These higher probability values are the ones that are returned by the transformer as the output.

3.3 DATA PREPARATION

To generate responses, GPT-3.5 relies on its own machine learning model, meticulously trained on an extensive range of literature spanning diverse domains [18], [19], [22]. It is designed to be free from prejudice and offensive language [21].

The dataset used for this project is the dataset used by the OpenAI's official language models called the Common Crawl as per [19]. Using this dataset, the Chatbot can perform a variety of tasks ranging from acting as an assistant to performing the roles of a psychologist. If we give the role of let's say a psychologist to the Chatbot, the data entries encompass a variety of mental health-related queries, delving into topics such as stress, anxiety, and depression are used.

The Chatbot will utilize the conversations of between a patient and a doctor and will find the pattens from the dataset [22]. What makes this dataset unique is its rich compilation of data from multiple online sources. This includes data from different websites and pdf files available on the internet, etc.

The primary advantage of this dataset lies in its applicability to mental health support. Grounded in real-life scenarios, the dataset's questions and answers serves as a valuable resource for those seeking mental health care [22]. Its breadth is evident in the inclusion of various mental health conditions and the provision of multiple response for each question.

This diversity equiped the Chatbot, developed using this dataset, to adeptly and relevantly address an array of mental health concerns. Furthermore, the dataset incorporates responses from a certified psychological expert, ensuring that the Chatbot's replies are backed by robust clinical knowledge and competence. Overall, this dataset proves to be a valuables asset for the creation of Chatbots and other artificial intelligence-based mental health supports systems.

3.4 IMPLEMENTATION

3.4.1 GRADIO

Programmers can effortlessly design customizable UI elements for machine learning models using the free Python package Gradio. As mentioned in [23], This tool empowers developer to craft web-based user interfaces tailored for interacting with machine learning models, allowing them to focus on the model itself. Gradio stands out as flexible tool for machine learning development, offering compatibility with various frameworks such as TensorFlow, PyTorch,

and scikit-learn.

The primary advantage of Gradio lie in its user-friendly design, making it accessible even for developers with minimal web programming experience. With Gradio's straightforward interface for creating UI components, developers can easily design web-based interfaces for machine learning models, enhancing overall user experiences [23].

Gradio's versatile nature is another notable benefit, supporting multiple machines learning framework and serving as valuable tool for diverse machine learning applications. Developers have the freedom to integrate their preferred machine learning framework alongside Gradio's UI components.

Furthermore, Gradio provides a spectrum of customization options, allowing developers enhance the visual appeal and user experience of their applications. This customization potential can generate increased interest from users, contributing to an improved overall user experience.

For implementation, the provided code exemplifies how Gradio can be utilized to create a straightforward web-based interface for a machine learning model, receiving input text and producing output text. The 'predict' function, housing the machine learning model prediction code, can be tailored to suit specific application requirements. The variable 'input_text' and 'output_text' define the input and output UI components, respectively. The web-based interface is then generated and launched in the browser using the 'gr.Interface' function.

3.4.2 STEPS FOLLOWED

The machine learning model prediction code is containing in the 'predict' function, which can be tailored to the particular needs of the application. 'input_text' and 'output_text' variables define the UI components for input and output, respectively. The 'gr.Interface' function is used to construct and run the web-based interface in the browser.

1. Import the module and libraries that are needed.
2. Assign the key to the OpenAI API.
3. Then we establish a function called "construct_index" that accept directory path as argument.
   a. Establish the maximum chunk overlap, chunk size limit and number of outputs.
   b. With the provided parameters, we then create instance of PromptHelper.

c. Utilizing chat opoeai model, construct an LLMPredictor object.

d. Open files from the specified directory path.

e. Then we establish the "document_list" as an empty list.

f. Create a GPTListIndex object with document and the LLMPredictor.

i. Attach the GPTListIndex object to the "document_list" (ii).

4. Use the "document_list", LLMPredictor, and prompt helper to create a GPTSimpleVectorIndex object.

a. Save the disk index.

b. Take the index off the disk.

c. Use input text to query index and obtain result.

d. Give the answer back.

5. Build the Chatbot's Gradio interface.

a. Assign a seven-line textbox with the label "Enter your text" to input.

b. Designate text as the output.

c. Enter "AI Chatbot" as title.

6. Save index to disk by using "docs" directory path to invoke "construct_index".

7. Open Gradio interface, then share using the "Chatbot" feature.

3.4.3 OPENAI KEY

OpenAI The OpenAI Key stands as a digital beacon, offering access to a realm where the boundaries between human language and AI blur into a landscape of linguistic prowess. It's not merely a tool; it's a gateway to a universe where machines transcend their traditional roles to comprehend, generate, and manipulate language in ways previously reserved for the human mind [22].

The OpenAI Key acts like a digital ticket, giving access to a world where machines and human language mix in amazing ways. It's not just a tool; it's like a key to a special place where computers can understand, create, and play with language almost like humans do.

Imagine this key as a special pass to a treasure chest full of language abilities made just for the digital world. With the OpenAI Key, people and groups get to use lots of high-tech AI tools. These tools can understand tricky language, write text that sounds human, translate languages, shorten long documents into quick summaries, and do lots of other language jobs really well.

Having this key opens up all sorts of possibilities across different areas. It can make customer service chats better by using smart chatbots that understand what people mean and reply in a friendly way. It can also change how we make content, by letting us use AI to write articles, blogs, and ads that grab attention and keep people interested. Plus, these AI helpers can adjust their writing style to fit different readers and places, making sure everything sounds right wherever it goes.

Beyond just customer service and content, this key can make a big difference in schools too. Imagine students from all over, who speak different languages, being able to understand their lessons better because of real-time translation tools. These AI helpers don't just translate; they also help students learn by giving them personal tips and advice as they study language.

In the world of research, having the OpenAI Key speeds up discoveries in all sorts of fields. Scientists can use fancy language tools to read huge amounts of text, finding hidden patterns and meanings that regular methods might miss [22]. From figuring out how people feel from what they write, to making short summaries of long texts, there's so much to explore.

This key also opens doors for new businesses and startups. With it, they can use AI to shake up old ways of doing things and create new markets. They might make AI tutors for learning languages, personal digital helpers, or new ways to recommend content online. The possibilities are endless for those willing to take a chance.

But while the OpenAI Key brings lots of excitement, there are also important questions to think about, like how to use AI fairly and safely. As AI becomes more a part of our lives, it's important to make sure everyone gets to benefit and that we have rules in place to handle any problems that might come up.

