**Naan Mudhalvan**

# **CHATBOT IN PYTHON**

**Documentation**

|  |  |
| --- | --- |
| **Name** | DARSHAN . C |
| **Naan Muhalvan Id** | au732121104012 |
| **Group** | III |
| **College** | Nandha College Of Techonology |
| **Date** | 01.11.2023 |

**CHATBOT IN PYTHON**

# Table of Contents

* Executive Summary
* Problem Statement
* Design Thinking Process
* Phase 1: Problem Definition and Design Thinking
  + Problem Statement
  + Design Thinking
* Phase 2: Innovation
* Phase 3: Development Part 1
  + Step 1: Importing Libraries
  + Step 2: Loading the Dataset
  + Step 3: Preprocess data
  + Step 4: Perform Analysis
* Phase 4: Development Part 2
  + Step 1: Model Selection
  + Step 2: Model Training
  + Step 3: Model Evaluation
  + Step 4: Deployment
* Phase 5: Project Documentation & Submission
  + Documentation
    - Problem Statement and Design Thinking
    - Dataset and Data Preprocessing
    - Machine Learning Algorithm and Model Training
    - Innovative Techniques
  + Submission

# Executive Summary

This document presents a comprehensive overview of the Chatbot in python, including problem definition, design thinking, development phases, and innovative techniques used throughout the project.

## **Problem Statement**

The project aims to develop a chatbot in python with features of the Artificial Intelligence to improve the chatbot and give good responses to user.

## **Design Thinking Process**

The design thinking process played a leading role in shaping the project. It began with a deep understanding of the problem statement, followed by innovative solutions and structured development phases. The key design principles included data exploration, preprocessing, feature engineering, model selection, evaluation and deployment.

## **Phases of Development**

1. **Phase 1:** Problem Definition and Design Thinking

* Defined the problem statement.
* Conceptualized the system's aims and scope.

1. **Phase 2:** Innovation

* Explored innovative approaches to improve accuracy of Chatbot
* Use pretrained model to enhance quality of responses.

1. **Phase 3:** Development Part 1

* Importing Libraries.
* Loaded the dataset
* Preprocess the Dataset
* Perform Analysis

1. **Phase 4:** Development Part 2

* Model Training
* Model Selection
* Hyper parameter Tuning
* Deployment

## **Dataset and Preprocessing**

The project used a dataset that consists of a simple conversation between a user and a bot that have some basic questions and reply for those questions. We can preprocess the data by checking for missing values and clean by checking for tokens in the dataset.

## **Model Selection and Evaluation**

Here we used Pre trained GPT-2 model for training the chatbot using the Hugging Face repository that is in the Transformers Library.

## **Innovative Techniques**

Innovation was a core element of this project. While innovative techniques are continuously evolving, this system embraces the latest advancements in data preprocessing, model selection, and evaluation, ensuring that it gives good response to questions.

# **Phase 1: Problem Definition and Design Thinking**

**Problem Understanding:**

The problem at hand is to develop an AI chatbot using the Python programming language. This chatbot should be capable of engaging in natural language conversations with users, answering questions, and potentially performing various tasks based on user input.

**Solution For Solving The Problem:**

To address the problem for building a chatbot using python, we propose the following approach:

**Proposed System design:**

Designing a system for an AI chatbot involves structuring the components and interactions necessary to build and deploy the chatbot effectively. Here's a proposed system design for your AI chatbot solution:

**Functionality:**

Define the scope of the chatbot's abilities, including answering common questions, providing guidance, and directing users to appropriate resources.

**User Interface :**

The user interacts with the chatbot through a user interface. Here we use a web based Interface.

**Responses:**

Plan responses that the chatbot will offer, such as accurate answers, suggestions, and assistance. The responses are trained by the machine learning process to give accurate and good suggestions to the questions that are asked by users to the chatbot

**Testing and Improvement:**

Continuously test and refine the chatbot's performance based on user interactions. Improvements can be done by taking feedback from the users and we can improve the defects and upgrade it to better versions.

# **Phase 2: Innovation**

## **Proposed System Design:**

Exploring innovative techniques like ensemble methods and deep learning architectures to improve prediction system accuracy and robustness involves a structured approach. Here are the steps you can follow:

**Data Preprocessing:**

Clean and preprocess your data, handling missing values, outliers, and feature engineering. High-quality data is crucial for accurate predictions.

**Select the Right Model:**

Choose a pre-trained language model that suits your project's requirements. GPT-3 is a powerful option, but there are other models like BERT, T5, and more, each designed for specific NLP tasks.

**Performance Evaluation:**

Continuously evaluate the quality of responses generated by the model using appropriate metrics. Make adjustments and refinements as necessary.

