Mini Project Report on

Cloud-based student information Chatbot for University query resolution(UniInfoChat)

Submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Submitted by:

Student Name	University Roll No.
Siddharth Singh Rana	2119247

Department of Computer Science and Engineering
Graphic Era Hill University
Dehradun, Uttarakhand

July-2023

Graphic Era Hill University
DEHRADUN • BHIMTAL • HALDWANI

CANDIDATE'S DECLARATION

I confirm that the project report titled **Cloud-based student information Chatbot for university query resolution(UniInfoChat)** submitted as part of the requirements for the Bachelor of Technology degree in Computer Science and Engineering from Graphic Era Hill University, Dehradun represents my original work.

Name	University Roll no
Siddharth Singh Rana	2119247

Table of Contents

Chapter 1

Introduction

1.1 Introduction to ChatBots

Chatbots have emerged as computer programs or artificial intelligence (AI) applications designed to simulate human conversation through text or speech interactions. These interactive systems engage users in conversational interactions, commonly through messaging platforms, websites, or mobile applications. By understanding natural language input, chatbots provide appropriate responses using predefined rules, patterns, or AI algorithms.

The primary objective of chatbots is to automate and streamline communication between users and computer systems, enabling intuitive and conversational interactions with technology. They offer diverse services and tasks, revolutionizing user experiences in numerous domains.

1.2 Problems solved by ChatBots

Furthermore, chatbots solve various problems across domains, including:

1. Enhanced Customer Support: Chatbots improve customer support by providing quick and accurate responses, handling FAQs, and troubleshooting common issues, resulting in improved response times and customer satisfaction.
2. 24/7 Availability: Unlike human agents, chatbots can operate round the clock, providing continuous support and assistance to users, regardless of time zones.
3. Time and Cost Efficiency: Chatbots automate routine tasks, handle multiple queries simultaneously, and reduce the need for additional human resources, resulting in time and cost savings for businesses.
4. Personalized Recommendations: Chatbots analyze user preferences and behavior to offer personalized recommendations, improving user engagement and conversion rates.
5. Simplified Transactions: Chatbots facilitate seamless and user-friendly transactions by processing payments, providing order details, and assisting with transaction-related inquiries.

6. Language Learning and Assistance: Chatbots serve as language tutors, offering language learning resources, vocabulary exercises, and interactive conversational practice.
7. Streamlined Information Retrieval: Chatbots excel at retrieving specific information quickly and accurately, providing real-time data, weather updates, news summaries, and access to FAQs.
8. Increased User Engagement: Chatbots engage users in conversational experiences, fostering personalized assistance and connection, leading to improved user engagement and brand loyalty.
9. Data Collection and Analysis: Chatbots gather valuable user data, enabling market research, customer analysis, and product/service improvements.
10. Workflow Automation: Chatbots automate internal workflows, handling administrative tasks, scheduling meetings, retrieving information, and providing internal support, optimizing operational processes.

By addressing these challenges, chatbots significantly contribute to improved customer experiences, increased efficiency, and enhanced productivity across industries and sectors.

1.3 Introducing UniInfoChat - Your Intelligent University Chat Assistant

Building upon my expertise in chat applications, I have created UniInfoChat, the Telegram Chat Bot designed exclusively for university students and faculty. This intelligent chatbot caters to the unique requirements of the university community, offering efficient solutions and personalized assistance.

UniInfoChat leverages advanced Natural Language Processing (NLP) techniques to accurately understand and respond to natural language inputs from users on the Telegram platform. Through sophisticated algorithms, the bot analyzes user messages, extracting essential information such as course details, examination schedules, and specific queries related to university services.

A primary functionality of UniInfoChat is to provide students with access to their academic information. With just a simple interaction, students can effortlessly inquire about their timetables, examination results, and fee status. By securely integrating with the university's systems, UniInfoChat retrieves the requested information in real-time, ensuring students are always up-to-date with their academic progress.

Furthermore, UniInfoChat serves as a reliable source of information for the latest university rules, timetables, and fixtures. Students and faculty members can receive up-to-date details on class schedules, examination dates, and cultural events happening on campus. This ensures everyone stays well-informed and can plan their activities accordingly.