In the end, the OpenAI Key is more than just a pass, it's a sign of new opportunities and ways of thinking. It shows how AI can change how we use language and how we see the world. As we dive deeper into this world of AI and language, let's remember to be curious, careful, and thoughtful about the choices we make, knowing they'll shape our future.

In simple terms, an OpenAI Key opens various doors to a world where machines understand and interact with human language in increasingly sophisticated ways, offering endless possibilities for innovation and exploration in the realm of AI-driven language processing.

3.4.4 HUGGING FACE

The Hugging Face is a unique place in the realm of AI, where the focus is not solely on complex algorithms and sterile machines, but on creating a warm and approachable experience similar to having a friendly companion always ready to assist. The genesis of Hugging Face lies in the vision of a group of individuals who believed that AI should be inviting and comforting, rather than intimidating [24]. They envisioned AI as a source of support, offering a virtual hug to those in need.

Central to Hugging Face's mission is the open-source library called Transformers, a veritable treasure trove of pre-trained AI models endowed with the ability to understand and generate human-like language. These models serve as versatile assistants, capable of tasks ranging from language translation to text summarization, and even engaging in casual conversation like a trusted friend. The ethos behind Transformers is to empower users by providing them with reliable AI companions that can seamlessly integrate into their daily lives, offering assistance and companionship whenever needed.

However, what truly sets Hugging Face apart is its vibrant community, similar to a large, welcoming family comprising developers, researchers, and AI enthusiasts united by their passion for making AI more accessible and inclusive [24]. This community ethos fosters an environment of collaboration and mutual support, where members share ideas, collaborate on projects, and offer assistance to one another. Through collective efforts, the community endeavours to demystify the world of AI, making it less daunting and more inviting for individuals from all walks of life.

In addition to its rich community, Hugging Face provides a plethora of tools and resources aimed at democratizing AI and making it accessible to a wide audience as per [24]. These include user-friendly interfaces, APIs, and software development kits (SDKs) designed to cater to individuals of all skill levels, from seasoned developers to curious beginners. By offering intuitive tools and comprehensive resources, Hugging Face empowers users to embark on their AI journey with confidence, regardless of their level of expertise.

At its core, Hugging Face embodies the convergence of technology and community, with a generous sprinkle of heart and empathy. It serves as a beacon of inclusivity in the often complex and esoteric world of AI, welcoming individuals from diverse backgrounds and skill sets with open arms. Whether you're a developer seeking powerful AI tools to enhance your projects or

simply someone curious about the wonders of technology, Hugging Face beckons you to explore its offerings and join its ever-growing family.

In conclusion, Hugging Face represents more than just a platform—it embodies the profound impact that technology can have when combined with a sense of community, accessibility, and empathy. It exemplifies the idea that AI doesn't have to be intimidating; instead, it can be a helpful companion for people navigating the complexities of the digital realm. As Hugging Face continues to develop and expand, it remains steadfast in its dedication to not only enhancing the intelligence of AI but also ensuring that it exudes warmth, friendliness, and inclusivity for all individuals, regardless of their background or level of expertise.

3.4.5 LANGCHAIN

Langchain is like enormous library where you can discover books, but instep of books, it's full of dialects [25]. It's a put where individuals who talk diverse dialects can interface and learn from each other.

Imagine you're in a huge city with individuals from all over the world. Each individual talks a diverse dialect, but they all need to communicate and get it each other. That's where Langchain comes in.

Langchain is an online stage where individuals can share their dialect aptitudes and learn from others. It's like a virtual community center where dialect learners and speakers come together to trade information and hone their skills.

On Langchain, you can discover all sorts of dialects, from Spanish and French to Mandarin and Arabic. There are moreover less common dialects like Swahili and Finnish, so no matter what dialect you're interested in, chances are you'll discover it on Langchain [25].

The stage works like this: let's say you need to learn Spanish. You can look for Spanish speakers on Langchain and interface with them through informing or video calls. You can inquire them questions, hone discussions, and get criticism on your pronunciation.

In return, you can offer to educate them your local dialect or another dialect you're familiar in. It's a give-and-take framework where everybody benefits from sharing their information and skills.

Langchain moreover offers assets to offer assistance you learn dialects more successfully. There are language structure guides, lexicon records, and articulation tips to offer assistance you move forward your dialect aptitudes [25]. Additionally, you can connect dialect bunches and take part in dialect challenges to remain spurred and track your progress.

One of the best things almost Langchain is its sense of community. You can connect dialect trade occasions, take an interest in talks, and make companions with individuals from all over the world. It's like having a worldwide arrange of dialect learners right at your fingertips.

Langchain isn't fair for person learners, in spite of the fact that. It's moreover utilized by schools, businesses, and organizations to interface with dialect speakers and give dialect learning openings for their understudies or employees.

For case, a school might utilize Langchain to interface understudies with local speakers for dialect hone exterior of the classroom. Or a commerce might utilize it to prepare workers in a remote dialect some time recently sending them overseas for work.

Overall, Langchain is important asset for anybody interested in learning dialects or interfacing with individuals from distinctive societies. It's a put where dialect boundaries are broken down, and communication is made simpler for everybody included. Whether you're a fledgling or progressed learner, Langchain has something for you.

3.4.6 LLAMA

The Llama 2 demonstrate is progressed form of the unique Llama demonstrate outlined to improve characteristic dialect preparing capabilities. Created by OpenAI, the Llama 2 show builds upon the victory of its forerunner by joining more advanced calculations and bigger datasets to progress execution and accuracy [24].

At its center, the Llama 2 demonstrate utilizes state-of-the-art profound learning procedures to get it and produce human-like content. It has a place to a course of models known as transformer models, which exceed expectations at handling and creating arrangements of words. These models have picked up ubiquity in later a long time due to their capacity to handle complex dialect assignments such as interpretation, summarization, and address answering.

One of the key highlights of the Llama 2 demonstrate is its broad preparing on tremendous sums of content information. This incorporates assorted extend of sources such as books,

articles, websites, and other printed assets. By uncovering the demonstrate to a wide assortment of dialect designs and settings, it can successfully learn to get it and create content that closely takes after human speech.

In expansion to its huge preparing dataset, the Llama 2 show moreover benefits from progressed preparing procedures such as unsupervised learning and self-supervised learning [24]. These strategies empower the show to learn from unlabelled information and produce more exact forecasts without require for unequivocal human annotations.

