

REAL TIME FINANCIAL ASSISTANT CHATBOT

A PROJECT REPORT

Submitted by

HARIHARAN E	92132223048
HARIHARAN M	92132223046
LOKESH	92132223080

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY



PSNA COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution, Affiliated to Anna University, Chennai)

DINDIGUL - 624622

OCTOBER 2024

PSNA COLLEGE OF ENGINEERING AND TECHNOLOGY,
(Autonomous Institution Affiliated to Anna University, Chennai)
DINDIGUL – 624622

BONAFIDE CERTIFICATE

Certified that this idea report “**REAL TIME FINANCIAL ASSISTANT CHATBOT**” is the bonafide work of “**HARIHARAN E (92132223048), HARIHARAN M (92132223046), LOKESH (92132223080)**” who carried out the idea work under my supervision

SIGNATURE	SIGNATURE
Dr. A. VINCENT ANTONY KUMAR, M.E, Ph.D., HEAD OF THE DEPARTMENT PROFESSOR & HEAD DEPARTMENT OF IT PSNA COLLEGE OF ENGINEERING TECHNOLOGY, DINDIGUL -624622	Dr. P. PRIYADHARSHINI M.E, Ph, ASSISTANT PROFESSOR DEPARTMENT OF IT PSNA COLLEGE OF ENGINEERING TECHNOLOGY, DINDIGUL -624622

Submitted for the idea on_____

ABSTRACT

In today's financial landscape, managing personal finances effectively can be complex and time-consuming. The financial assistant chatbot provides an intelligent solution by offering real-time financial guidance and task automation through conversational AI. Utilizing cutting-edge technologies such as natural language processing (NLP), machine learning (ML), and predictive analytics, the chatbot is designed to assist users in various financial tasks, including budgeting, expense tracking, investment advice, and financial goal setting. The system integrates speech-to-text, text-based queries, and contextual financial models to cater to different user needs, from simple financial inquiries to advanced financial planning.

The chatbot captures user input using advanced NLP techniques, which enable it to understand and process both spoken and written language. This input is then analyzed using machine learning algorithms that consider user behavior and preferences to generate personalized financial recommendations. The assistant can connect to banking and financial platforms, allowing users to automate routine tasks such as bill payments, fund transfers, and investment portfolio management. Through real-time data analysis, it provides users with up-to-date financial insights, while ensuring a user-friendly interface that simplifies financial decision-making.

Balancing speed, accuracy, and user-friendliness presents challenges, particularly when dealing with complex financial terms or varying user intents. However, with adaptive learning models and efficient processing methods, the chatbot ensures high-quality responses with minimal latency. Additionally, by incorporating domain-specific expertise, the system can handle specialized financial topics, such as tax regulations, investment strategies, and loan management, enhancing its reliability for professional and personal financial use cases.

The financial assistant chatbot has the potential to revolutionize financial management by making it more accessible and user-centric. It can support individuals in better understanding their financial health, help businesses streamline financial operations, and assist financial advisors in providing tailored services. As AI technology continues to evolve, the future of financial chatbots will see even more sophisticated, context-aware systems that empower users to achieve greater financial well-being and security.

INTRODUCTION

The financial assistant chatbot offers a solution by delivering intelligent financial guidance and task automation through a conversational interface. By leveraging advanced natural language processing (NLP), machine learning (ML), and predictive analytics, the chatbot can interpret user input, provide tailored financial advice, and automate routine tasks such as bill payments, investment tracking, and account monitoring. This technology facilitates seamless financial management, allowing users to engage in interactive conversations and receive actionable insights without the need for extensive financial knowledge.

In this paper, we explore the development, capabilities, and potential applications of a financial assistant chatbot, emphasizing its impact on personal finance and the challenges of optimizing the system for accuracy, user engagement, and scalability. We also discuss the benefits of integrating AI-driven financial tools into everyday life and the potential for such technologies to transform the way individuals and businesses approach financial management

PROBLEM STATEMENT

In today's complex financial landscape, managing personal finances can be challenging for individuals and businesses alike. The need for effective budgeting, expense tracking, investment planning, and financial goal setting often overwhelms users who lack extensive financial knowledge or access to professional financial advisors. Traditional financial management tools, such as spreadsheets, manual budgeting, or costly advisory services, often fail to provide the real-time insights, automation, and personalized advice needed for effective decision-making. As a result, there is a growing demand for a more accessible and intelligent financial management solution that can simplify these processes while providing personalized guidance.

