GMR Institute of Technology
An Autonomous Institute Affiliated to JNTUK, Kakinada
GMRIT
Training Tomorrow's Engineers Today

Chatbot for Naïve Programmers using NLP and Deep Learning

A project report submitted in partial fulfilment of the requirement

for the award of degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted by

G. Jothsna	**(20341A0570)**
K. Nikhil Kumar	**(20341A0595)**
N. Lokesh	**(20341A05C8)**
M. Divya	**(20341A05B1)**
L. Amrutha	**(20341A05B0)**

Under the esteemed guidance of

Dr. M. Satish
Associate Professor, Dept. of CSE

GMR Institute of Technology

An Autonomous Institute Affiliated to JNTU-GV, Vizianagaram
(Accredited by NBA, NAAC with 'A' Grade & ISO 9001:2023 Certified Institution)

GMR Nagar, Rajam – 532127,
Andhra Pradesh, India

April 2023

GMR Institute of Technology
An Autonomous Institute Affiliated to JNTUK, Kakinada

GMRIT
Training Tomorrow's Engineers Today

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the thesis entitled **Chatbot for Naïve Programmers using NLP and Deep Learning** submitted by **G. Jothsna(20341A0570), K. Nikhil Kumar(20341A0595), N. Lokesh(20341A05C8), M. Divya (20341A05B1), L. Amrutha (20341A05B0)** has been carried out in partial fulfillment of the requirement for the award of degree of **Bachelor of Technology** in **Computer Science and Engineering** of **GMRIT, Rajam** affiliated to **JNTU-GV, Vizianagaram** is a record of bonafide work carried out by them under my guidance & supervision. The results embodied in this report have not been submitted to any other University or Institute for the awardof any degree.

Signature of Supervisor	**Signature of HOD**
Dr. M. Satish	**Dr. A. V. Ramana**
Associate Professor	Professor & Head
Department of CSE	Department of CSE
GMRIT, Rajam.	GMRIT, Rajam.

The report is submitted for the viva-voce examination held on ……………….

Signature of Internal Examiner   Signature of External Examiner

ACKNOWLEDGEMENT

It gives us an immense pleasure to express deep sense of gratitude to my guide **Dr. M. Satish,** Associate Professor, Department of Computer Science and Engineering for his whole hearted and invaluable guidance throughout the project work. Without his sustained and sincere effort, this project work would not have taken this shape. He encouraged and helped us to overcome various difficulties that we have faced at various stages of our project work.

We would like to sincerely thank our Head of the department **Dr. A. V. Ramana**, for providing all the necessary facilities that led to the successful completion of our project work.

We would like to take this opportunity to thank our beloved Principal **Dr.C.L.V.R.S.V. Prasad**, for providing all the necessary facilities and a great support to us in completing the project work.

We would like to thank all the faculty members and the non-teaching staff of the Department of Electronics and Communication Engineering for their direct or indirect support for helping us in completion of this project work.

Finally, we would like to thank all of our friends and family members for their continuous help and encouragement.

G. Jothsna (20341A0570)

K. Nikhil Kumar(20341A0595)

N. Lokesh(20341A05C8)

M. Divya(20341A05B1)

L. Amrutha(20341A05B0)

ABSTRACT

The main aim of project is to develop a text - based conversational agent - chat bots to guide and support novice programmers. The chat bot offer "human-like" conversational capabilities in natural language or engaging experiences with a familiar turn-based messaging interface. It deals with chat bots answering to the errors occurred when execution of a program gets terminated due to errors (syntax or runtime) in code. It clarifies doubts, sustain conversation, handles dialog failures and end conversations gracefully. Natural Language Processing(NLP) Tools are applied to understand natural language queries and provide accurate and relevant responses and Deep learning algorithms are used to train model. The project will involve developing a user-friendly interface for the chatbot, which will be accessible through various platforms such as web and mobile applications. The chatbot will also be integrated with various programming environments and tools to provide a seamless experience for beginner programmers. User enters the query and it is used by model to generate output. This result will be displayed through the chat bot UI. The model generates the most picked answer for the respective user query. Data required for modelling is taken from Kaggle and other data sources.

Keywords: Chat bot, Natural Language Processing, word embeddings, Deep Learning, Text-based

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS & ABBREVIATIONS

AI	:	Artificial Intelligence
DL	:	Deep Learning
LSTM	:	Long Short-Term Memory search
NLP	:	Natural language processing
NLTK	:	Natural Language Processing ToolKit
BERT	:	Bidirectional Encoder Representation from transformers
GORU	:	Gated Orthogonal recurrent Unit
GUI	:	Graphical User Interface
KG	:	Knowledge graph
QA	:	Question Answering
RNN	:	Recurrent Neural Network
TF-IDF	:	Term Frequency Inverse document frequency

1. INTRODUCTION

1.1 Introductory paragraph:

In the last few decades, chat-bots have become increasingly sophisticated and have taken on a greater role in our daily lives. Developed in the 1960s to interact with computers through natural language, chat-bots ,it has come a long way since then and is now essential in many industries. One of the biggest advances in chat bot technology is the ability to use artificial intelligence (AI) to enable more natural, human-like conversations. This is made possible by using machine learning algorithms. This allows chat bots to improve their responses and interactions over time based on data and user feedback.

Programming is a complex skill that takes years of practice and experience to master. For novice coders, this journey is a daunting one, with a myriad of challenges and obstacles to overcome. Fortunately, recent advances in Natural Language Processing (NLP) and Deep Her Learning (DL) have opened up new opportunities to provide guidance and support to novice coders. This project aims to use NLP and DL to develop a chatbot that provides novice programmers with a user-friendly and accessible platform to learn and improve their programming skills.

1.2 Major challenges in the current literature in line with proposed work:

One of the biggest challenges that new programmers face is simply figuring out where to start. With so many resources and so much information available online, it can be overwhelming to try to figure out which ones are the most helpful and relevant. That is where our chatbot comes in. By asking simple questions and providing concise, easy-to-follow instructions, the chatbot can guide new programmers through various tasks and concepts, helping them to build their skills and confidence as they go. Whether it’s troubleshooting a bug, learning a new programming language, or simply getting started with a new project, the chatbot is there to help.

Another challenge regarding chatbots is the ability to remember more patterns and usage of programmers to determine his work and recommend actions. The art of remembering the historical patterns to generate the answers using the past actions and work may help the developer to rely upon the chatbot. This is a growing issue with respect to chatbot and NLP. The generated result for the given query may not help the user all the time. As it is a machine generated answer and programmer cannot make worthy of it unlike Object relational mapping used by most of the web applications like stackoverflow.

1.3 Solutions to these challenges

To achieve this task , we used natural language processing ,which is one of the prominent sub-field of Artificial Intelligence. This can be done using techniques such as part-of-speech tagging, named entity recognition, and dependency parsing, which can help to extract relevant information from the question and identify the key concepts and relationships that it contains. The user gives the error in the form of a query. The model fetches the data requested. It maps the key elements with the knowledge base and responds with the possible answers.

This Chatbot is used during troubleshooting a bug or learning a new language or normally.The chatbot will be designed to understand the queries of novice programmers and provide them with helpful responses in natural language. Using advanced NLP techniques, the chatbot will be able to interpret the user's queries accurately and provide relevant and actionable advice. Additionally, we will leverage the power of DL algorithms to enable the chatbot to learn from user interactions, improving its accuracy and effectiveness over time. We use deep learning model like sequential, LSTM which is a variant transformer model. It involves fine – tuning of large datasets to specific task datasets. The chatbot will be developed using advanced NLP techniques to ensure that it can understand natural language queries and provide accurate and relevant responses. It will be trained on a large data set of programming-related queries and responses, and will use machine learning algorithms to improve its accuracy and effectiveness over time.

Today, practically every business has a chatbot to interact with clients and support them by answering their questions. Although chatbots will be practically ubiquitous, this is not guarantee that they will all work properly. Here, creating a chatbot is not the difficult part; rather, creating one that works well is. For each sort of task, the appropriate artificial intelligence solution can be roughly divided into two categories: “Data Complexity” or “Work Complexity”. Efficiency, Expert, Effectiveness, and Innovation are the four analytics models that can be used to further segment these two groups. This project involves developing an easy-to-use interface for chatbots that can be accessed from various platforms such as web and mobile applications. The chatbot integrates with various programming environments and tools to provide a seamless experience for novice coders. Through this project, programming is more accessible and enjoyable for novice programmers, and improve their skills in this important field. It imparts confidence in beginner learners to excel in coding.

There are many different varieties of chatbots. A few them fall within the following categories:

1. **Text-based chatbot**: A text-based chatbot responds to user inquiries using a text-based user interface.
2. **Voice-based chatbot**: A voice-based chatbot, also known as a speech-based chatbot, responds to user inquiries using a human voice interface.

The design of chatbots primarily follows two methods, which are as follows:

1. With a rule-based method, a bot responds to queries in accordance with some pre-trained rules. The rules specified might range in complexity from very simple to highly complicated. Simple inquiries are handled by the bots, but complicated ones are not.
2. Self-learning bots, which employ a few machines learning-based techniques, are unquestionably more effective than rule-based bots. These bots can also be divided into two categories: retrieval-based bots and generative bots.

The Architecture of chatbots:

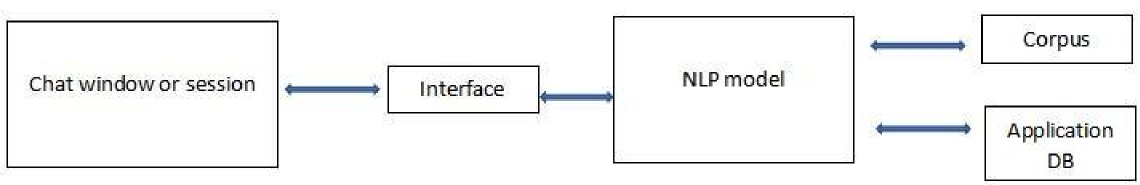

Figure 1.1 : Architecture of Chatbots

Natural language processing (NLP) has helped improve historical chatbots by enabling them to accurately and effectively understand and respond to natural language queries. Historically, chatbots were limited to simple, keyword-based responses that lacked context or nuance, resulting in a frustrating user experience. However, with the development of NLP techniques such as sentiment analysis, named entity recognition, and contextual awareness, chatbots can now understand the intent behind user requests and provide relevant and helpful responses. Hence, the programmer can easily depend on the application to get their queries clarified.