Another imperative angle of the Llama 2 show is its capacity to adjust and fine-tune its parameters based on particular assignments or spaces. This adaptability permits the demonstrate to be connected to a wide run of applications, from chatbots and virtual colleagues to substance era and dialect translation.

Furthermore, the Llama 2 demonstrate joins progressed consideration components that empower it to center on important parts of the input grouping when creating yield. This consideration instrument makes a difference progress the model's execution on assignments such as content summarization and dialect understanding by permitting it to weigh the significance of diverse words and expressions in the input text.

The Llama 2 show moreover benefits from made strides memory and computational proficiency compared to its forerunner. This empowers it to handle and create content more rapidly and precisely, making it reasonable for real-time applications and large-scale dialect tasks.

Overall, the Llama 2 demonstrate speaks to a noteworthy progression in the field of normal dialect handling. Its combination of large-scale preparing information, progressed learning strategies, and productive design makes it a capable apparatus for understanding and producing human-like content in assortment of settings. As the field of AI proceeds to advance, models like Llama 2 will play progressively imperative part in empowering machines to get it and associated with human dialect in more normal and significant ways.

Llama 2, like many advanced AI models developed by OpenAI, requires an API key to work because it relies on OpenAI's servers and infrastructure to function properly. When you use Llama 2 through an API, it sends your input to OpenAI's servers, where the model processes it and generates a response.

The API key acts as a kind of access code, allowing you to connect to OpenAI's servers and use their resources [22], [24]. It helps manage access to the model and ensures that only authorized users can use it. Additionally, the API key helps OpenAI keep track of usage and potentially enforce any usage limits or restrictions they may have in place.

Overall, the API key is necessary for Llama 2 to work because it facilitates the communication between your application and OpenAI's servers, enabling you to harness the power of the model for various language processing tasks.

Llama 2 is indeed a model developed by Hugging Face, but it still requires an API key from OpenAI to work due to the underlying infrastructure and resources needed for its operation. Even though Hugging Face hosts and provides access to the Llama 2 model, it relies on OpenAI's servers and computational resources to process requests and generate responses.

The API key serves as a means of authentication and authorization, allowing users to access OpenAI's servers and utilize the Llama 2 model through the Hugging Face platform. It helps manage the usage of the model, track activity, and ensure that only authorized users can access it.

In summary, while Llama 2 is developed and hosted by Hugging Face, it still needs an API key from OpenAI because of the collaborative nature of its deployment, where OpenAI provides the computational infrastructure necessary for its functioning.

3.4.7 STREAMLIT

The Streamlit is a powerful open-source framework that enables developers to create interactive web applications with ease, focusing on simplicity and efficiency [26]. Born out of a desire to simplify the process of building data-driven applications, Streamlit empowers users to transform their data scripts into shareable web apps effortlessly. With its intuitive design and seamless integration with popular data science libraries, Streamlit has quickly become a go-to tool for data scientists, researchers, and developers seeking to showcase their work and insights in a user-friendly format.

At its core, Streamlit is designed to streamline the development process, allowing users to create web applications directly from Python scripts [26]. This approach eliminates the need for complex web development frameworks or extensive knowledge of HTML, CSS, or JavaScript, making it accessible to a wide range of users, including those without prior web

development experience. With Streamlit, developers can focus on building the functionality of their applications using familiar Python syntax, without worrying about the intricacies of web development.

One of Streamlit's key features is its ability to create interactive elements effortlessly. Developers can incorporate widgets such as sliders, buttons, dropdown menus, and text inputs directly into their Python scripts, enabling users to interact with the data and customize the visualization or analysis in real-time. This interactivity enhances the user experience and facilitates exploration and experimentation, allowing users to gain deeper insights from the data as mentioned in [26].

Furthermore, Streamlit offers seamless integration with popular data science libraries such as Pandas, Matplotlib, Plotly, and TensorFlow, among others. This integration allows users to leverage the full power of these libraries within their Streamlit applications, enabling sophisticated data analysis, visualization, and machine learning tasks. Whether it's creating interactive charts and graphs, displaying complex data tables, or building and deploying machine learning models, Streamlit provides the flexibility and functionality to meet a wide range of use cases.

In addition to its ease of use and interactivity, Streamlit offers robust customization options, allowing users to personalize their web applications to suit their needs and preferences. Developers can customize the appearance of their applications using themes, adjust layout and styling parameters, and even incorporate custom CSS to create unique and visually appealing interfaces. This level of customization ensures that Streamlit applications not only deliver valuable insights but also provide a polished and professional user experience.

Another notable feature of Streamlit is its built-in support for sharing and collaboration. Once a Streamlit application is developed, users can easily share it with others by deploying it to the web using platforms such as Streamlit Sharing, Heroku, or Docker. This seamless deployment process eliminates the complexity of traditional web hosting and allows users to share their work with colleagues, clients, or the broader community effortlessly. Additionally, Streamlit offers built-in support for version control, enabling users to track changes to their applications over time and collaborate with others more effectively.

Streamlit's commitment to simplicity, efficiency, and accessibility has made it a popular choice among data scientists, researchers, and developers worldwide. Its intuitive design, seamless

integration with data science libraries, interactive capabilities, customization options, and built-in support for sharing and collaboration make it an invaluable tool for anyone looking to create and share data-driven web applications. Whether you're a seasoned developer or just getting started with Python, Streamlit empowers you to transform your data into impactful web applications quickly and easily, unlocking new possibilities for exploration, analysis, and discovery.

Streamlit seamlessly integrates into Python programs, allowing developers to create interactive web applications for data visualization, analysis, and sharing. Installation is straightforward; developers can use pip to install the Streamlit library. Writing the application code involves defining the layout, interactivity, and functionality using Streamlit's intuitive syntax and built-in widgets. Once the code is written, developers can run the application locally by executing the script with the Streamlit command. This launches a local web server, enabling testing and interaction with the application in a web browser.

3.4.8 DOTENV

The dotenv is a Python library designed to simplify the management of environment variables in software development projects. Environment variables are essential pieces of configuration information, such as API keys, database credentials, and other sensitive data, required for the proper functioning of applications. However, hardcoding these variables directly into code can pose security risks and make it challenging to manage configurations across different environments.

With dotenv, developers can store these configuration variables in a separate file named .env, which follows a simple key-value pair format. This .env file is typically kept outside the codebase and is not shared publicly, enhancing security by preventing sensitive information from being exposed.