The challenge lies in developing a financial assistant chatbot that not only delivers accurate and personalized financial advice but also automates routine financial tasks such as bill payments, portfolio monitoring, and expense categorization. The system must handle a wide range of user queries, from basic financial questions to complex investment strategies, while maintaining low response times to facilitate seamless user interactions. Additionally, the solution must be adaptable across different platforms, including mobile apps, web interfaces, and voice-activated devices, to ensure a user-friendly experience for diverse audiences. Addressing these challenges requires an advanced technological approach that integrates natural language processing (NLP), machine learning (ML), and predictive analytics to deliver a reliable and effective financial assistant chatbot.

CHALLENGES

1. **Accuracy:** Providing precise and personalized financial advice while considering individual user needs, financial goals, and the context of their financial situation.
2. **Latency:** Delivering real-time responses to user queries with minimal delays, which is essential for a smooth conversational experience.
3. **Natural Language Understanding:** Accurately interpreting diverse user inputs, including slang, abbreviations, or vague financial terms, and understanding context-specific queries.
4. **Cross-Platform Support:** Ensuring the chatbot functions seamlessly across various platforms, including mobile apps, web interfaces, and voice-activated devices.

PROPOSED MODEL

The proposed solution is an AI-driven financial assistant chatbot designed to help users manage personal finances through real-time guidance and task automation. The chatbot will utilize advanced Natural Language Processing (NLP), Machine Learning (ML), and predictive analytics to deliver accurate, personalized financial advice. It will support a range of financial tasks, including budgeting, expense tracking, investment management, and financial goal setting, providing users with actionable insights and automating routine financial activities.

The chatbot will be developed as a cross-platform application, accessible through web interfaces, mobile apps, and voice-activated devices to ensure a consistent user experience across various platforms. The system will integrate with banking and investment platforms to enable real-time account monitoring, transaction tracking, and automation of tasks like bill payments and savings transfers. Additionally, cloud-based processing will be used to manage complex computations and data analysis, ensuring that the chatbot provides timely responses with minimal latency.

To enhance the chatbot's financial expertise, domain-specific models will be incorporated to handle specialized topics such as tax planning, retirement strategies, and investment risk management. This will enable the system to offer more reliable advice across different financial contexts. The chatbot will also be equipped with adaptive learning capabilities to continuously improve its recommendations based on user behavior and preferences.

By offering an AI-powered financial assistant that delivers real-time, context-aware financial guidance and automation, this solution aims to simplify financial management for individuals and businesses. The chatbot will empower users to make informed financial decisions, achieve their financial goals, and reduce the complexity of managing personal finances.

SOURCE CODE

Backend code:(python)

```
import numpy as np
import tensorflow as tf
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing.sequence import pad_sequences
import pickle
import json
import re
import random
__tokenizer = None
__model = None
__lbl_enc = None
__intents = None
def load_artifacts():
    global __tokenizer
    global __model
    global __lbl_enc
    global __intents
    __model = load_model('chatbot_model.h5')
    with open('tokenizer.pkl', 'rb') as handle:
        __tokenizer = pickle.load(handle)
    with open('label_encoder.pkl', 'rb') as handle:
        __lbl_enc = pickle.load(handle)
    with open('intents1.json', 'r') as f:
        __intents = json.load(f)
def preprocess_text(pattern):
    text = []
    txt = re.sub('[^a-zA-Z\']', ' ', pattern)
    txt = txt.lower()
    txt = txt.split()
    txt = " ".join(txt)
    text.append(txt)
    return text
def generate_answer(pattern):
    if pattern.lower() in ['goodbye', 'bye']:
        response = "Goodbye! Have a great day!"
        print(response)
        return response

    load_artifacts()

    text = preprocess_text(pattern)
    x_test = __tokenizer.texts_to_sequences(text)
    x_test = pad_sequences([x_test], padding='post', maxlen=__model.input_shape[1])
    y_pred = __model.predict(x_test)
    y_pred = y_pred.argmax()
    tag = __lbl_enc.inverse_transform([y_pred])[0]

    # Get the corresponding responses for the predicted tag
    for intent in __intents['intents']:
        if intent['tag'] == tag:
            response = random.choice(intent['responses'])
            break
    return response

app.py:
from flask import Flask, render_template, request, jsonify
import util

app = Flask(__name__, template_folder='templates')
```

```

@app.route('/')
def hello():
    return render_template("index.html")

@app.route("/get_response", methods=['GET', 'POST'])
def get_response():
    user_message = request.form['userMessage']

    print(user_message)

    response = jsonify({
        "bot_msg": util.generate_answer(user_message)
    })
    response.headers.add("Access-control-Allow-Origin", "*")
    return response

if __name__ == "__main__":
    app.run(debug=True)