# **Phase 3: Development Part 1**

## **Step 1: Importing Libraries**

Libraries play a crucial role in the development of the AI-based Diabetes Prediction System. The following libraries were used in the project:

* **nltk :** is a library for natural language processing and text analysis in Python. It provides tools for working with human language data, such as tokenization, stemming, lemmatization, part-of-speech tagging, and more.
* **string:** The string module is a standard Python library that provides a collection of string constants, like punctuation characters, ASCII letters, and digits. It's often used in text processing tasks to manipulate and clean text data. For example, you can use it to remove punctuation or filter characters in your chatbot's text processing pipeline.
* **torch:** PyTorch is an open-source deep learning framework. In chatbot development, torch can be used for building and training neural networks, including sequence-to-sequence models, which are commonly used for chatbots. You can also use PyTorch to manipulate and process data for model training and deployment.
* **transformers (Hugging Face Transformers):** The transformers library from Hugging Face provides pre-trained models and tools for working with state-of-the-art natural language processing models. In chatbot development, you can use it to work with pre-trained models like GPT-2, BERT, or others for tasks like text generation, understanding user input, and providing context-aware responses. It simplifies the process of fine-tuning and using these models for chatbot applications.
* **torch.utils.data:** The torch.utils.data module is part of PyTorch and provides tools for handling and processing datasets in a format suitable for deep learning

## **Step 2: Loading the Dataset**

### **Dataset Source and Description**

The dataset is obtained from the Kaggle for the training of the chatbot. It is a text file that consists of the data for User and Bot .

### **Loading the Dataset**

The dataset was loaded into the Jupyter Notebook

## **Step 3: Preprocess the data**

### **Displaying the First Few Rows of the Dataset**

To understand the structure of the dataset, the first few rows were displayed.

Tokenization: Split the text into words or subword tokens.

Cleaning: Remove any irrelevant characters or symbols.

Formatting: Organize the data into a structure suitable for training.

## **Step 4: Data Analysis**

This includes the analysis of the dataset by checking for the missing values.

# **Phase 4: Development Part 2**

## **Step 1: Model Selection**

### **Description of Machine Learning Algorithm**

Here we use the GPT-2 , a pre trained model by Hugging Face. It consists of a pre trained model for the chatbot responses.

## **Step 2: Model training**

* + Split your dataset into training and validation sets.
  + Fine-tune a pre-trained language model or train a model from scratch using your conversation dataset.
  + Train the model to predict the next response in a conversation given the previous messages. This can be done using techniques like sequence-to-sequence models or other NLP architectures.

## **Step 3: Hyperparameter Tuning:**

## **Experiment with different hyperparameters (e.g., learning rate, batch size, model architecture) to optimize the performance of your chatbot.**

## **Step 4: Deployment:**

In this phase we will deploy our chatbot in a website using flask library. It consists of html, css file to make elements of the website and make the website attractive. We use a javascript file to make the website dynamic.

# **Phase 5: Project Documentation & Submission**

## **Documentation**

In this section, we supply an in-depth overview of the key aspects of the project, including the problem statement, design thinking process, dataset description, data preprocessing, machine learning algorithm, model training, and evaluation metrics.

### **Problem Statement and Design Thinking**

The problem addressed in this project is the development of chatbot using python. Chatbot consists of AI concepts that gives a good interactions to users and give solution to the questions of user.

The design thinking process began with the identification of the problem. It involved problem definition, understanding the scope, and outlining the aims of the project.

Design thinking was instrumental in conceptualizing the project's solution, which involved building a predictive model to train the chatbot. Potential solutions were explored through the design of machine learning algorithms, data preprocessing techniques, and the selection of evaluation metrics.

### **Dataset and Data Preprocessing**

The dataset used in this project is the "Simple dialogs for chatbot" obtained from Kaggle. It has simple dialogs that is used in daily life conversations, making it a suitable choice for training the prediction model.

Data preprocessing played a crucial role in preparing the dataset for model development. Missing values were addressed, and relevant features were selected based on domain knowledge and data analysis.

### **Machine Learning Algorithm, Model Training, and Evaluation Metrics**

The machine learning algorithm used is NLP that recognizes natural language but in our dataset we have only English questions and responses for training. NLP is used for tokenization.

Model training involved the training of the chatbot

For model evaluation, a set of metrics were chosen to supply a comprehensive understanding of the model's performance. These metrics included accuracy, precision, recall, F1-score, and ROC-AUC. These metrics collectively evaluated the model's correctness, its ability to minimize false positives and false negatives, and its capability to discriminate between individuals with and without diabetes.

## **Submission**

For project submission, all code files, including data preprocessing, model training, and evaluation steps, have been compiled. Additionally, a detailed README file is provided, offering clear instructions on how to run the code and listing the necessary dependencies.

The dataset source from Kaggle has been included, along with a brief description of the dataset's columns and purpose.

The project is made available on a GitHub repository for easy access and review by others.

Darshan.c

Nandha College Of Technology