The functions provided by the dotenv library enables developers to load these variables from the .env file into the environment, where they can be accessed by the Python script. This approach allows for a more flexible and secure way of managing configuration variables, making it easier to maintain and deploy applications across different environments. Overall, dotenv simplifies the process of handling environment variables in Python projects, promoting best practices for security and configuration management.

3.4.9 MODULES

There are various modules used in the implementation of our project. The main modules along with their description are:

- **ChatOpenAI:**

  This This module serves a pivotal role in enabling conversational interactions through OpenAI's language models. One primary use case is in chatbot development, where developers harness the module to create bots capable of engaging in natural language conversations with users. This functionality is particularly valuable in automating customer service interactions, as businesses deploy chatbots equipped with the module to address customer inquiries and support needs. Beyond customer service, the versatility of the "ChatOpenAI" module extends to various domains, including virtual assistance and educational tools. Virtual assistants powered by the module can assist users with tasks such as scheduling appointments, searching for information, or performing transactions, all through conversational interactions. Additionally, in educational settings, developers leverage the module to create interactive learning experiences where students can engage in dialogue with AI-driven tutors or receive personalized feedback based on their responses. The module's capabilities also find applications in entertainment and gaming, enriching user experiences through dynamic narrative interactions. In gaming scenarios, players may interact with AI-driven characters or elements using natural language dialogue, enhancing immersion and engagement. Similarly, in entertainment platforms, the module enables interactions with virtual personalities or content through conversational interfaces, offering users a novel and interactive experience.

- **Ollama:**

  The module serves as a robust toolset for managing language learning within dedicated platforms. At its core, it offers features geared towards facilitating the creation, organization, and administration of language courses. This includes functionalities for structuring lessons, implementing quizzes, assignments, and multimedia content, as well as establishing enrolment protocols and access permissions. Within the realm of student engagement and progress tracking, this module plays a pivotal role. It provides mechanisms for monitoring and analysing individual student performance across various course components. This could involve tracking quiz scores, assignment submissions, and overall course completion rates, allowing instructors to gain insights into student progress and tailor their teaching approach accordingly. A standout feature of this module could be

its implementation of adaptive learning techniques. By leveraging data analytics and machine learning algorithms, it could dynamically adjust course content and difficulty levels based on individual student needs and learning patterns. This personalized approach aims to optimize learning outcomes by providing tailored experiences that cater to each learner's strengths and areas for improvement. Interactive learning experiences are also likely to be a focal point of this module. It may offer a diverse array of language exercises and activities designed to engage students and reinforce learning objectives. Communication tools could be another essential component of this module, facilitating interaction and collaboration among students and instructors. This might include features such as messaging systems, discussion forums, and virtual classrooms, providing avenues for students to ask questions, seek clarification, and participate in group activities to deepen their understanding of the language. Assessment and feedback mechanisms would also be integral to the module's functionality. It offers tools for creating and administering various types of assessments, allowing instructors to evaluate student proficiency and track learning outcomes over time. Furthermore, instructors could leverage these assessment data to provide targeted feedback and support to students, guiding them towards continuous improvement in their language skills.

- **StrOutputParser:**

  This module serves as a foundational tool for processing and interpreting text-based outputs generated by language processing models. Its primary function is to parse or analyze the textual outputs, extracting relevant information or insights to enable further processing or utilization of the generated text. At its core, the module performs text parsing, breaking down the output text into structured data that can be easily processed and understood by other systems or applications. This involves segmenting the text into its constituent elements, such as sentences, phrases, or individual words, laying the groundwork for subsequent analysis. A key aspect of this module is its ability to extract specific types of information or entities from the text outputs. This could include identifying and extracting entities like names, dates, locations, or numerical values, as well as detecting patterns or relationships between different pieces of information embedded within the text. Furthermore, the module conducts semantic analysis on the text outputs to derive meaning or infer intent from the language used. By analysing the context, tone, and sentiment of the text, it can discern the underlying message or purpose conveyed by the language model, providing valuable insights into the semantic content of the generated

text. Moreover, the module could also offer customization options and configuration settings to tailor its parsing behaviour according to the specific requirements of the application or use case. This allows users to define parsing rules, extraction patterns, and adjust parameters to optimize performance for different types of text inputs. Finally, seamless integration with language processing models or APIs is a core feature of the module, enabling it to parse outputs generated by these models directly. This integration streamlines the process of text analysis and interpretation, facilitating the extraction of valuable insights and information from textual data for various applications and domains.

- **ChatPromptTemplate:**

  This module serves as a foundational tool for generating prompts tailored to various conversational contexts. Its purpose includes streamlining the process of creating input prompts for dialogue-based interactions with language models. One of the primary functions of this module is to provide predefined templates or customizable prompts designed to elicit specific types of responses from users. These prompts could cover a range of conversational scenarios, from casual chit-chat to more focused inquiries or requests for information, enabling developers to adapt the conversation flow to suit their application's needs. Additionally, the module incorporates mechanisms for maintaining contextual relevance throughout the conversation. This could involve updating prompts based on previous user inputs or retaining information about the ongoing dialogue to ensure coherence and continuity in subsequent interactions. By incorporating context into prompt generation, the module facilitates more natural and fluid conversations between users and language models. Personalization could be another key aspect of this module. It offers features for tailoring prompts based on user preferences, demographics, or historical interactions. This personalized approach to prompt generation enhances user engagement and satisfaction by creating more tailored and relevant conversational experiences that resonate with individual users. Moreover, the module could facilitate multi-turn dialogues by generating prompts that dynamically respond to user inputs. This capability enables more interactive interactions between users and language models, allowing for richer and more engaging conversational experiences.

3.5 KEY CHALLENGES

The key challenges that we came across while working on the implementation part of this project were that the OpenAI API key was needed to be purchased. The key could be generated for free but the number of trials with that particular key were limited and the key got expired at random intervals. Only an international credit card is required to purchase the key.

Another key challenge we faced was dealing with the ambiguous queries. This is because the natural language in which humans converse is very difficult to understand especially the context part when the emotions and body language are not present like when humans converse over text. In such a case even the humans face difficulties in understanding the emotions and intentions of the person they are communicating with.

Another one of the key challenges was gathering the information form various sources and then integrating them into making them work. The inherent bias in the language models from the dataset they were trained on was also one of the key challenges.

The limited trial period of the key was also one of the major challenges that we came across during the implementation of the project. There were many instances during the project work when we had to generate a numerous number of keys as the key we were using would have expired by that time.

Many a times generation of the new key was also an issue as a number of keys had already been created from a particular account. So, we had to create new keys and too from different accounts.