2. LITERATURE SURVEY

Hao, T., Li, X., He, Y., Wang, F. L., & Qu, Y. (2022). Recent progress in leveraging deep learning methods for question answering. Neural Computing and Applications, 1-19. Recent works regarding different tasks in QA area, including question classification, answer extraction, question–answer matching, knowledge base question answering, and question generation are discussed. Models used for question classification are BERT model , kernel ,neural based networks and other dependency-based CNN , RNNs with GORU were a novel RNNs-based model that is used for answer extraction. Experiments were conducted upon SNIPS, Stack Overflow, ECDT, and Question datasets, containing English and Chinese questions, covering different intent categories and various topics

Nassiri, K., & Akhloufi, M. (2022). Transformer models used for text-based question answering systems. Applied Intelligence, 1-34 To build an automated QA system that has access to FAQs and company documentation could streamline the customer experience
The versatility of deep learning models has become quite increasingly popular for NLP due to their capacity to effectively generalize over a range of contexts and languages . The transformers that are considered the cornerstone of most modern models of question answering are LSTM, RNN and other transformer models. For encoding based ,models like BERT, Albert, XLNet ,GlaM,Electra , tinybert are used , for for decoding models used are GPT , GPT-2 , GPT -3.The attention mechanism is implemented to overcome the problem of loss of information when reached to decoder state.

Abdel-Nabi, H., Awajan, A., & Ali, M. Z. (2022). Deep learning-based question answering: a survey. Knowledge and Information Systems, 1-87 Understanding the implemention of different types of question answering system i.e context , reasoning ,answer format,domain etc Deep Learing models such as BERT, ELMo, LUKE, ELECTRA, BIGBIRD, ANNA. The Convolutional and attention-based models are used for answer selection ,Recurrent and attention-based models are used to learn to address the facts ,graph based models are used for the conversational machine reading comprehension task

Do, P., & Phan, T. H. (2022). Developing a BERT based triple classification model using knowledge graph embedding for question answering system. Applied Intelligence, 52(1), 636-651. Using the knowledge graph (KG) for enhancing the question answering system is a promising study in recent years and plays an important role in natural language Q&A systems. The KG-Bert model is used to classify triples for knowledge graph completion . The BERT model achieved the top accuracy for text classification and has highest performance when added with softmax added layers over CNN .

Abbasiantaeb, Z., & Momtazi, S. (2022). Entity-aware answer sentence selection for question answering with transformer-based language models. Journal of Intelligent Information Systems, 59(3), 755-777. Classification of QA models into several categories like factoid , non factoid questions or text-based , knowledge based , retrieval-based and reading comprehension BiLSTM , Conv-based LSTM , QA-LSTM are used for vector representation of each sentence and matching score between the question and the answer sentences . The study proposed two different architectures namely Ent-match and Ent-add which we use the matching signal between question type and answer sentence's Nes as an auxiliary feature

Lilian, J. F., Sundarakantham, K., & Shalinie, S. M. (2021). Anti-negation method for handling negation words in question answering system. The Journal of Supercomputing, 77(5), 4244-4266. The study proposed a novel anti-negation method algorithm to handle the negation words in the content and compared the state-of-the-art embedding models. Research on Negation in the area of Sentiment Analysis, Clinical Text and at present into the Question Answer system. It analyzes whether the given sentence is of positive polarity or negative polarity. Models used for are bi-directional long short-term memory model for handling the question answer task to retrieve the exact phrase for the factoid-type question

Wang, J., Jatowt, A., Färber, M., & Yoshikawa, M. (2021). Improving question answering for event-focused questions in temporal collections of news articles. Information Retrieval

Journal, 24(1), 29-54. For efficiency , the model categorizes questions into two crude types: (1) explicitly time-scoped questions: ones containing explicit temporal expressions and implicitly time-scoped questions: ones without any explicit temporal expression in their statements. To improve the capability of QANA to better utilize the temporal information, and also introduce an additional method to answer the frst type of event-focused questions – the explicitly time-scoped questions, as well as providing more detailed experimentation. Because of the omission of the temporal information, the systems process the questions and the news articles in essentially the sameway as in the case of synchronic document collection.

Gomes, J., de Mello, R. C., Ströele, V., & de Souza, J. F. (2022). A study of approaches to answering complex questions over knowledge bases. Knowledge and Information Systems, 1-33. The paper presents the foundation and compares work with other reviews in the KBQA field. It describes the systematic mapping protocol that has been followed.
The PICOC method helps to identify relevant keywords from the objectives associated with each of its entries. The computational cost is one of the main problems in the complex question approaches. The QA systems need to handle too many triples and hops to answer some complex questions.

Alzubi, J. A., Jain, R., Singh, A., Parwekar, P., & Gupta, M. (2021). COBERT: COVID-19 question answering system using BERT. Arabian journal for science and engineering, 1-11. The practices from the information retrieval with BERT to cre_x0002_ate a framework focusing on an end-to-end closed domain question answering system. The CDQA based COVID-19 Search engine is implemented based on a retriever-reader dual algorithmic approach. In the COBERT Retriever Pipeline schema ,the retriever schema chooses a couple of documents from a database to answer those questions that are set to a system. The COBERT system is working the same as a retriever of DrQA which creates TF-IDF features based on uni grams and bi-grams.

Mondal, S., Rahman, M. M., Roy, C. K., & Schneider, K. (2022). The reproducibility of programming-related issues in Stack Overflow questions. Empirical Software Engineering, 27(3), 1-52. Report on an exploratory study on the reproducibility of programming issues discussed in 400 Java and 400 Python questions from Stack Overflow. Dependency on external libraries is another major challenge towards issue reproducibility from the submitted code. In many cases, the model do not find any hints that point to the appropriate libraries in error code. Programmers might be seeking general help or asking for source code that is more efficient than the submitted code segment.

Zaib, M., Zhang, W. E., Sheng, Q. Z., Mahmood, A., & Zhang, Y. (2022). Conversational question answering: A survey. Knowledge and Information Systems, 1-45

An effort to present a comprehensive review of the state-of-the-art research trends of CQA primarily based on reviewed papers over the recent years. The question type consists of characteristics such as comparisons, temporal reasoning, and anaphora, to make it more closely related to real-world challenges. Methods only focus on answering multi-hop questions via Graph Neural Network (GNN) , reinforcement learning , and memory network. Entity linking and relation extraction are two main tasks that are used in KBQA systems along with having a robust EL and RE can improve the performance without changing the whole algorithm and reduce errors.

Cao, X., Zhao, Y., & Shen, B. (2022). Improving and evaluating complex question answering over knowledge bases by constructing strongly supervised data. Neural Computing and Applications, 1-21. This paper provides an in-depth analysis of the model's ability to answer different types of questions, contributing a comprehensive evaluation of the performance of CQA models. A lack of supervised labels during each reasoning step , Uninterpretable reasoning are the biggest challenges for the proposed model. GraftNet is a conventional model for the KBQA task that combines a knowledge base with additional text to build hierarchical graphs and perform multi-hop reasoning and NSM is widely used.

Arora, R., Singh, P., Goyal, H., Singhal, S., & Vijayvargiya, S. (2021, March). Comparative question answering system based on natural language processing and machine learning. In 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS) (pp. 373-378). IEEE. This research is to build the best accuracy model using Glove and Infer Sent to use vector of various dimensionality to represent it numerically so that machine can interpret it.This approach solves the problem that the model only grasps the topological structure of the knowledge graph while ignoring the textual information.In the model RNN bidirectional LSTM model, loss is quite high Multinomial Logistic Regression and Boost model

Tommy, L., Kirana, C., & Riska, L. (2020, October). The Combination of Natural Language Processing and Entity Extraction for Academic Chatbot. In 2020 8th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-6). IEEE. The created chatbot makes the academic information be accessed easily, quickly, and efficiently by students without having to zoom in and zoom out. The proposed chatbot can improve higher-education institutions current services, to reduce labor costs, and to create new innovative services. The study elaborated knowledge-based chatbot called as Dinus Intelligent Assistance (DINA) which acts as a conversation agent that can play a role as student candidate service.

Deriu, J., Rodrigo, A., Otegi, A., Echegoyen, G., Rosset, S., Agirre, E., & Cieliebak, M. (2021). Survey on evaluation methods for dialogue systems. Artificial Intelligence Review, 54(1), 755-810. The paper mainly focus on evaluation of question answering systems which are time and cost intensive. Evaluating a dialogue system can prove to be problematic because the definition of what constitutes a high-quality dialogue is not always clear and often depends on the application. In this paper the problem of evaluation is evolving in tandem to the progress of the dialogue system technology itself.

Kim, B., Seo, J., & Koo, M. W. (2021). Randomly wired network based on RoBERTa and dialog history attention for response selection. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29, 2437-2442. The experimental demonstration that feed-forward networks , replaced by randomly wired neural networks and calculating the correlation between dialog history and the last utterance of the context, reflecting this correlation as an attention network that yielded an answer prediction. BERT model (RoBERTa) , Randomly wired neural network , DHA are used for pre-training the tasks There was a problem, however, in that Recall@5 and recall@10 which results in low scores. This problem was related to the number of “NONE” entries to the prediction of the no answer problem.

Sheth, V., & Damevski, K. (2022). Grouping related stack overflow comments for software

developer recommendation Automated Software Engineering, 29(2), 1-21. To assess the strength of the relationship between selected pairs of Stack Overflow comments posted on a single question or answer. The main aim of the paper is to assess clustering technique that forms groups from the given set of comments on a Stack overflow post , evaluation of the effectiveness of the comment relatedness and clustering using a manually annotated corpus. Random Forest classifer, K-Nearest Neighbors, Artifcial Neural Network, Support Vector Machine, XGBoost and LightGBM are some of the machine learning models used to check the performance.

Rai, S., Belwal, R. C., & Gupta, A. (2022). Accurate module name prediction using similarity based and sequence generation models. Journal of Ambient Intelligence and Humanized Computing, 1-13. Emphasize the module name and propose the module name prediction approach. The sequence generation models can predict the module name tokens in a sequence, while similarity-based models only suggest pre-stored module names.The study shows a proposed model of an attention based neural network that convolves over the input tokens and generates method name tokens as a sequence. It presented a multi-lingual general path-based representation to learn from programs and suggest identifier names.The models are implemented using TF-IDF , word2vec ,GloVe ,CSGM , Latent Dirichlet allocation for topic modelling, etc.

Kuhail, M. A., Alturki, N., Alramlawi, S., & Alhejori, K. (2022). Interacting with educational chatbots: A systematic review. Education and Information Technologies, 1-46. Chatbots on a web platform to teach computer science, language, general education, and a few other fields such as engineering and mathematics.The study classifies the chatbots in terms of application as peer agents , teachable agents , motivational agents with models being used for implementation.Insufficient or Inadequate Dataset Training , user centered design ,lack of feedback and deviations are the main limitations of the dialog system.