```

Frontend Code(HTML,Css)

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Financial assistant Chatbot</title>
    <link rel="stylesheet"
href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@20..48,100..700,0..1,-
50..200"/>
    <link rel="stylesheet" href="{{url_for('static', filename='styles.css')}}">
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
    <div class="wrapper">
        <div class="chatbot">
            <header><h1>Financial assistant</h1></header>
            <ul class="chatbox">
                <li class="chat incoming">
                    <span class="material-symbols-outlined">smart_toy</span>
                    <p>Hi there! <br>
                        How can i help you today?
                    </p>
                </li>
            </ul>

            <div class="chat-input">
                <textarea name="chat-input" id="chat-input" placeholder="Ask your financial doubts..."></textarea>
                <span id="send-button" class="material-symbols-outlined">send</span>
            </div>
        </div>
    </div>
    <script src="{{url_for('static', filename='script.js')}}"></script>
</body>
</html>

```

Css

```

@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600&display=swap');
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;

```

```

    font-family: "Poppins", sans-serif;
}
body{
    min-height: 100vh;
    display: flex;
    background: #fdf3ff;
}
.wrapper{
    margin: auto;
    width: 550px;
    max-width: 100%;
}
.chatbot{
    display: flex;
    flex-direction: column;
    width: 100%;
    height: 100%;
    border: 2px solid #ba15e4;
    border-radius: 15px;
    box-shadow: 0 15px 20px rgba(0, 0, 0, 0.2);
    background: #ffff;
}
.chatbot header{
    /* margin: 0;
    display: flex;
    justify-content: center;
    background: #ba15e4; */

    background: #ba15e4;
    width: 100%;
    border-radius: 15px 15px 0px 0px;
    padding: 16px 0;
    text-align: center;
}
.chatbot header h1{
    color: #fdf3ff;
}
.chatbot .chatbox{
    height: 520px;
    overflow-y: auto;
    padding: 15px 20px 70px;
}
.chatbox .chat{
    display: flex;
}
.chatbot .chat p{
    color: #480055;
    font-size: 1.2rem;
    padding: 12px 16px;
}
.chatbot .outgoing p{
    background: #f3ccff;
    border-radius: 10px 10px 0 10px;
}
.chatbot .incoming p{
    background: rgba(0, 0, 0, 0.05);
    border-radius: 10px 10px 10px 0px;
}

.chatbot .outgoing{
    margin: 20px 0px;
    justify-content: flex-end;
}

```



```

.chatbot .incoming span{
  height: 32px;
  width: 32px;
  color: #480055;
  align-self: flex-end;
  background: #fdf3ff;
  text-align: center;
  line-height: 30px;
  border: 1px solid #480055;
  border-radius: 5px;
  margin: 0 10px 7px 0;
}
.chatbot .chat-input{
  width: 100%;
  display: flex;
  gap: 5px;
  padding: 5px 30px;
  border-top: 1px solid #ba15e4;
}
.chat-input textarea{
  height: 60px;
  width: 100%;
  border: none;
  outline: none;
  font-size: 1.2rem;
  resize: none;
  padding: 16px 15px 16px 0;
}
.chat-input span{
  color: #480055;
  height: 60px;
  line-height: 60px;
  align-self: flex-end;
  font-size: 1.7rem;
  cursor: pointer;
  visibility: hidden;
}
.chat-input textarea:valid ~ span{
  visibility: visible;
}

```

script.js

```

const sendChatBtn = document.querySelector(".chat-input span");
const chatInput = document.querySelector(".chat-input textarea");
const chatBox = document.querySelector(".chatbox");
const url = "http://127.0.0.1:5000/get_response";
let userMsg;
const createChatLi = (message, className) => {
  const chatLi = document.createElement("li");
  chatLi.classList.add("chat", className);
  let chatContent = className === "outgoing" ? <p>${message}</p> : <span class="material-symbols-outlined">smart_toy</span><p>${message}</p>;
  chatLi.innerHTML = chatContent;
  return chatLi;
}
const generateResponse = () => {
  $.post(url, {
    userMessage: userMsg
  }, function(data, status){
    console.log(data.bot_msg.toString(), status);
    chatBox.appendChild(createChatLi(data.bot_msg.toString(), "incoming"));
  });
}

```