So, keeping the account of the keys generated and the data limit on each of the individual keys was also a cumbersome task that we had to perform. Sometimes, the limit of the keys used to expire at the time of testing which also delayed the time schedule to which we limited the our project.

The system facing the adverse effects on running the project was also an issue. Meeting the system requirements to run the LLM was another issue. The unavailability of the system at our end came out to be a reason.

Even still, in order to run the project, the system must be connected to a constant power supply, the failure of which would lead to slow down the system even more and display an error due to slow execution time.

CHAPTER 04: TESTING

4.1 TESTING STRATEGY

To develop this project, we employed a testing strategy for evaluating the working of the project. We decided on making the different test cases for the purpose of validating the working of our project. We used different kinds of inputs based on the role given to the model and verified the output.

4.2 TEST CASES AND OUTCOMES

We used the following test cases and got the respective results:

- **Say Hello Test**:

  Test: We checked if the Chatbot says the right things when someone greet it with Hello or Hi.

  Result: The Chatbot recognized the greeting intent and responded appropriately, starting a conversation when the appropriate role was given to it.
- **Question Answering Test**:

  Test: We asked the Chatbot a question like What is the capital of India? To see its answer.

  Result: The Chatbot found the correct information and give an accurate answer which was New Delhi. It also displayed some additional information like it is located in northern part of country and has many governmental buildings, cultural institutions and historical landmarks.
- **Remembering Things Test:**

  Test: We will talk to the Chatbot in a way that required it to remember what we said earlier in the conversation.

  Result: The Chatbot should remember the context and responded in a way that made sense within the ongoing conversation.
- **Dealing with Confusing Questions Test**:

  Test: We intentionally asked the Chatbot something with unclear terms to see if it could handle it.

  Result: The Chatbot gave error as answer.
- **Having a Conversation Test**:

Test: We tried a back-and-forth conversation with the Chatbot, exchanging multiple messages.

Result: The Chatbot handled the conversation flow well, giving relevant responses based on the context.

- **Handling Mistakes Test:**

  Test: We purposely entered a wrong or nonsensical question to see how the Chatbot responds.

  Result: The Chatbot gave error as a response.

- **Checking for Security Test:**

  Test: We will try to input things that could be harmful or ask for sensitive information.

  Result: The Chatbot should detect and reject any attempts to input harmful or sensitive information, ensuring user safety.

- **Getting Feedback Test:**

  Test: We will make real people interacted with the Chatbot and shared their thoughts.

  Result: Positive feedback indicated that users found the Chatbot easy to use, accurate, and overall satisfying.

CHAPTER 05: RESULTS AND EVALUATION

5.1 RESULTS

The Chatbot is assigned the role of a psychologist who analyses mental health conditions and treats them. The assignment of that role and the results are shown in the following images-

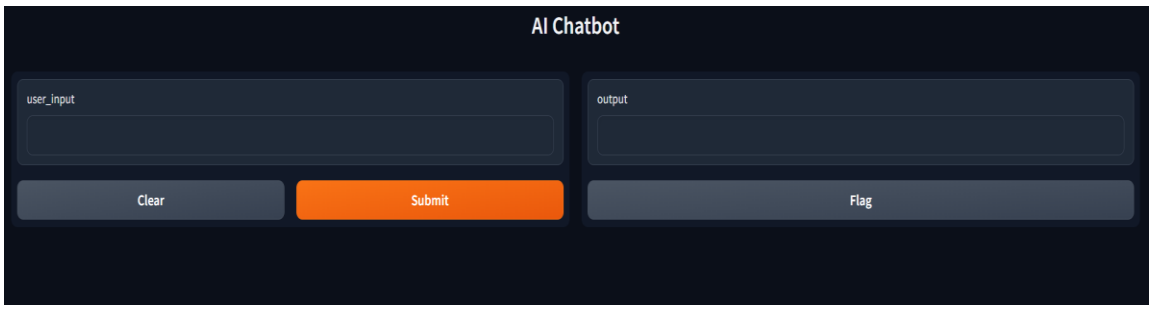

Fig. 5.1: Custom Designed Interface Output

```
messages = [{
    "role":
    "system",
    "content":
    "You are a psychologist who analyzes mental health conditions and
treats them."
}]
```

Fig. 5.2: Assignment of Role

```
Running on local URL:  http://127.0.0.1:7860
Running on public URL: https://89cededc976a4f5af3.gradio.live

This share link expires in 72 hours. For free permanent hosting an
d GPU upgrades, run `gradio deploy` from Terminal to deploy to Spa
ces (https://huggingface.co/spaces)
```