Bird, J. J., Ekárt, A., & Faria, D. R. (2021). Chatbot Interaction with Artificial Intelligence: human data augmentation with T5 and language transformer ensemble for text classification. Journal of Ambient Intelligence and HumanizedComputing, 1-16. An intelligent chatbot system based on Artificial Intelligence Markup Language (AIML) which can be used as an e-commerce assistant. This Chatbot is applied to the Telegram application.The

State-of-the-art transformer-based text calculation algorithms (BERT, DistilBERT, RoBERTa, DistilRoBERTa, XLM, XLM-RoBERTa, and XLNet) are benchmarked for both sets after fine-tuning on the training data for two epochs. Logistic Regression and Random Forest methods of ensembling Transformer for predictions.

Table 2.1 : Comparison Table

Reference	Reference (i.e. author names with reference number)	year	Objectives	Limitations	Advantages	Performance metrics	Gaps
1	Hao, T., Li, X., He, Y., Wang, F. L., & Qu, Y. (2022). Recent progress in leveraging deep learning methods for question answering. Neural Computing and Applications, 1-19.	16th Jan ,2022	This survey aims to contribute to summarizing recent research progress and future directions for Deep learning and learn to answer questions	The model have the shortcoming of CNNs on insufficiently capturing syntactic information inside a sentence	Building a deep learning model requires a long training phrase and consumes a lot of computation resources	precision, recall, F1, accuracy, ER, EM, MAP, and MRR	
2	Nassiri, K., & Akhloufi, M. (2022). Transformer models used for text-based question answering systems.	29th July , 2022	research trends in text QA datasets by highlighting and classifying the alculation of QA	All deep learning models suffers from the problem of the gap between	Introduced an open source library developed to simplify the implementation of	Mean reciprocal rank , accuracy ,exact match, word mover's distance	

	Applied Intelligence, 1-34		systems them according to different criteria	QA datasets and real-world scenarios	Transformer models		
3	Abdel-Nabi, H., Awajan, A., & Ali, M. Z. (2022). Deep learning-based question answering: a survey. Knowledge and Information Systems, 1-87.	23, Oct , 2022	This paper provides a new taxonomy of question answering systems from multiple criteria based on the used context, the answer, the domain, and the reasoning that characterize these systems	the complex and memory based deep learning models introduce high computational costs that may be unsuitable for practical application	The efficiency of the transformers lies in their parallelization ability to compute self-attention for the words in the text, which makes their hardware optimization more manageable	precision, recall, F1, accuracy, ER, EM, MAP, and MRR	enhancement can be done at many levels-powerful representations of the words, i.e., word embeddings and LMs; enhancing and improving the existing deep learning models by addressing the weakness and limitations of the previous models
4	Do, P., & Phan, T. H. (2022). Developing a BERT based triple	2021	Generating the triples from all meta paths of HIN by scanning Network Schema and	The proposed model have not regarded heteroge	The BERT model achieved the top accuracy for text classificati	Recall , precision	

	classification model using knowledge graph embedding for question answering system. Applied Intelligence, 52(1), 636-651.		using Motif Finding of Apache Spark Graph Frames on large HIN of KG	neity and imbalance of alculat	on and has highest performance when added with SoftMax		
5	Abbasiantaeb, Z., & Momtazi, S. (2022). Entity-aware answer sentence selection for question answering with transformer-based language models. Journal of Intelligent Information Systems, 59(3), 755-777.	14, alc,2022	To investigate the impact of Ent-match and Ent-add architectures on the pre-trained transformer-based language models	The proposed model has a pipeline architecture and is not trained in an end-to-end manner which makes it vulnerable to error propagation	Creating a question aware representation for the answer sentence which gives more weight to the informative and most important tokens of the answer sentences	Mean Average Precision, Mean Reciprocal Rank	
6	Lilian, J. F., Sundarakantham, K., & Shalinie, S. M. (2021). Anti-negation method for handling negation words in	24 sept 2021	Proposed a novel anti-negation method algorithm to handle the negation words in the content and compared the state-of-the-	negate words which occur in prefixes or suffixes do not be considered by	Methods are used detect the negation words that are present in the text based on the entity other than which	Precision, recall, accuracy	enhance the visibility of the negation terms in the above-mentioned applications with more

	question answering system. The Journal of Supercomputing, 77(5), 4244-4266.		art embedding models	the model	improve the efficiency for retrieving them		accuracy
7	Wang, J., Jatowt, A., Färber, M., & Yoshikawa, M. (2021). Improving question answering for event-focused questions in temporal collections of news articles. Information Retrieval Journal, 24(1), 29-54.	2 jan 2021	Providing alculati models for answering questions against temporal document collections by exploiting diverse temporal alculation i c of both questions and document	Due to their large sizes, complexities and different context, it is difficult for users to use news archives effectively	Crucial for event-oriented questions because of temporal information gathering	F1 – score and EM score	extend the test set and to conduct more detailed evaluation on the longer temporal alculatio of the news articles
8	Gomes, J., de Mello, R. C., Ströele, V., & de Souza, J. F. (2022). A study of approaches to answering complex questions over knowledge bases. Knowledge and Information Systems, 1-	July, 2022	the use of a systematic method to provide an overview of the state of the art in complex knowledge base question answering	errors can be inserted in the protocol definition and the search string may not contain all the relevant keywords, causing the loss	Pre-trained models are faster in training new solutions	Accuracy, precision, processing time	Advances in C-KBQA and Deep learning can create architectures that demand less memory and have a lower training cost

	33.			of some valuable studies			
9	Alzubi, J. A., Jain, R., Singh, A., Parwekar, P., & Gupta, M. (2021). COBERT: COVID-19 question answering system using BERT. Arabian journal for science and engineering, 1-11.	2021	practices from the information retrieval with BERT to cre_x0002_at e a framework focusing on an end-to-end closed domainquesti on answering system	errors can be inserted in the protocol definitio n and the search string may not contain all the relevant keywor ds, causing the loss of some valuable studies	fine-tunes pre-trained versions of BERT with SQaUAD ade_x0002 _quately accomplish ing high scores in recognizing alcul	precision , F1-score	
10	Mondal, S., Rahman, M. M., Roy, C. K., & Schneider, K. (2022). The reproducibilit y of programming -related issues in Stack Overflow questions. Empirical Software Engineering,	2022	Report on an exploratory study on the reproducibilit y of programming issues discussed in 400 Java and 400 Python questions from Stack Overflow	errors can be inserted in the protocol definitio n and the search string maynot contain all the relevant keywor ds, causing the loss	Programme rs might be seeking general help or asking for source code that is more efficient than the submitted code segment	Accurac y , precision , F1-score	

	27(3), 1-52.			of some valuable studies			
11	Zaib, M., Zhang, W. E., Sheng, Q. Z., Mahmood, A., & Zhang, Y. (2022). Conversational question answering: A survey. Knowledge and Information Systems, 1-45.	2022	An effort to present a comprehensive review of the state-of-the-art research trends of CQA primarily based on reviewed papers over the recent years	Code segments submitted as a part of Stack Overflow questions might not always be sufficient enough to reproduce the reported issues	The alculat type consists of characteristics such as comparisons, temporal reasoning, and anaphora, to make it more closely related to real-world challenges	F1-score , accuracy	This literature survey will serve as a quintessence for the researchers and pave a way forward for streamlining the research in this important area.
12	Cao, X., Zhao, Y., & Shen, B. (2022). Improving and evaluating complex question answering over knowledge bases by constructing strongly supervised data. Neural	2022	This paper provides an in-depth analysis of the model's ability to answer different types of questions, contributing a comprehensive evaluation of the performance of CQA models	The survey lacks the discussion on architectures based on traditional or flow-based models for CQA	This approach solves the problem that the model only grasps the topological structure of the knowledge graph while ignoring the textual information.	F1 and H1 scores	Add more types of logic alculati, such as negation logic alculati, when constructing query graphs to enrich the types of complex questions in the dataset.

	Computing and Applications, 1-21.						
13	Arora, R., Singh, P., Goyal, H., Singhal, S., & Vijayvargiya, S. (2021, March). Comparative question answering system based on natural language processing and machine learning. In 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS) (pp. 373-378). IEEE.	2021	This research is to build the best accuracy model using Glove and Infer Sent to use vector of various dimensionality to represent it numerically so that machine can interpret it.	The constraints were added during query generation to reduce the search space of the query graph. The lack of supervised data for answers at each reasoning step remains a major challenge.	System works in two types of domains: closed and opened domain	Accuracy	The further plans are, to reach better accuracies, increase the speed of performance and implementation of this model with deep learning algorithms to improve performance
14	Tommy, L., Kirana, C., & Riska, L. (2020, October). The Combination of Natural Language Processing and Entity Extraction for	2020	The created chatbot makes the academic information be accessed easily, quickly, and efficiently by students without having to zoom in and	In the model RNN bidirectional LSTM model, loss is quite high here Multino	The proposed chatbot can improve higher-education institutions current services, to reduce labour costs, and	Recall	

	Academic Chatbot. In 2020 8th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-6). IEEE.		zoom out.	mial Logistic Regression and Boost model	to create new innovative services.		
15	Deriu, J., Rodrigo, A., Otegi, A., Echegoyen, G., Rosset, S., Agirre, E., & Cieliebak, M. (2021). Survey on evaluation methods for dialogue systems. Artificial Intelligence Review, 54(1), 755-810.	2021	we differentiate between the various classes of dialogue systems (task-oriented,conversational, and question-answering dialogue systems)	NLP may not works properly when the alc ula information is different from the training one	In this paper the problem of evaluation is evolving in tandem to the progress of the dialogue system technology itself.	Mean reciprocal rank (MRR), Recall	
16	Kim, B., Seo, J., & Koo, M. W. (2021). Randomly wired network based on RoBERTa and dialog history attention for response selection. IEEE/ACM	2021	The experimental demonstration that feed-forward networks can be replaced by randomly wired neural networks and calculate the correlation between dialog history (i.e., context)	Drawback of this approach is that potentially correct utterances among the candidates could be	The final prediction was the response candidate having the highest probability of being true. They show better performance than simple deep neural	Mean reciprocal rank (MRR), Recall	

	Transactions on Audio, Speech, and Language Processing, 29, 2437-2442.		and the last utterance of the context, reflecting this correlation as an attention network that yielded an answer prediction.	regarded as incorrect	networks without skip alculatio.		
17	Sheth, V., & Damevski, K. (2022). Grouping related stack overflow comments for software developer recommendation on Automated Software Engineering, 29(2), 1-21.	2022	To assess the strength of the rela_x0002_tionship between selected pairs of Stack Overflow comments posted on a single question or answer	There was a problem, however, in that Recall @5 and recall@10 which results in low scores. This problem was related to the number of "NONE" entries to the prediction of the no answer problem.	User, reading one of the comments can be recommended another highly related common	F1 – score	The future work of this project includes investigating machine learning models that are more effective towards comment relatedness detection and clustering by examining additional features and algorithms.
18	Rai, S., Belwal, R. C., & Gupta, A.	2022	Emphasize the module name and	Comment hiding	Make a basic understandi	Accuracy	Creation of a bigger and