```

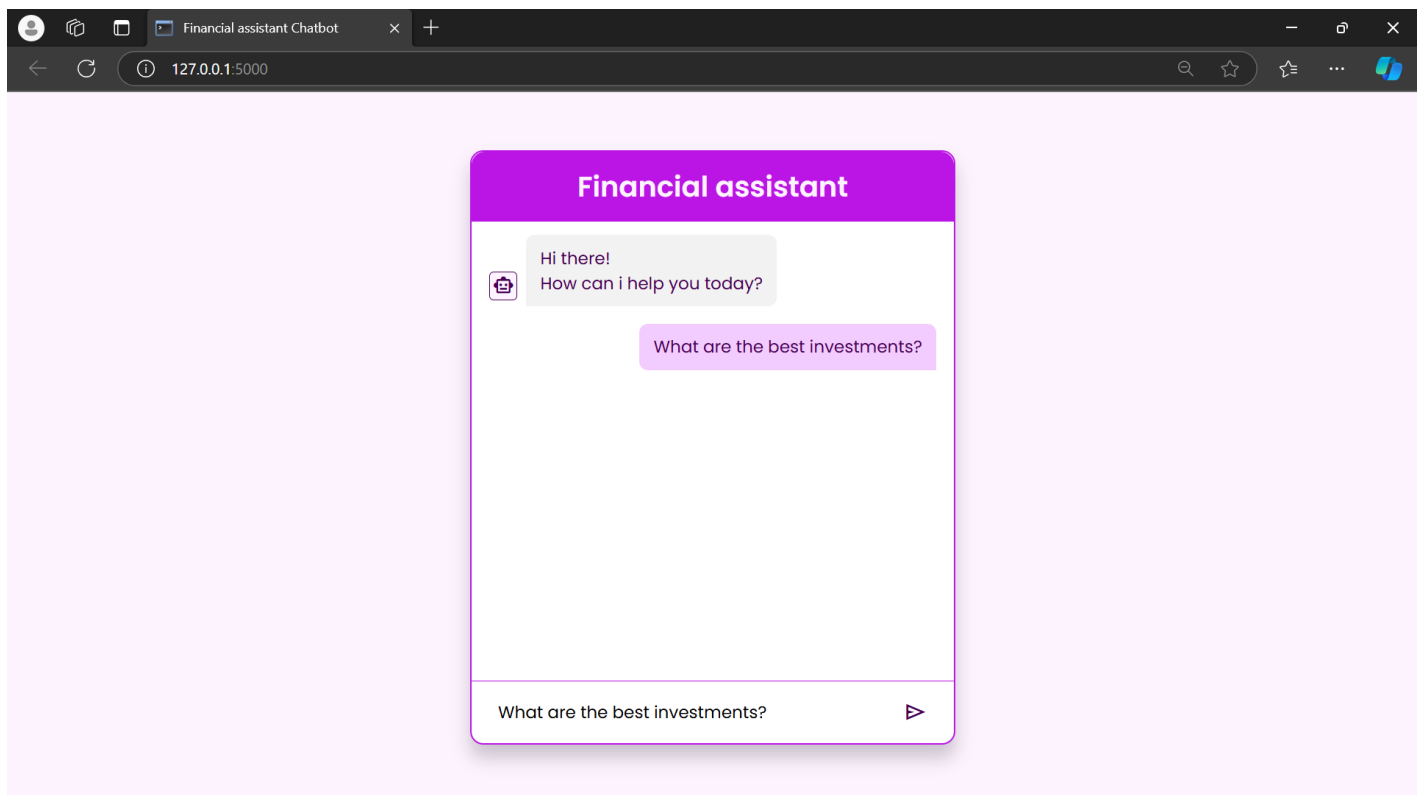
}
const handleChat = () => {
  userMsg = chatInput.value.trim();
  console.log(userMsg);

  if(!userMsg){
    return;
  }
  chatBox.appendChild(createChatLi(userMsg, "outgoing"));