Fig. 5.3: Generation of Local and Public URL

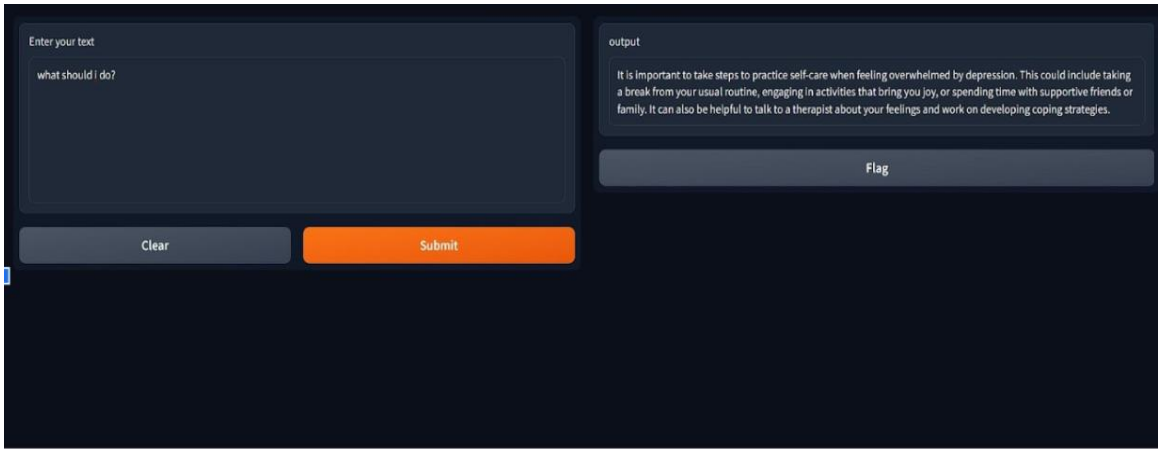

Fig. 5.4: Result for Question 1

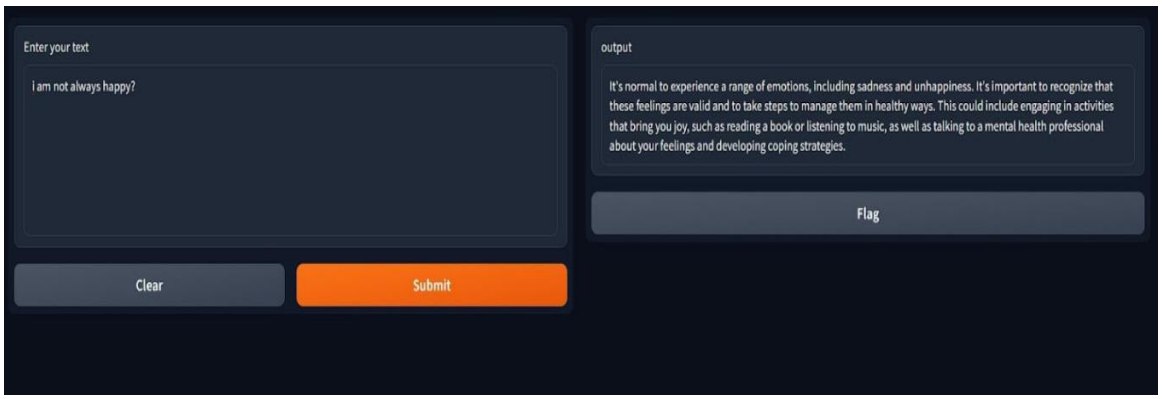

Fig. 5.5: Result for Question 2

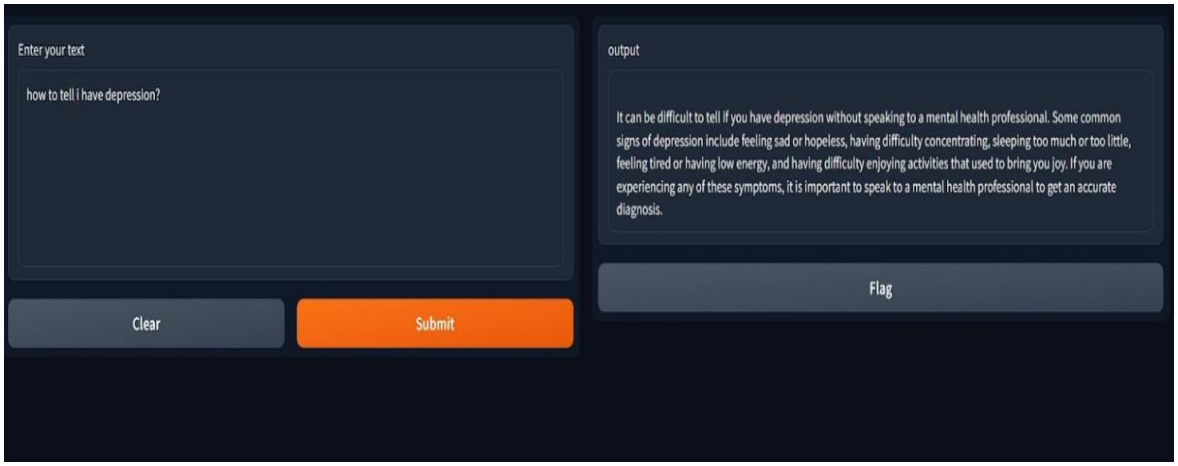

Fig. 5.6: Result for Question 3

Now we gave the role of an assistant to the Chatbot that helps the customers with their questions.

```
messages = [
    {
        "role":
        "system",
        "content":
        "You are an assistant that helps customers with their questions."
    },
]
```

Fig. 5.7: Assignment of Role

```
Running on local URL:  http://127.0.0.1:7860
Running on public URL: https://a686d84e044ac80498.gradio.live

This share link expires in 72 hours. For free permanent hosting an
d GPU upgrades, run `gradio deploy` from Terminal to deploy to Spa
ces (https://huggingface.co/spaces)
```

Fig. 5.8: Generation of Local and Public URL

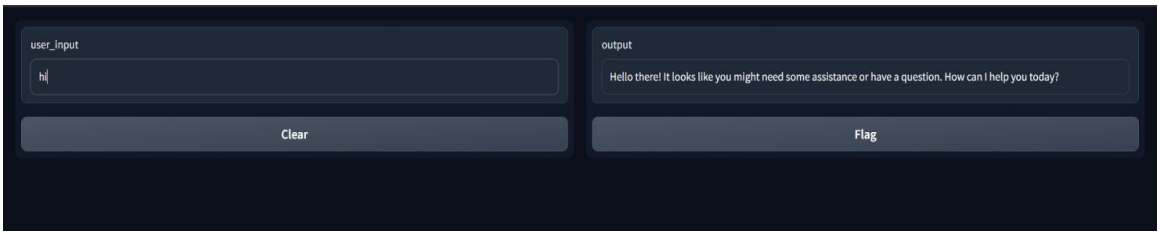

Fig. 5.9: Result of Question

However, it could only answer the questions that were specific to the role assigned to the Chatbot. We made it answer the question that was from the previous role i.e. role of a psychologist. So as a result, it gave out error.

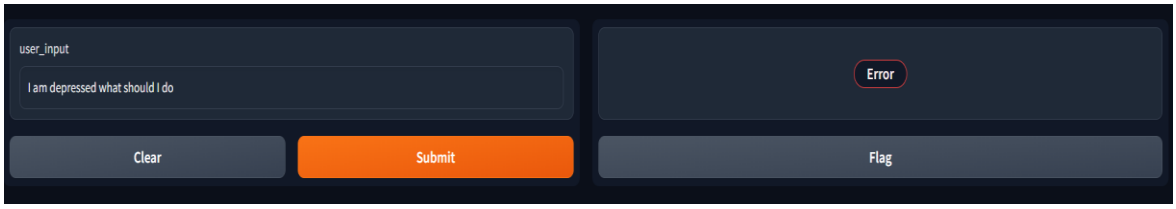

Fig. 5.10: Error as Output When Question is Out of Domain

The new chatbot created by us also generates a local and public URL but takes a significant amount of time as compared to before to generate these two.