	(2022). Accurate module name prediction using similarity based and sequence generation models. Journal of Ambient Intelligence and Humanized Computing, 1-13.		propose the module name prediction approach. The sequence generation models can predict the module name tokens in a sequence, while similarity-based models only suggest pre-stored module names	mechanism, useful discussion and knowledge are often not easily alculat i to the user	ng of the building blocks of the alculat . The architecture is simpler than the skip-gram model, where we try to predict a group of context words for a given single target word.		more diverse corpus, and test the alcul a approaches could be interesting work in the future.
19	Kuhail, M. A., Alturki, N., Alramlawi, S., & Alhejori, K. (2022). Interacting with educational chatbots: A systematic review. Education and Information Technologies, 1-46.	2022	Chatbots on a web platform to teach computer science, language, general education, and a few other fields such as engineering and mathematics.	The corpus size is not very large compared to the alc ula alculat i method or code block level corpora.	This study will help the education and human-computer interaction community aiming at designing evaluating educational chatbots.	Largely point to high perceived usefulness. However a few participants pointed out that it was sufficient for them to learn with a human partner.	Future studies should explore chatbot localization, where a chatbot is customized based on the culture and context it is used in.
20	Bird, J. J., Ekárt, A., &	2022	An intelligent chatbot	Applied exclusio	Chatbot conversatio	Accuracy ,	Providing recommen

	Faria, D. R. (2021). Chatbot Interaction with Artificial Intelligence: human data augmentation with T5 and language transformer ensemble for text classification. Journal of Ambient Intelligence and HumanizedComputing, 1-16.		system based on Artificial Intelligence Markup Language (AIML) which can be used as an e-commerceassistant. This Chatbot is applied to the Telegram application.	n criteria to find relevant articles that were possible to assess. Decision might have caused a bias: for example, we could have alcula short papers presenting original ideas or papers without sufficient evidence.	ns are very accurate because the chatbot system can answer user questions well, are relevant and successfully carry out stock alculation, orders, and payments. Chatbots can also provide clear instructions to the user.	precision, F1-score	dations or automatic suggestions for user input is also needed to approach the correct conversation in future.

PERFORMANCE OF DIFFERENT ALGORITHMS :

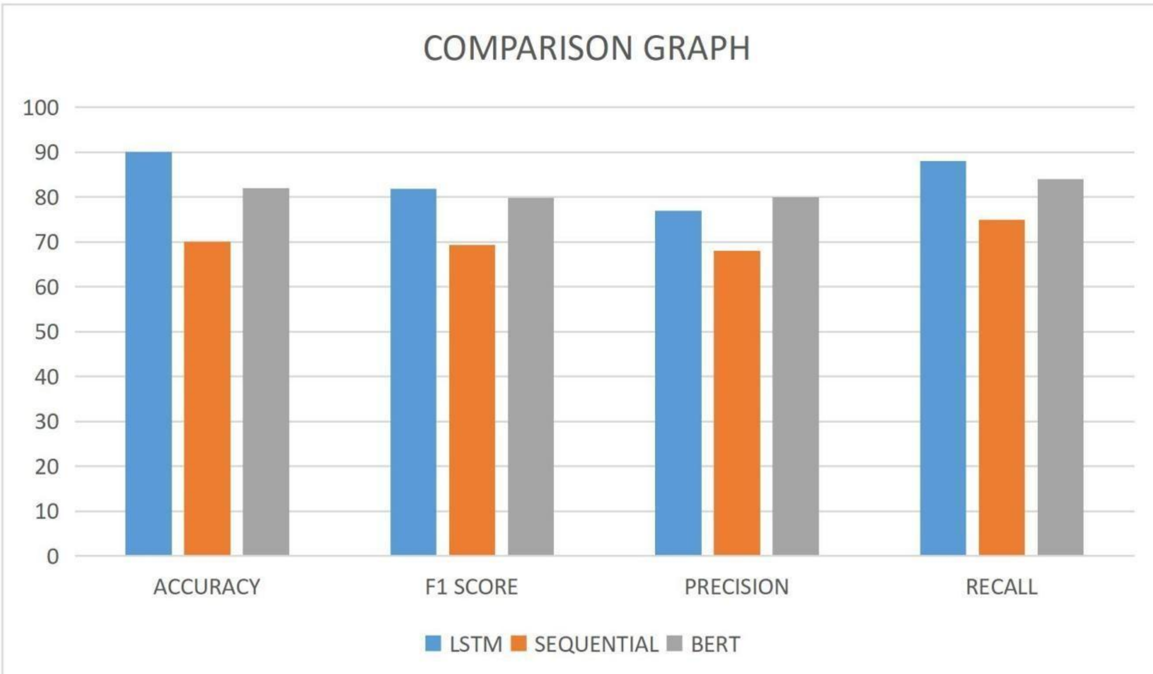

Figure 2.1 : Comparison of models

3.REQUIREMENT SPECIFICATION

Functional Requirements :

Functional requirements are those that describe the tasks and features that the AI-driven chatbot must perform, and they can be broken down into several categories:

User Input: The chatbot must be able to understand and interpret natural language input from users, including text, voice, and other inputs.

Response Generation: The chatbot must be able to generate responses that are appropriate, accurate, and relevant to the user's input. These responses may include text, voice, images, and other media.

Conversation Flow: The chatbot must be able to manage the flow of the conversation, including understanding the context of previous interactions, managing multiple conversation threads, and transitioning between topics.

Integration with other systems: The chatbot may need to integrate with other systems, such as databases, APIs, or third-party services, in order to provide relevant responses or actions.

Non-Functional Requirements :

Non-functional requirements are those that describe the performance and quality attributes of the chatbot, such as:

Performance: The chatbot should be able to handle a large volume of user requests and respond quickly to user inputs.

Accuracy: The chatbot should be accurate in understanding user inputs and generating appropriate responses.

Security: The chatbot should be designed with security in mind, including protecting user data and preventing unauthorized access.

Scalability: The chatbot should be able to handle increasing volumes of traffic and interactions as user adoption grows.

Usability: The chatbot should be easy to use and understand for users, with an intuitive interface and clear instructions.

Software Requirements :

Integrated Python Notebook with installed python libraries – TensorFlow , transformers and Natural language Processing toolkit and other data handling libraries like pandas ,NumPy etc.

The scope of the project can be improved by feeding the models with more data on errors regarding programming and make it reliable for any kind of programming language. Since object-relational mapping is used by question-answering systems like stack-over-flow , the project may work completely with machine intelligence and no more uses key-to-value matched answers for predicting the reason for the error given as input.

Using efficient framework to deploy the model , to decrease the waiting time for the user may increase the usability of the application. The more focus is kept on running th e model rather than importing files of css or front-end framework.

4. SYSTEM ANALYSIS AND DESIGN

4.1 Existing methodology :

The existing methodology for developing chatbots using NLP and deep learning typically involves the following steps:

Data acquisition: The first step involves collecting a large dataset of user queries and responses related to a specific domain or topic. This dataset is used to train the chatbot using machine learning algorithms.

Preprocessing: To remove irrelevant information, correct spelling and grammatical errors, and convert the text data into a machine-readable format.

NLP: The preprocessed data is then fed into an NLP pipeline that uses techniques.

Deep Learning: The extracted features are then fed into a deep learning model, such as a recurrent neural network (RNN) or a transformer, to learn the mapping between user queries and responses.

Evaluation: The trained chatbot is then evaluated using various metrics such as accuracy, precision, and recall.

4.2 Proposed Architecture

- The purpose of the Chatbot is to accompany Naïve programmers by recommending possible answers to their queries.
- The data required for training the model is collected from various online resources like stack overflow etc.
- Initially the data-set is converted to json format.
- Preprocessing the text :The text collected must be cleaned and the words must be converted to numerical representation.
- The numerical vectors must be splitted into training and validation sets.
- After encoding the text , we have to define the model architecture. This will determine the structure of the neural network. This includes choosing the type of layer, the number of nodes in each layer, and the activation function.
- Train the model: Train the model using the preprocessed data. This involves providing the model with input and output data, and adjusting the weights of the network to minimize the error.
- Evaluate the model to see how well it’s performing. This can be done by comparing the

model's predictions with the actual outputs and calculating metrics such as accuracy and loss.

- Saving the model: Save the trained model for later use. This allows you to use the same model for encoding and decoding the chatbot.
- Encode the chatbot: Use the trained model to encode the chatbot. This involves feeding the input data (questions asked by the user) into the model and using the output data (the encoded representation of the input) to generate a response.
- Decode the chatbot: Use the trained model to decode the encoded representation of the input and generate a response. This involves using the encoded representation as input to the model and using the output to generate a response.
- Finally test the chatbot by asking some queries and checking if it gives expected results good conversation flow of chatbot. Deploy the chatbot using a python back-end framework and maintain the chatbot with regular updation.

Algorithms :

Seq2Seq models: These are sequence-to-sequence models that are well-suited for building chatbots. Seq2Seq models consist of an encoder and a decoder, which respectively encode the input sequence into a fixed-length vector and then decode the vector into the output sequence.

The Long Short-Term Memory (LSTM) is used by the seq2seq model, commonly known as the encoder-decoder model, to generate text from the training corpus. Additionally important in machine translation applications is the seq2seq paradigm.

This strategy for text generation from user input will be used when developing our generative chatbot.

Seq2Seq has the benefit that modern translation systems may produce arbitrary output sequences after seeing the complete input, particularly with the use of LSTMs. To aid in producing a viable translation, they can even automatically zero in on portions of the input.

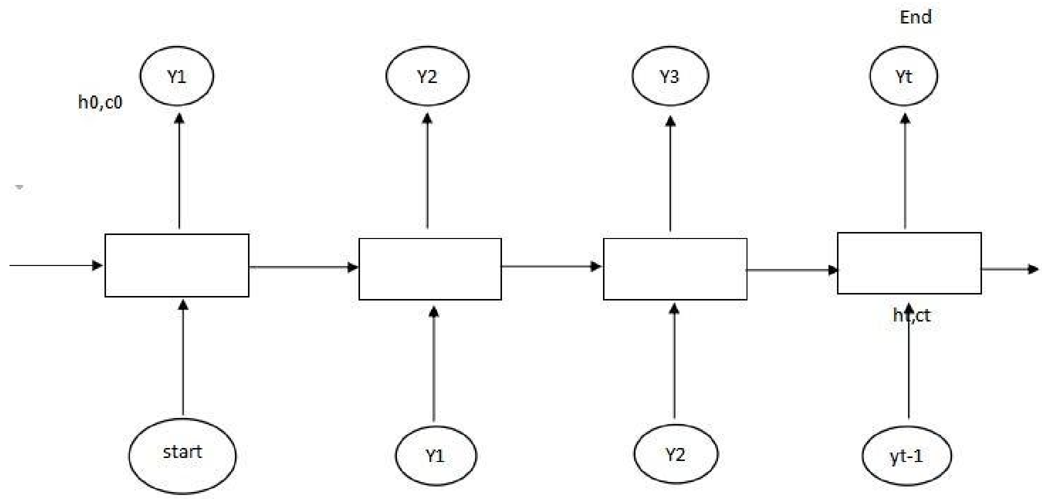

Figure 4.1 : Working of SeqToSeq model

Following the training procedure, the training accuracy is 53.35%, the validation accuracy is 52.77%, and the training loss and validation loss are, respectively, 0.0720 and 0.1137.