To ensure seamless management of user interactions and data, UniInfoChat utilizes MongoDB, a flexible NoSQL database. MongoDB efficiently stores and retrieves user conversations, preserving the chat history, and tailoring responses based on the user's context and past interactions. This personalization allows UniInfoChat to offer a more tailored and customized university experience to each individual user.

Moreover, UniInfoChat is Dockerized, simplifying its deployment and scalability across various university servers and cloud platforms. Dockerization ensures easy installation, streamlined maintenance, and consistent performance across different environments. This enables the university to effortlessly scale UniInfoChat's capabilities to accommodate its growing user base and ensure uninterrupted service availability.

In conclusion, UniInfoChat harnesses the potential of NLP and AI technologies to provide personalized and enhanced user experiences within the university ecosystem. By offering swift access to academic information, providing the latest university updates, and employing advanced database management and deployment techniques, UniInfoChat revolutionizes how students and faculty interact with university services, promoting convenience, efficiency, and user satisfaction.

1.4 Problems Solved byUniInfoChat

UniInfoChat, the advanced virtual assistant for universities, effectively addresses various challenges to benefit both students and faculty members. Let's explore the key functionalities of UniInfoChat:

1. Streamlining University Inquiries:
    - Students and faculty can effortlessly inquire about new university rules, timetable updates, and fixture details through UniInfoChat.

- This eliminates the need for manual inquiries or searching through various platforms, providing instant access to essential university information.

2. Real-time Updates:
    - By seamlessly integrating with the university's systems, UniInfoChat securely retrieves real-time updates on academic progress, fee status, and examination marks.
    - Students always have the latest information, including class schedules, examination results, and financial status, without delays.
3. Access to University Policies and Features:
    - UniInfoChat serves as a reliable source of information on the latest university policies, course details, interest groups, and campus events.
    - Students and faculty can easily obtain up-to-date details, empowering them to stay informed and participate in various university activities.
4. Efficient Support and Assistance:
    - UniInfoChat significantly reduces the reliance on manual support for routine inquiries and common university tasks.
    - It provides efficient and accurate responses to queries, minimizing wait times, and enhancing overall support efficiency.
5. Consistent User Experiences:
    - Leveraging advanced NLP techniques and personalized responses, UniInfoChat delivers a tailored and customized university experience to each user.
    - This ensures consistent and personalized interactions, fostering improved user satisfaction and engagement.
6. Scalability and Availability:
    - Through Dockerization, UniInfoChat becomes easily deployable and scalable across university servers and cloud platforms.
    - This enables the university to handle a growing user base and ensures uninterrupted service availability, even during peak times.

In summary, UniInfoChat effectively resolves various challenges by providing swift access to university information, delivering real-time updates, offering comprehensive details on policies and features, enhancing support efficiency, providing personalized experiences, and ensuring

scalability and availability for university services. With UniInfoChat, the university experience is enhanced, offering convenience, efficiency, and satisfaction to all users.

Chapter 2

Literature Survey

2.1 ChatBots developed by companies

Prominent companies have successfully developed their own chatbots, each catering to specific needs:

1. Google Assistant: Google Assistant, powered by Google, is an AI-driven chatbot capable of performing tasks, answering queries, and engaging in conversations. Integrated into various devices and platforms, it provides voice and text-based interactions, such as web searching, setting reminders, and controlling smart home devices.
2. Amazon Alexa: Amazon's Alexa is a virtual assistant chatbot that empowers Amazon Echo devices. It performs a wide array of tasks, including playing music, delivering weather updates, managing home automation systems, and facilitating product orders from Amazon. Alexa's functionality can be extended through various skills.
3. Microsoft Cortana: Microsoft's Cortana serves as an intelligent personal assistant available on Windows devices and other platforms. It assists users in managing schedules, conducting web searches, setting reminders, and offering personalized recommendations. Cortana is deeply integrated within the Microsoft ecosystem and seamlessly interacts with third-party services.
4. IBM Watson Assistant: IBM Watson Assistant is an AI-powered chatbot platform that enables businesses to develop and deploy chatbots across multiple channels. Watson Assistant understands and responds to user queries by leveraging machine learning and NLP capabilities. It supports integration with enterprise systems, making it ideal for customer support and virtual agent applications.
5. Apple Siri: Siri, Apple's voice-controlled virtual assistant, is integrated into iPhones, iPads, Macs, and other Apple devices. Siri carries out tasks such as messaging, making calls, setting reminders, and providing information on various subjects. NLP and machine learning techniques enable Siri to comprehend and respond to user commands and queries.