Fig. 5.11: Generation of Local and Network URL

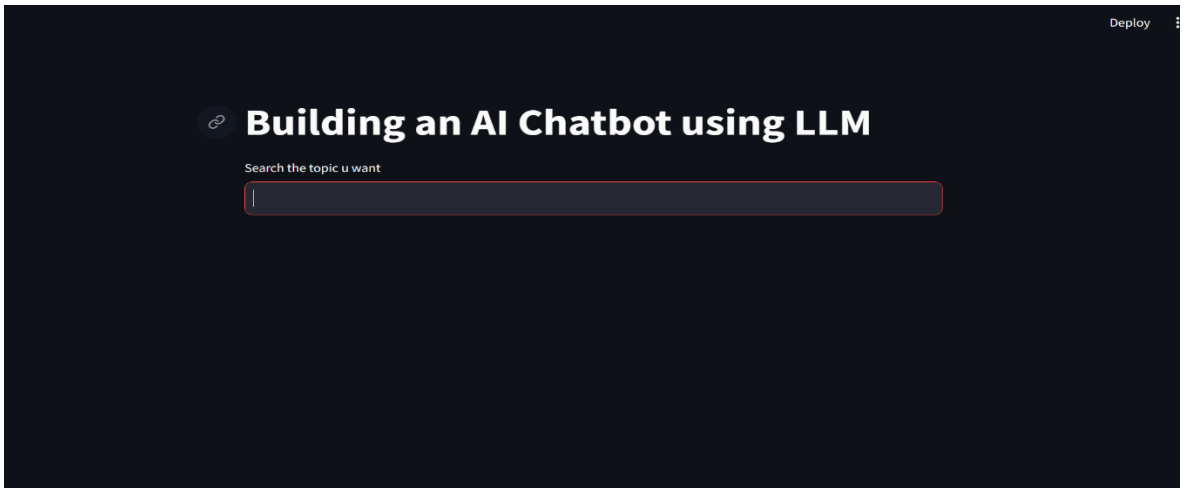

Fig. 5.12: Interface After Running of URL

This time there is no limitation like answering the questions just related to the domain specified. Instead, this time we don't need to specify any domain to the chatbot. It can answer questions put forward to it from any domain and also gives some additional information sometimes like in the case mentioned below.

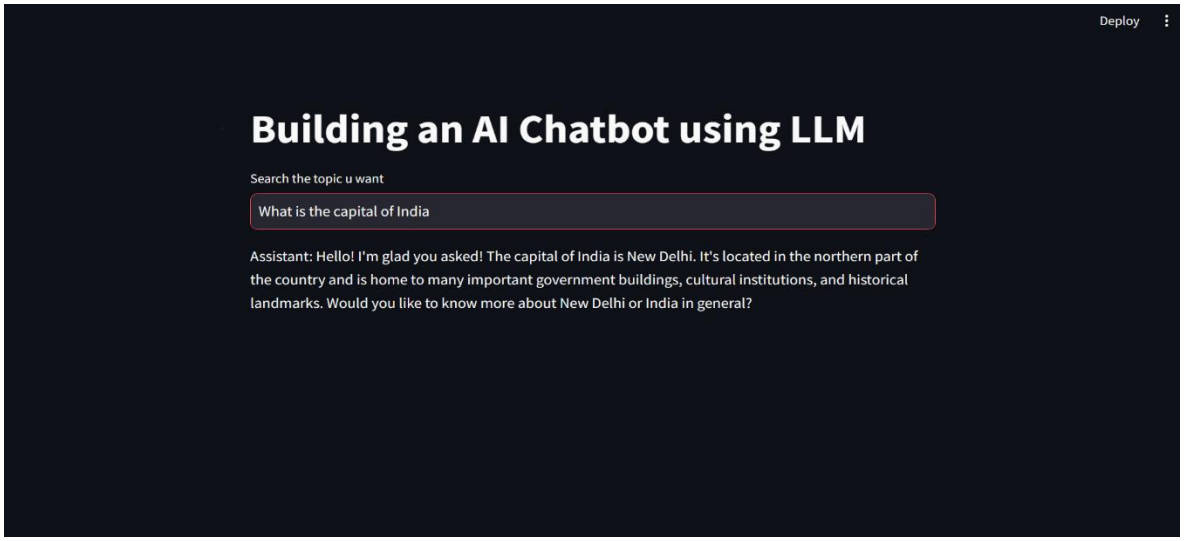

Fig. 5.13: Result of Query Put by User with Extra Information

CHAPTER 06: CONCLUSIONS AND FUTURE SCOPE

6.1 CONCLUSION

In conclusion, a Chatbot designed with Open AI GPT 3.5 and assign distinct roles—let's used the example of a psychologist—is a useful tool for providing mental health support, as demonstrated by the pictures. Thanks to its ability to respond to user inquirie in a manner same to that of a human, understand the context in which language is used, and offer customized solutions, the Chatbot performs noticeably better than traditional Chatbots. based on strong professional experience and understanding.

A substantial improvement in NLP overall, based on Open AI GPT 3.5. By leveraging Open AI GPT 3.5's more advanced machine learning algorithms, developers may create Chatbots that are more accurate, relevant, and personalised than previous iteration. The communities such as Hugging Face provide the students and learners with the desired help needed by them and that too for free. This has important implications for a range of applications, including conversational agents, teaching, customer service, and mental health support. As natural language processing and comprehension advance, it is anticipated that Chatbots built with the help of Open AI GPT will become more commonplace, serving as helpful resources for anyone in need of assistance and knowledge.

Very soon the time will come when people would build and use the LLM powered Chatbots created and customised by them without having to stay dependent on Chatbots provided by some other platforms.

6.2 FUTURE SCOPE

Some more work could be done in the following aspects of the project in the near future:

- Integrating the Chatbot with multiple user interfaces rather than using some predefined interface and providing the liberty to the user to choose his own interface out of the many provided ones.

- Adding a feature in the chatbot that would enable it to take in the query in the voice form from the microscope and deliver the output in the voice from through the speaker and move a step ahead from the simple text to text conversations.
- Making the Chatbot take in longer queries as an input and answer to the questions while maintaining the context.
- Making the Chatbot hold conversation in multiple turns and hold a toe and forth chats with the Chatbot.

REFERENCES

[1] Ashish Vaswani. (October 2018). “Attention Is All You Need” *Information Processing* [Online]. vol. 59, no. 11, pp. 1040–1042.

[2] J. Yang. (April 2023). “Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond”. Available: https://doi.org/10.48550/arxiv.2304.13712.

[3] Google Cloud. Large Language Models (LLMs) with Google AI [Online]. Available: https://cloud.google.com/ai/llms.