BERT: Question classification task for user intent classification combining BERT with capsule network, incorporates focal loss. Input embedding contained token information, position information, and segment information were fed into stacked transformers encoder which employed pre-trained language model BERT.

Bidirectional Encoder Representations from Transformers is abbreviated as BERT. Transformer encoders are layered on top of one another to form the BERT design. A feed-forward layer and a self-attention layer are the two sub-layers that make up each Transformer encoder.

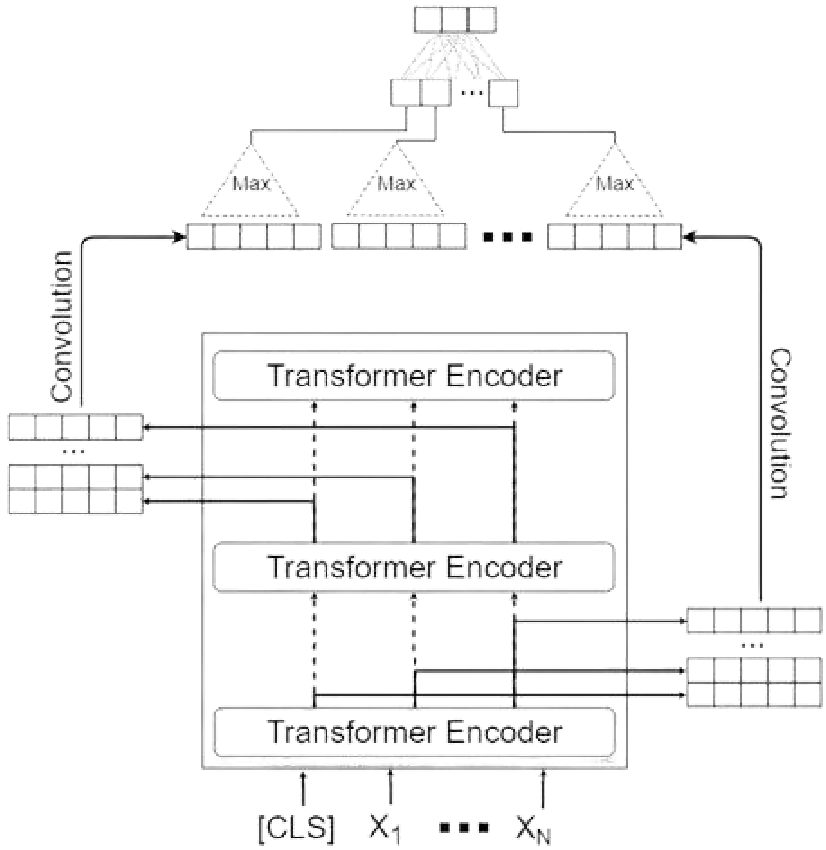

Figure 4.2 : Working of BERT Model

BERT is a free and open-source machine learning framework for dealing with natural language (NLP). BERT uses the surrounding text to provide context in order to help computers understand the meaning of ambiguous words in text. BERT accepts an input sequence and moves it up the stack continuously. It is first sent through a Self-Attention layer at each block, after which it is sent to a feed-forward neural network. It is given to the following encoder. Each position will ultimately produce a hidden size vector (768 in BERT Base).

GORU: Gated Orthogonal recurrent Unit were a novel RNN based that combines remembering ability of unitary RNNs with the ability of gated RNNs to effectively forget redundant or irrelevant information in memory applied the Gated Orthogonal Recurrent Unit (GORU) in an end-to-end fashion. The RNNs with GORU were a novel RNN-based model that combined unitary RNNs' capacity for remembering with gated RNNs' capacity for effectively forgetting redundant or irrelevant information in memory.

LSTM model: LSTMs (Long Short-Term Memory) are a type of Recurrent Neural Network (RNN) that can effectively handle the sequential nature of chatbot input and output. LSTMs have memory cells that allow them to remember past information, which is useful in a chatbot context where the response is generated based on the entire conversation history.
LSTM has three main gates : Forget gate ,input gate ,output gate.

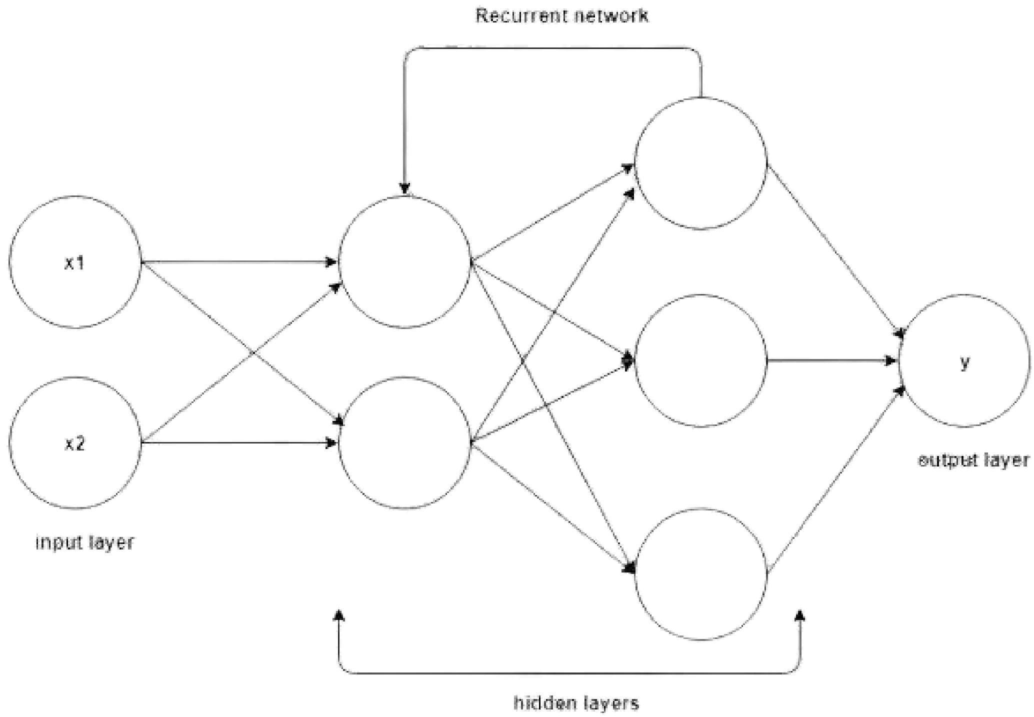

Figure 4.3 : Working of LSTM model

We forecast the following character of a given word in a sequence for the following generation. Text data can be viewed as a series of words or a series of discrete pieces of information. We have employed deep learning models like RNN/LSTM/GRU for sequence prediction.

4.3 PROCESS-FLOW :

The process model elaborates the keen steps involved in the application model from collecting data to modelling thereby deployment of the web application. The database along with functionality can be observed with the figure. It encases all the possible actions happening the model building and deployment.