  setTimeout(() => {
    generateResponse();
  }, 500);
}
sendChatBtn.addEventListener("click", handleChat);

```

OUTPUT



CONCLUSION

The financial assistant chatbot was successfully developed, providing real-time financial guidance and automating routine financial tasks to simplify personal finance management. It offers an accessible solution for users to make informed financial decisions, track expenses, and plan for future goals. For future enhancements, the project aims to expand the chatbot's capabilities to support more advanced financial scenarios, integrate with additional financial institutions for seamless data synchronization, and improve the user interface and user experience (UI/UX) for a more intuitive and engaging interaction.



REAL-TIME FINANCIAL ASSISTANT

**Mini-Project Presentation
by
Hariharan E (92132223048)
Hariharan M (92132223046)
Lokesh s (92132223080)**



Abstract

- The financial assistant chatbot provides an intelligent solution by offering real-time financial guidance and task automation through conversational AI.
- The assistant can connect to banking and financial platforms, allowing users to automate routine tasks such as bill payments, fund transfers, and investment portfolio management.
- The financial assistant chatbot has the potential to revolutionize financial management by making it more accessible and user-centric.
- the future of financial chatbots will see even more sophisticated, context-aware systems that empower users to achieve greater financial well-being and security.

Introduction

- The financial assistant chatbot offers a solution by delivering intelligent financial guidance and task automation through a conversational interface.
- The chatbot can interpret user input, provide tailored financial advice, and automate routine tasks such as bill payments, investment tracking, and account monitoring.
- Explore the development, capabilities, and potential applications of a financial assistant chatbot, emphasizing its impact on personal finance and the challenges of optimizing the system for accuracy, user engagement, and scalability.

Problem statement

- The complex financial landscape, managing personal finances can be challenging for individuals and businesses alike.
- There is a growing demand for a more accessible and intelligent financial management solution that can simplify these processes while providing personalized guidance.
- The system must handle a wide range of user queries, from basic financial questions to complex investment strategies, while maintaining low response times to facilitate seamless user interactions

Challenges

- **Accuracy:** Providing precise and personalized financial advice while considering individual user needs, financial goals, and the context of their financial situation.
- **Latency:** Delivering real-time responses to user queries with minimal delays, which is essential for a smooth conversational experience.
- **Natural Language Understanding:** Accurately interpreting diverse user inputs, including slang, abbreviations, or vague financial terms, and understanding context-specific queries.
- **Cross-Platform Support:** Ensuring the chatbot functions seamlessly across various platforms, including mobile apps, web interfaces, and voice-activated devices.

Tools & Technologies

Backend:

- Python, Flask for server-side handling.

Frontend:

- HTML, CSS for UI.
- JavaScript for interactivity.

Proposed Model

- The proposed solution is an AI-driven financial assistant chatbot designed to help users manage personal finances through real-time guidance and task automation.
- The chatbot will utilize advanced Natural Language Processing (NLP), Machine Learning (ML), and predictive analytics to deliver accurate, personalized financial advice.
- This will enable the system to offer more reliable advice across different financial contexts.

Implementation

```
File Edit Selection View Go Run ... Financial Assistant
```

EXPLORER

- FINANCIAL ASSISTANT
 - __pycache__
 - static
 - script.js
 - styles.css
 - templates
 - index.html
 - app.py
 - chatbot_model.h5
 - financial_ipynb
 - intents1.json
 - label_encoder.pkl
 - tokenizer.pkl
 - util.py

app.py > ...