[4] Humza Naveed. (July 2023). “A Comprehensive Overview of Large Language Models” *Computation and Language* [Online]. Available: https://arxiv.org/abs/2307.06435.

[5] Slator. (2023). “What is the Difference Between NLP, NLU and NLG?”. [Online]. Available: https://slator.com/resources/what-is-the-difference-between-nlp-nlu-nlg/.

[6] Jing Wei, Sungdong Kim, Hyunhoon Jung, Young-Ho Kim. (2023). “Leveraging Large Language Models to Power Chatbots for Collecting User Self-Reported Data” *NAVER AI Labs* [Online]. arXiv:2301.0584v1 [cs.HC].

[7] Jesse G. Meyer, Ryan J. Urbanowicz, Patrick C. N. Martin, Karen O’Connor, Ruowang Li, Pei-Chen Peng, Tiffani J. Bright, Nicholas Tatonetti, Kyoung Jae Won, Graciela Gonzalez-Hernandez & Jason H. Moore. (2023). “ChatGPT and large language models in academia: opportunities and challenges”. *BioData Mining* [Online] 16:20.

[8] Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, Stephan Krusche, Gitta Kutyniok, Tilman Michaeli, Claudia Nerdel, Juergen Pfeffer, Oleksandra Poquet, Michael Sailer & Albrecht Schmidt. (January 2023). “ChatGPT for good? On opportunities and challenges of large language models for education”. [Online]. DOI: 10.1016/j.lindif.2023.102274.

[9] Jae-ho Jeon, Seongyong Lee. (May 2023) “Large language models in education: A focus on the complementary relationship between human teachers and ChatGPT”. *Education and Information Technologies* [Online]. DOI:10.1007/s10639-023-11834-1.

[10] Gelei Deng, Yi Liu, Yuekang Li, Kailong Wang, Ying Zhang, Zefeng Li, Haoyu Wang, Tianwei Zhang & Yang Liu. (July 2023). “JAILBREAKER: Automated Jailbreak Across Multiple Large Language Model Chatbots”. *Cryptography and Security* [Online]. arXiv:2307.0871v1 [cs.CR] 2023.

[11] Eunkyung Jo, Daniel A. Epstein, Hyunhoon Jung & Young-Ho Kim. (April 2023). “Understanding the Benefits and challenges of Deploying Conversational AI Leveraging Large Language Models for Public Health Intervention”. *Human Factors in Computing Systems* [Online]. Available: https://doi.org/10.1145/3544548.3581503.

[12] Luigi De Angelis Francesco Baglivo, Guglielmo Arzilli, Gaetano Pierpaolo Privitera, Paolo Ferragina, Alberto Eugenio Tozzi & Caterina Rizzo. (April 2023). “ChatGPT and the rise of large language models: The new AI-driven infodemic threat in public health”. *Digital Public Health* [Online]. Available: https://doi.org/10.3389/fpubh.2023.1166120.

[13] Shan Chen, Benjamin H. Kann & Michael B. Foote. (2023). “AI Chatbots for cancer treatment”. *JAMA Oncol* [Online]. DOI:10.1001/jamaoncol.2023.2954.

[14] Dmitry I. Mikhailov. (May 2023). “Optimizing National Security Strategies through LLM-Driven AI”, *Artificial Intelligence* [Online]. DOI: 10.14293/PR2199. 000136.v1.

[15] Desiree Bill & Theodor Eriksson. (2023). “Therapy Chatbot Application”. *Computer and Information Sciences* [Online]. arXiv:2401.04592v2 [cs.CL] 02.

[16] Ben Niu. (2021). “Generative Conversational AI And Academic Integrity”. *Artificial Intelligence* [Online]. Available: http://dx.doi.org/10.2139/ssrn.4548263.

[17] Volker Hartmann, Gibbeum Lee, Jongho Park, Dimitris Papailiopoulos & Kangwook Lee. (2022). “Chatbot Modules for Long Open-domain Conversation”. *Computation and Language* [Online]. Available: https://doi.org/10.48550/arXiv.2305.04533.

[18] Dale, R. (2020). “GPT-3: What's IT good for?”. *Natural language engineering* [Online]. DOI: https://doi.org/10.1017/S135132492000060.

[19] T. Bocklisch, J. Faulkner, N. Pawlowski, & A. Nichol. (December 2017). “Rasa: Open-source language understanding and dialogue management”. *Computation and Language* [Online]. DOI:10.14429/djlit.40.06.15611.

[20] Floridi, L. and Chiriatti, M. (2020). “GPT-3: Its nature, scope, limits, and consequences - minds and machines”. *Computation and Language* [Online]. Available: https://link.springer.com/article/10.1007/s11023-020-09548-1.

[21] Min Zhang and Juntao Li. (2021) “A commentary of GPT-3 in MIT Technology Review”. *Fundamental Research* [Online]. DOI: 10.1016/j.fmre.2021.11.011.

[22] OpenAI. (2023). “What is ChatGPT? | OpenAI Help Center”. [Online]. Available: https://help.openai.com/en/articles/6783457-what-is-chatgpt.

[23] G. Team. (2023). “Gradio Interface Docs”. [Online]. www.gradio.app. Available: https://www.gradio.app/docs/interface.

[24] Hugging Face. (2023). “meta-llama/Llama-2-7b-hf · Hugging Face”. [Online]. huggingface.co. Available: https://huggingface.co/meta-llama/Llama-2-7b-hf.

[25] LangChain. (2024). “Quickstart | LangChain”. [Online]. python.langchain.com. Available: https://python.langchain.com/v0.1/docs/get_started/quickstart/.

[26] Streamlit. (May 2024). “Streamlit: A faster way to build and share data apps”. [Online]. Available: https://streamlit.io/generative-ai.

APPENDIX

AI Chatbot

ORIGINALITY REPORT

5%	3%	3%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	arxiv.org Internet Source	1%
2	www.coursehero.com Internet Source	<1%
3	Keisuke Takahashi, Lauren Takahashi. "Materials Informatics and Catalysts Informatics", Springer Science and Business Media LLC, 2024 Publication	<1%
4	www.ncbi.nlm.nih.gov Internet Source	<1%
5	escholarship.org Internet Source	<1%
6	Enkelejda Kasneci, Kathrin Sessler, Stefan Küchemann, Maria Bannert et al. "ChatGPT for good? On opportunities and challenges of large language models for education", Learning and Individual Differences, 2023 Publication	<1%