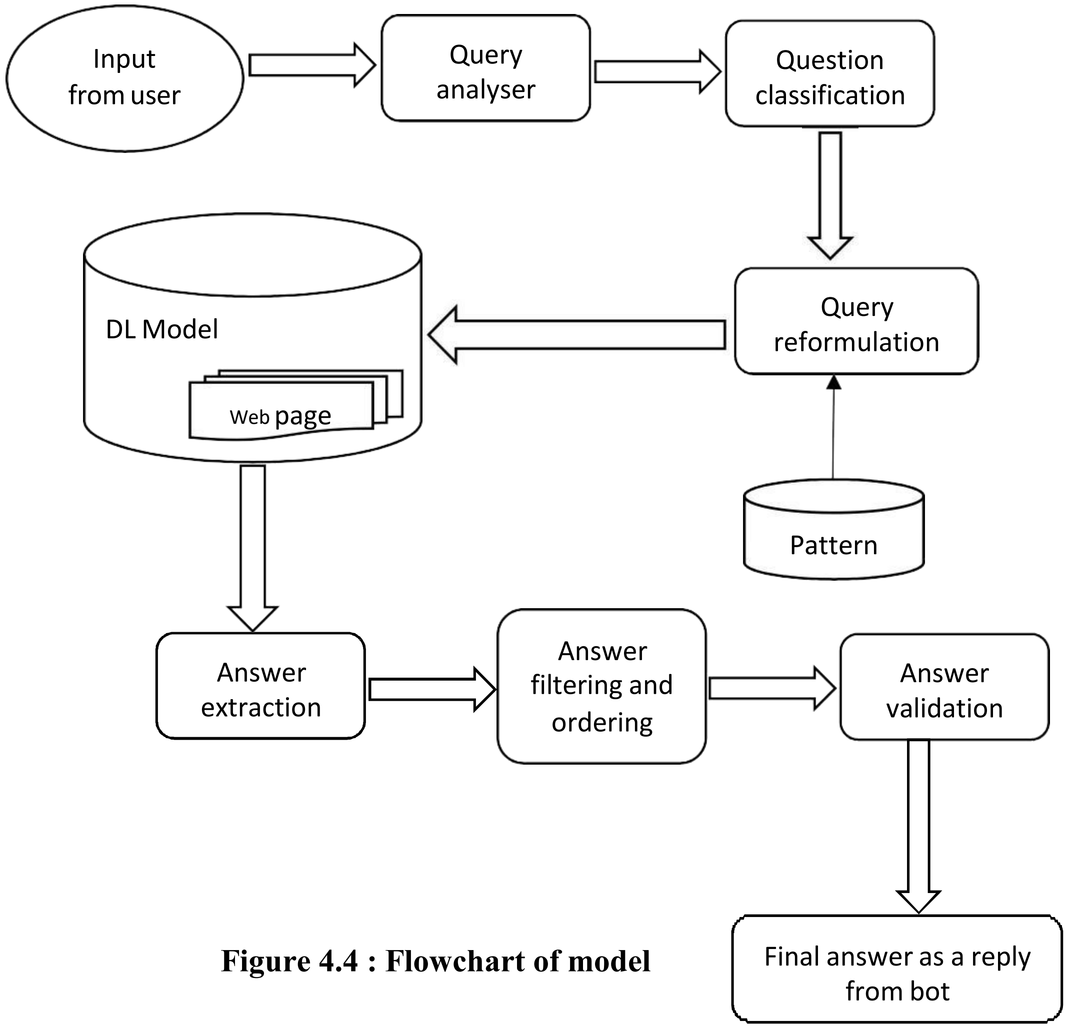

Figure 4.4 : Flowchart of model

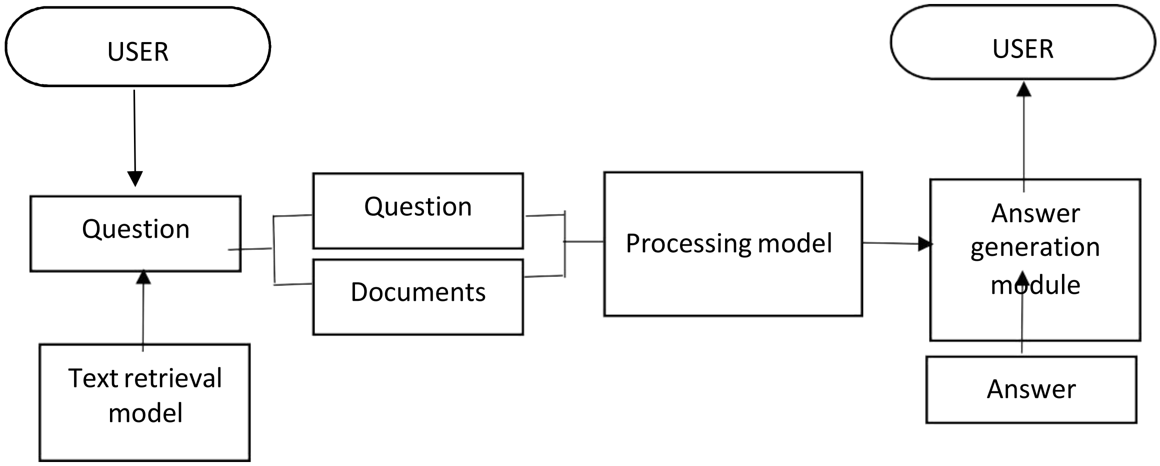

Figure 4.5 : Flowchart of deployment

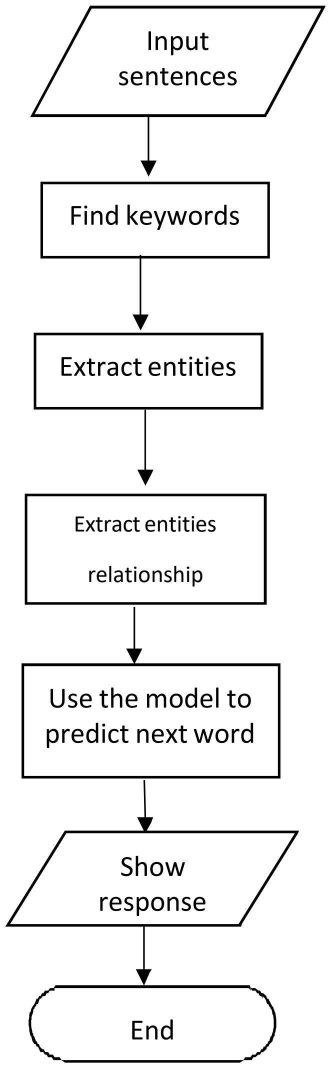

Figure 4.6 : Flowchart of deployment

5. IMPLEMENTATION

How the model is trained :

The dataset we choose for training is a json file with tags, patterns, responses as the key attributes.

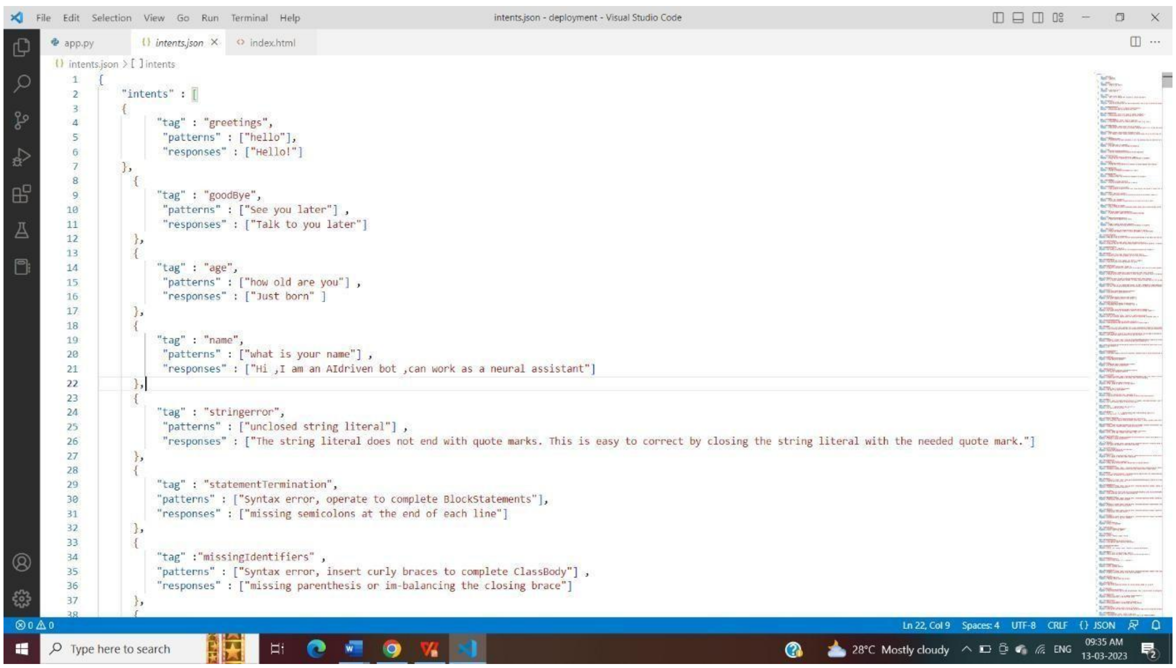

Figure 5.1 : Preview of dataset

Initially we import all the required libraries for training the model in python. All the models are imported and json file is loaded for training.

```python
import json
import random
import torch
import torch.nn as nn
from transformers import AutoTokenizer, AutoModel

with open('data.json', 'r') as f:
    data = json.load(f)

model_names = ['bert-base-uncased', 'roberta-base', 'distilbert-base-uncased']
models = [AutoModel.from_pretrained(name) for name in model_names]
tokenizers = [AutoTokenizer.from_pretrained(name) for name in model_names]
```

Figure 5.2 : Import of dataset & libraries

Now we define a method to convert the given sentence of bag of words into a numerical representation with frequency of the corresponding word as the point.

```
def predict_class(sentence):
    bow = bag_of_words(sentence)
    res = model.predict(np.array([bow]))[0]
    ERROR_THRESHOLD = 0.25
    results = [[i,r] for i, r in enumerate(res) if r > ERROR_THRESHOLD]
    results.sort(key=lambda  x:x[1], reverse=True)
    return_list = []
    for r in results:
        return_list.append({'intent': classes[r[0]], 'probability': str(r[1])})
    return return_list
```

Figure 5.3 : Bag of words

In the above code , after we use the bag_of_words , we use the predict method to predict the probability of each intent based on the bag of words representation of the sentence. The ERROR_THRESHOLD is used to filter out low-confidence predictions. The next line filters out the low-confidence predictions and creates a list results that contains the index of each intent and its corresponding probability, for all intents whose probability is above the ERROR_THRESHOLD. The next line sorts the results list in descending order of probability. Then it initializes an empty list return_list that will contain the predicted intents and their corresponding probabilities. Then we append a dictionary containing the predicted intent and its corresponding probability to return_list. The classes list contains the names of all possible intents, so classes[r[0]] returns the name of the predicted intent. Then the result is returned with containing all predicted intents and their corresponding probabilities in descending order of probability.

```
def clean_up_sentence(sentence):
    sentence_words = nltk.word_tokenize(sentence)
    sentence_words = [lemmatizer.lemmatize(word)  for word in sentence_words]
    return sentence_words

def bag_of_words(sentence):
    sentence_words= clean_up_sentence(sentence)
    bag = [0] * len(words)
    for w in sentence_words:
        for i, word in enumerate(words):
            if word == w:
                bag[i] = 1
    return np.array(bag)
```

Figure 5.4 : predict method

The above code are used for tokenization and stemming up the data for further process continual.

```
class ChatBot(nn.Module):
    def __init__(self):
        super(ChatBot, self).__init__()
        self.hidden_size = sum([model.config.hidden_size for model in models])
        self.fc1 = nn.Linear(self.hidden_size, self.hidden_size)
        self.fc2 = nn.Linear(self.hidden_size, self.hidden_size)
        self.fc3 = nn.Linear(self.hidden_size, len(data))

    def forward(self, input_ids, attention_mask):
        pooled_outputs = []
        for model, tokenizer in zip(models, tokenizers):
            input_tokens = tokenizer(input_ids, padding=True, truncation=True, return_tensors='pt')
            outputs = model(**input_tokens)
            pooled_outputs.append(outputs.pooler_output)
        pool (module) nn orch.cat(pooled_outputs, dim=1)
        x = nn.functional.relu(self.fc1(pooled_output))
        x = nn.functional.relu(self.fc2(x))
        x = self.fc3(x)
        return x

batch_size = 8
learning_rate = 1e-5
num_epochs = 10
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
model = ChatBot().to(device)
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(model.parameters(), lr=learning_rate)
```

Figure 5.5 : TensorFlow usage

This model takes as input her Input_ids and Attention mask, a tokenized and encoded version of the user's input text. This model uses several pre-trained models and their corresponding tokenizers to extract features from the input text. These features are then concatenated and passed through three fully connected layers (fc1, fc2, and fc3) to create a response prediction. The model's hidden size is calculated as sum of hidden sizes of all pre-trained models used in Chatbot. This hidden size is used to determine the size of the input and output of the fully connected layer. The Chatbot model's forward method first iterates over each pre- trained model and its corresponding tokenizer to get the model's output. These outputs are then concatenated along her second dimension and passed through a fully connected layer to produce the final prediction. Overall, this model is designed as a chatbot that can receive user input and generate responses based on trained models.