```
1 import json
2 import numpy as np
3 import pandas as pd
4 import re
5 import random
6 from flask import Flask, render_template, request, jsonify
7 from tensorflow.keras.preprocessing.text import Tokenizer
8 from tensorflow.keras.preprocessing.sequence import pad_sequences
9 from sklearn.preprocessing import LabelEncoder
10 from tensorflow.keras.models import load_model
11
12 app = Flask(__name__)
13
14 # Load and preprocess the intents data
15 with open('intents1.json', 'r') as f:
16     data = json.load(f)
17
18 df = pd.DataFrame(data['intents'])
19
20 # Transform data into a more usable format
21 dic = {"tag": [], "patterns": [], "responses": []}
22 for i in range(len(df)):
```

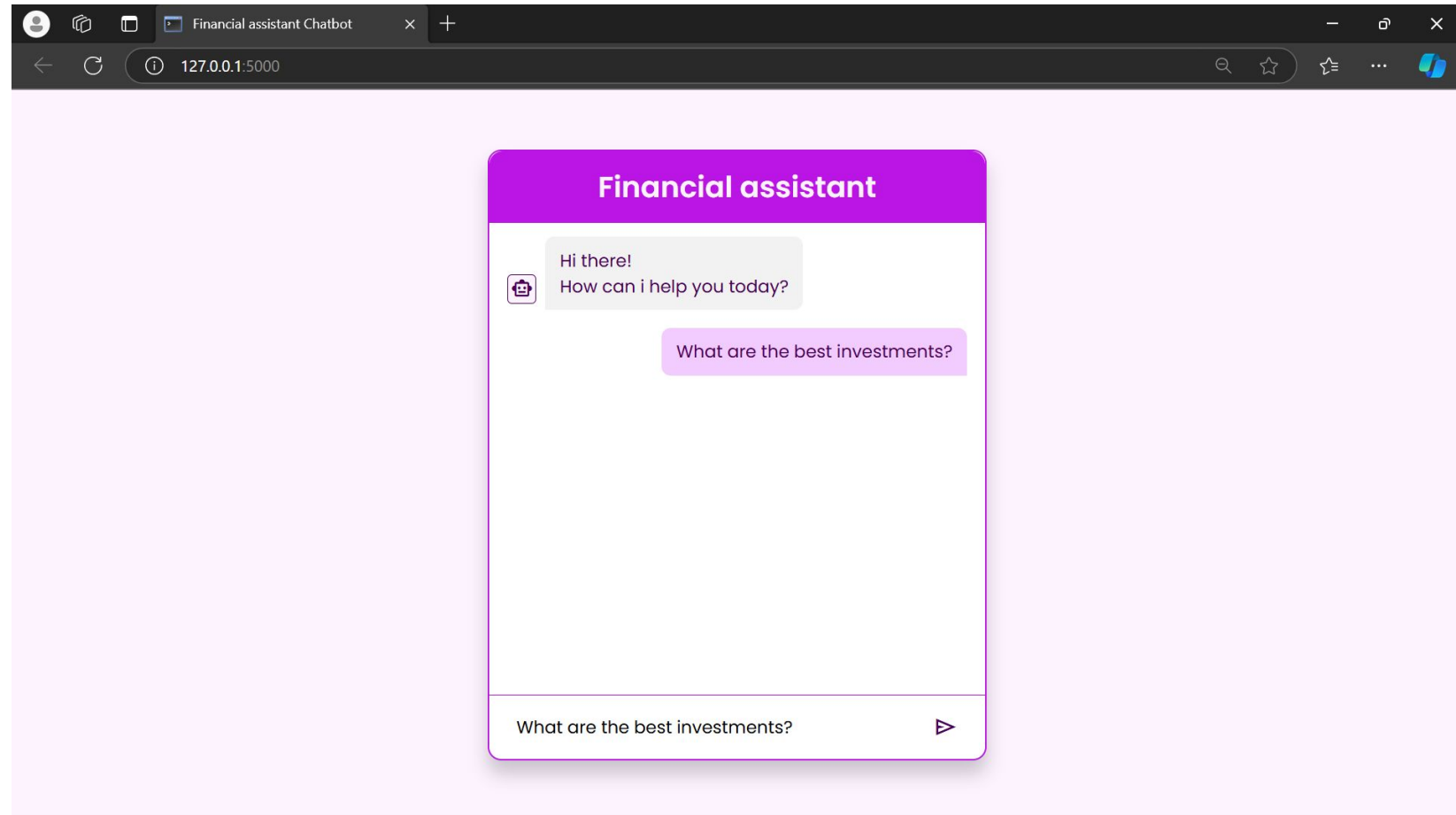
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SERIAL MONITOR

CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 AVX_VNNI FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
WARNING:abs1:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until you train or evaluate the model.
PS E:\hari\5th sem\TSA\Financial Assistant>
* History restored

PS E:\hari\5th sem\TSA\Financial Assistant> python app.py

Ln 91, Col 24 Spaces: 4 UTF-8 CRLF {} Python Select Interpreter Go Live Prettier

Demo screenshot



Conclusion

Summary:

- Successfully created a real-time financial assistant chatbot.
- It offers an accessible solution for users to make informed financial decisions

Future Improvements:

- Integrate with additional financial institutions for seamless data synchronization.
- Enhanced UI/UX.