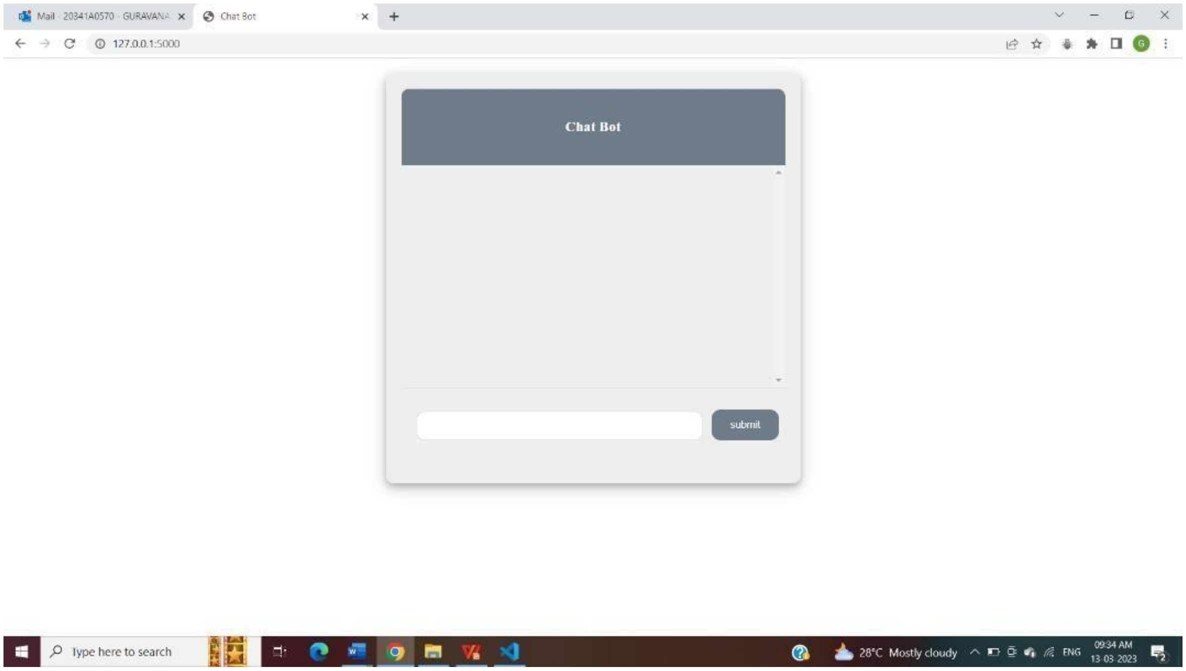

Figure 5.6 : Sample GUI

The backend was made using flask library with other in-built python libraries.The data is rendered through http response and displayed through jinja syntax engulfed in template file.

6. RESULTS AND DISCUSSIONS

6.1 Comparison metrics

Table 6.1: Results Comparison Table

Algorithm	**Accuracy**	**F1-score**
BERT(ensemble)	80.9	81.62
ALBERT(ensemble)	89.4	77.37
LSTM	85.4	77.0
SeqtoSeq	80.5	60.0

Existing System :

Chatbots these days , mainly use NLP to better understand and reply to the human query. They are integrated with other technologies like voice assistants ,virtual reality. They have multi-channel support and are personalized. They have good contextual analysis and analyse the given input pro-actively.

Building a BERT-based triple classification model for triple classification by using the triples generated from all meta paths and building a BERT-based text classification model for the content of triples by converting the generated triples into text and accomplish text classification problems.

Proposed System :

In the model building , we mainly prefer in improving sentimental analysis and a accurate predictive model which is trained with fine tuning.

Ensemble methods, where multiple models are combined, can improve the accuracy of a chatbot by combining the strengths of different models and reducing the impact of individual model weaknesses.

Robust Training Data: A chatbot's accuracy is only as good as the training data it receives, so it's crucial to provide it with a large, diverse, and representative set of data to learn from.

6.2 Results with GUI obtained

GUI Results obtained : The data which is trained is prepared using the errors which are more prone for beginners learning java. The data can be made using multi-programming languages.

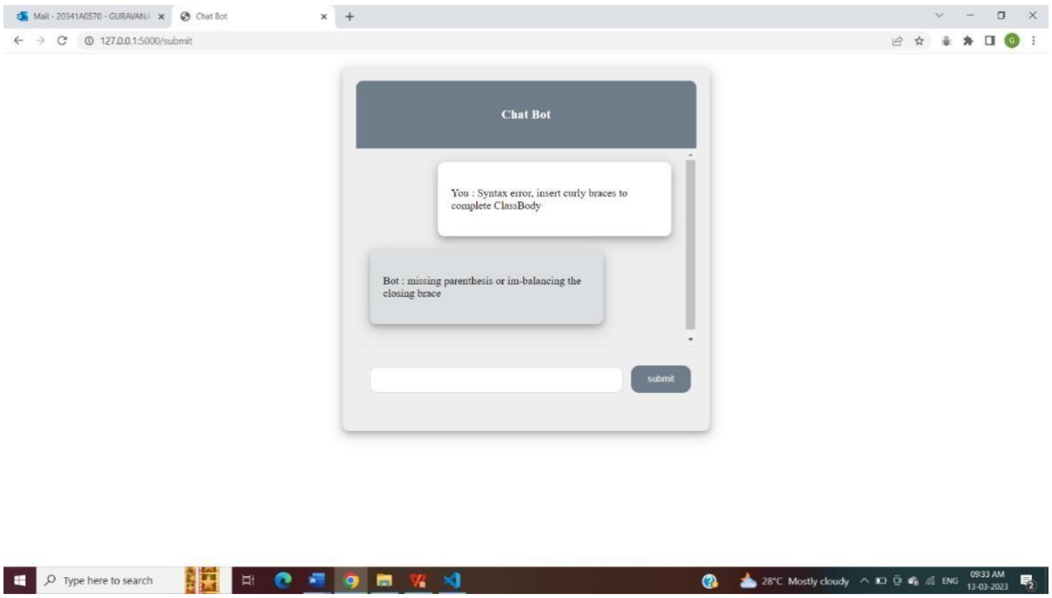

Figure 6.1 : Giving Input queries

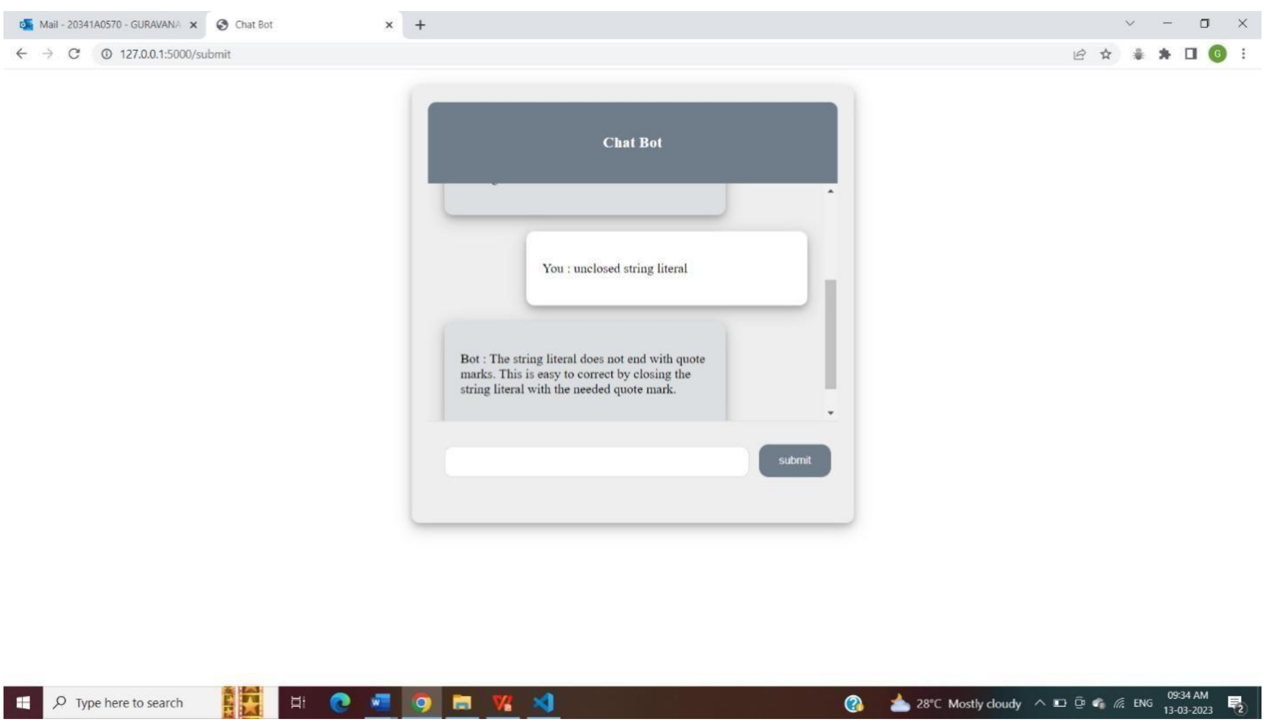

Figure 6.2 : Display of results

Existing System -Comparison metrics :

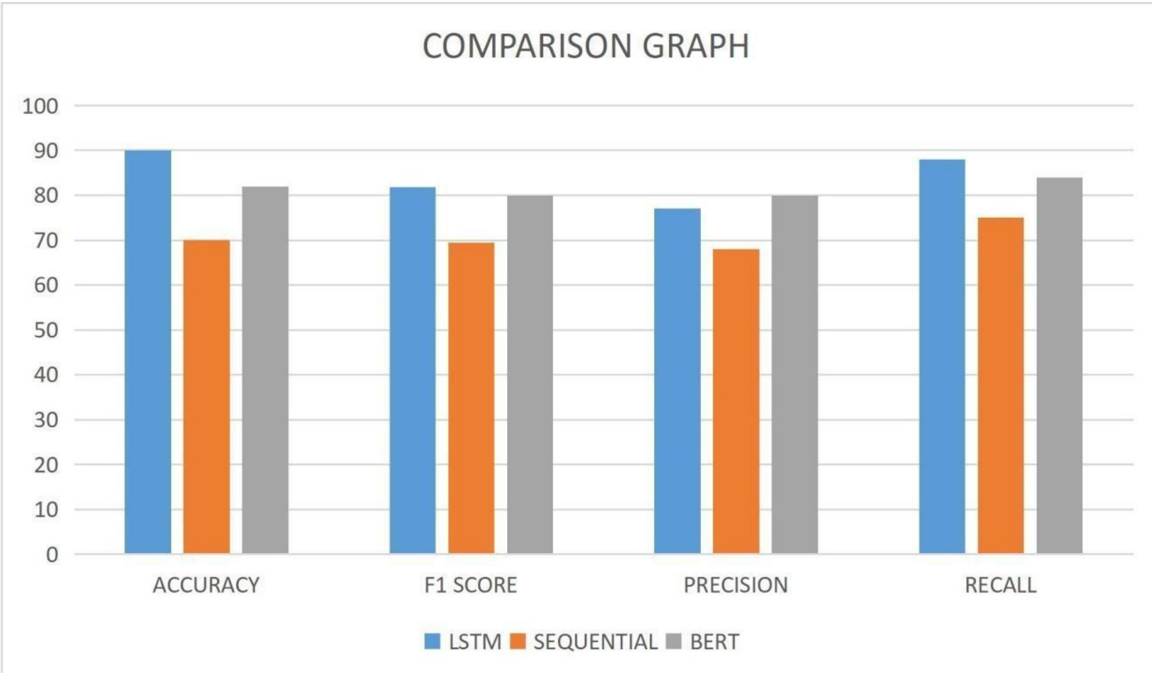

Figure 6.3 : Comparison graph

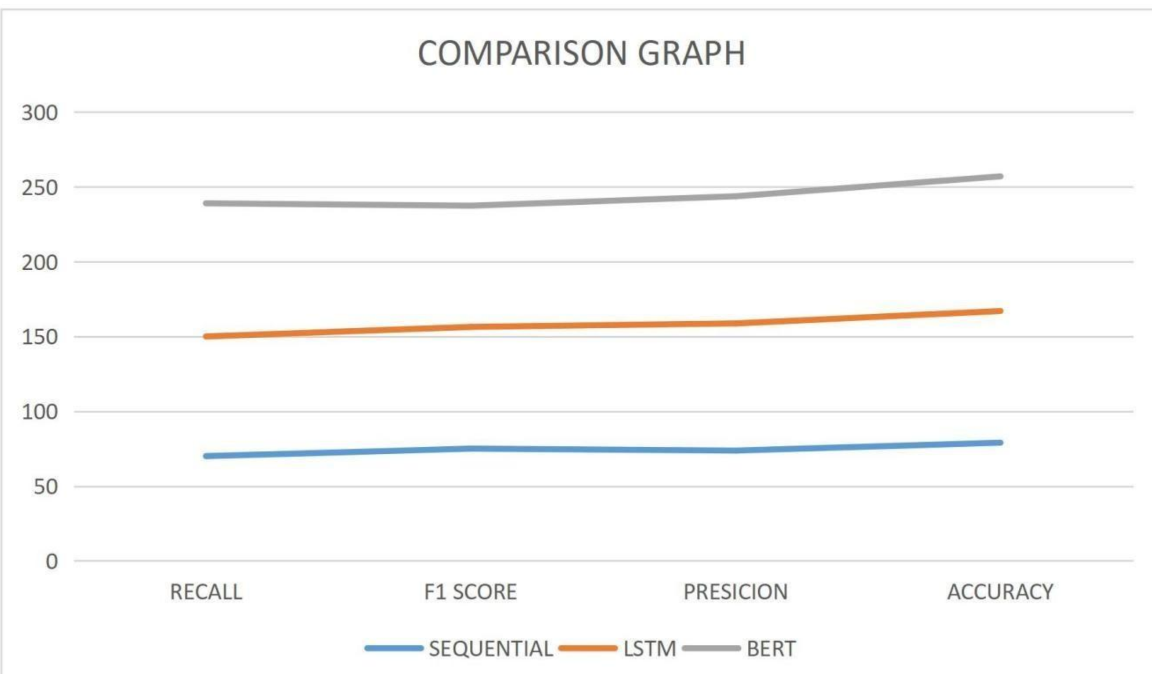

Figure 6.4 : Metrics in references

Proposed Architecture :

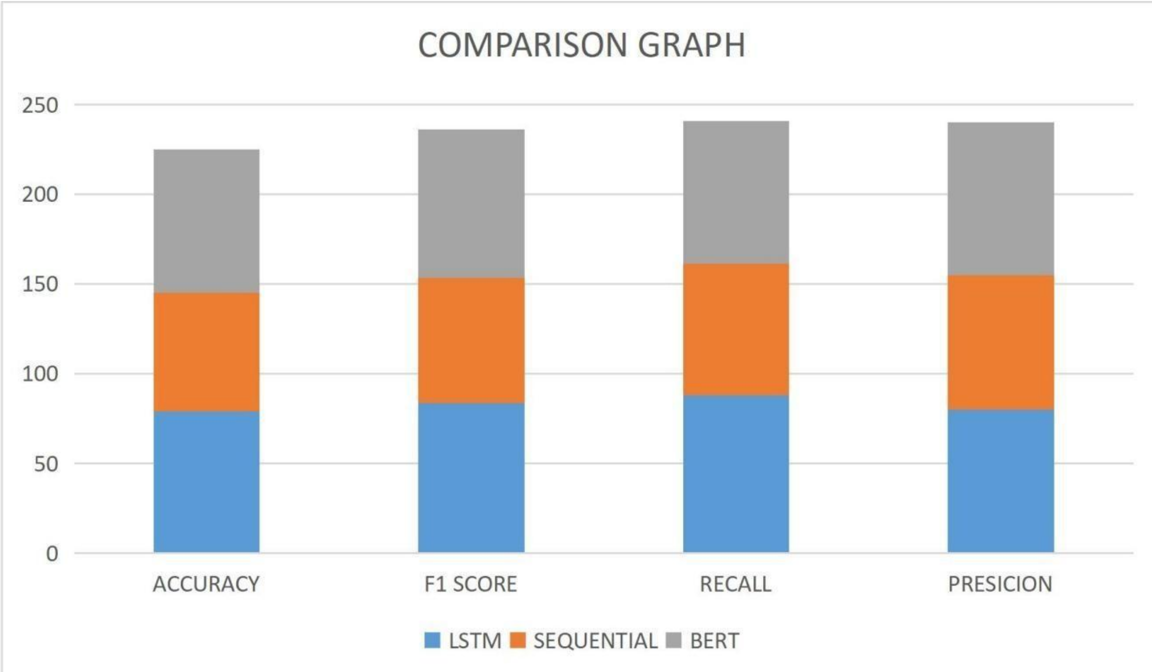

Figure 6.5 : Proposed model accuracy

7. CONCLUSION AND FUTURE SCOPE

Using NLP and deep learning, chatbots have become popular and effective tools for businesses to improve customer engagement and services. These chatbots can use advanced algorithms to understand and respond to natural language queries, providing users with personalized and helpful responses. Using machine learning, chatbots can learn from previous interactions to improve responses and provide a better customer experience. Overall, chatbots using NLP and deep learning offer significant benefits such as increased efficiency, reduced costs, and improved customer satisfaction. The implementation of an AI-driven chatbot has the potential to change the way the software interact with their clients. With its advanced language proc. While there are some challenges to be overcome, such as ensuring the chatbot has sufficient knowledge and the ability to handle unexpected scenarios, the benefits of implementing an AI-driven chatbot far outweigh the demerits. As AI technology continues to develop, chatbots will become more sophisticated, making them a valuable part of any service provided online.

The future scope of chatbots using NLP and deep learning is very promising and demanding in every sector. Advances in natural language processing and deep learning techniques are making chatbots more sophisticated and able to understand and respond to complex user requests. Additionally, chatbots can provide users with a more personalized experience based on past interactions and preferences. Additionally, chatbots can be integrated with other emerging technologies such as voice assistants, augmented reality, and virtual reality to create seamless and intuitive user experiences. As more companies adopt chatbots, the demand for developers and data scientists with expertise in natural language processing and deep learning will rise. Overall, the future looks bright for chatbots using NLP and deep learning, and could become an essential tool for businesses to improve customer engagement and automate operations.

Reference Papers:

[1] Hao, T., Li, X., He, Y., Wang, F. L., & Qu, Y. (2022). Recent progress in leveraging deep learning methods for question answering. *Neural Computing and Applications*, 1-19.

[2] Nassiri, K., & Akhloufi, M. (2022). Transformer models used for text-based question answering systems. Applied Intelligence, 1-34.

[3] Abdel-Nabi, H., Awajan, A., & Ali, M. Z. (2022). Deep learning-based question answering:a survey. Knowledge and Information Systems, 1-87.

[4] Do, P., & Phan, T. H. (2022). Developing a BERT based triple classification model using knowledge graph embedding for question answering system. Applied Intelligence, 52(1), 636- 651.

[5] Abbasiantaeb, Z., & Momtazi, S. (2022). Entity-aware answer sentence selection for question answering with transformer-based language models. Journal of Intelligent InformationSystems, 59(3), 755-777.

[6] Lilian, J. F., Sundarakantham, K., & Shalinie, S. M. (2021). Anti-negation method for handling negation words in question answering system. The Journal of Supercomputing, 77(5), 4244-4266.

[7] Wang, J., Jatowt, A., Färber, M., & Yoshikawa, M. (2021). Improving question answering for event-focused questions in temporal collections of news articles. Information Retrieval Journal, 24(1), 29-54.

[8] Gomes, J., de Mello, R. C., Ströele, V., & de Souza, J. F. (2022). A study of approaches to answering complex questions over knowledge bases. Knowledge and Information Systems, 1- 33.

[9] Alzubi, J. A., Jain, R., Singh, A., Parwekar, P., & Gupta, M. (2021). COBERT: COVID-19 question answering system using BERT. Arabian journal for science and engineering, 1-11.

[10] Mondal, S., Rahman, M. M., Roy, C. K., & Schneider, K. (2022). The reproducibility of programming-related issues in Stack Overflow questions. Empirical Software Engineering, 27(3), 1-52.

[11] Zaib, M., Zhang, W. E., Sheng, Q. Z., Mahmood, A., & Zhang, Y. (2022). Conversational question answering: A survey. Knowledge and Information Systems, 1-45.

[12] Cao, X., Zhao, Y., & Shen, B. (2022). Improving and evaluating complex question answering over knowledge bases by constructing strongly supervised data. Neural

Computing and Applications, 1-21.

[13] Arora, R., Singh, P., Goyal, H., Singhal, S., & Vijayvargiya, S. (2021, March). Comparative question answering system based on natural language processing and machine learning. In 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS) (pp. 373-378). IEEE.

[14] Tommy, L., Kirana, C., & Riska, L. (2020, October). The Combination of Natural Language Processing and Entity Extraction for Academic Chatbot. In 2020 8th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-6). IEEE.

[15] Deriu, J., Rodrigo, A., Otegi, A., Echegoyen, G., Rosset, S., Agirre, E., & Cieliebak, M. (2021). Survey on evaluation methods for dialogue systems. Artificial Intelligence Review, 54(1), 755-810.

[16] Kim, B., Seo, J., & Koo, M. W. (2021). Randomly wired network based on RoBERTa and dialog history attention for response selection. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29, 2437-2442.

[17] Sheth, V., & Damevski, K. (2022). Grouping related stack overflow comments for software developer recommendation Automated Software Engineering, 29(2), 1-21.

[18] Rai, S., Belwal, R. C., & Gupta, A. (2022). Accurate module name prediction using similarity based and sequence generation models. Journal of Ambient Intelligence and Humanized Computing, 1-13.

[19] Kuhail, M. A., Alturki, N., Alramlawi, S., & Alhejori, K. (2022). Interacting with educational chatbots: A systematic review. Education and Information Technologies, 1-46.

[20] Bird, J. J., Ekárt, A., & Faria, D. R. (2021). Chatbot Interaction with Artificial Intelligence: human data augmentation with T5 and language transformer ensemble for text classification. Journal of Ambient Intelligence and Humanized Computing, 1-16.