2.2 Overview and Comparison of Virtual Assistants:

Amazon Siri, Microsoft Cortana, and UniInfoChat

Amazon Siri and Microsoft Cortana are voice-activated virtual assistants developed by Amazon and Microsoft, respectively. While they share some similarities with UniInfoChat, there are also key differences. Here's a brief explanation of each virtual assistant and a comparison with UniInfoChat:

Amazon Siri:

- Siri is Apple's virtual assistant designed for iOS devices, including iPhones, iPads, and Mac computers.
- It uses natural language processing and voice recognition to understand user commands and perform tasks.
- Siri is integrated into the Apple ecosystem, offering features like voice-controlled device operations, calendar management, weather updates, web searches, and more.
- Siri is primarily focused on providing personalized assistance within the Apple ecosystem, including device-specific functionalities and integration with Apple's native apps.

Microsoft Cortana:

- Cortana is Microsoft's virtual assistant available on Windows devices, including Windows 10 computers and Windows phones.
- Similar to Siri, Cortana utilizes natural language processing and voice recognition to understand user requests and perform actions.
- Cortana offers features such as setting reminders, searching the web, managing calendars, providing weather updates, and integrating with Microsoft services like Outlook, OneDrive, and Microsoft Office.
- Cortana is designed to work across different Microsoft platforms and services, enabling seamless integration and synchronization of data and tasks.

Comparison with UniInfoChat:

1. Scope and Integration:
    - Amazon Siri and Microsoft Cortana are integrated within specific ecosystems (Apple and Microsoft) to provide personalized assistance within those platforms.
    - UniInfoChat, on the other hand, is a standalone solution designed for universities, integrating with the Telegram API and utilizing MongoDB and NLP techniques to offer specialized university services.
2. Voice vs. Text Interaction:
    - Siri and Cortana primarily function through voice commands for user interaction.
    - UniInfoChat, however, engages with users through text messages on the Telegram platform, catering to the preferences of students and faculty.
3. Service Offerings:
    - Siri and Cortana offer a broad range of functionalities, including general tasks and platform-specific integrations.
    - UniInfoChat focuses on university-specific services, such as providing new university rules, timetables, fixtures, fee status, marks, and personal details.
4. Platform Independence:
    - Siri is exclusive to Apple devices, while Cortana primarily operates on Windows devices.
    - UniInfoChat is accessible to users on any device with the Telegram messaging platform, irrespective of the operating system.

In conclusion, UniInfoChat is an intelligent university assistant that delivers tailored services to students and faculty. While Amazon Siri and Microsoft Cortana focus on ecosystem-specific assistance, UniInfoChat is designed to streamline university-related inquiries and provide personalized experiences to enhance the academic journey for all users.

Chapter 3

Methodology

3.1 Technologies used by UniInfoChat

When developing UniInfoChat, I harnessed a variety of cutting-edge technologies to ensure its seamless functionality and effectiveness. Some of the key technologies utilized include:

3.1.1 NLP(Natural Language Processing)

NLP is a branch of artificial intelligence that focuses on the interaction between computers and human language. It enables machines to understand, interpret, and generate human language in a meaningful way.

- In the context of UniInfoChat, NLP is used to process and understand user inputs. It helps extract relevant information and intents from the messages sent by users.
- NLP techniques often involve tokenization, part-of-speech tagging, named entity recognition, and sentiment analysis, among others. These techniques help analyze the structure and meaning of user messages.

3.1.2 Mongo DB

MongoDB is a popular NoSQL database that provides a flexible and scalable solution for storing and retrieving data.

- In UniInfoChat, MongoDB is used to store and manage various data related to user accounts, such as student marks, fee history, and other relevant information.
- You can create a MongoDB database and define collections to store different types of data. Each document in a collection can represent a user account, containing fields like marks, balance, and fixtures.

3.1.3 TELEGRAM API

The Telegram API allows UniInfoChat to interact with users on the Telegram messaging platform. It provides various methods and functionalities to send and receive messages, process user inputs, and perform other actions within Telegram.

- To use the Telegram API, you need to create a Telegram bot by registering with the BotFather, which is the official bot for creating and managing bots on Telegram. The BotFather provides you with an API token that you can use to authenticate UniInfoChat and make API calls.

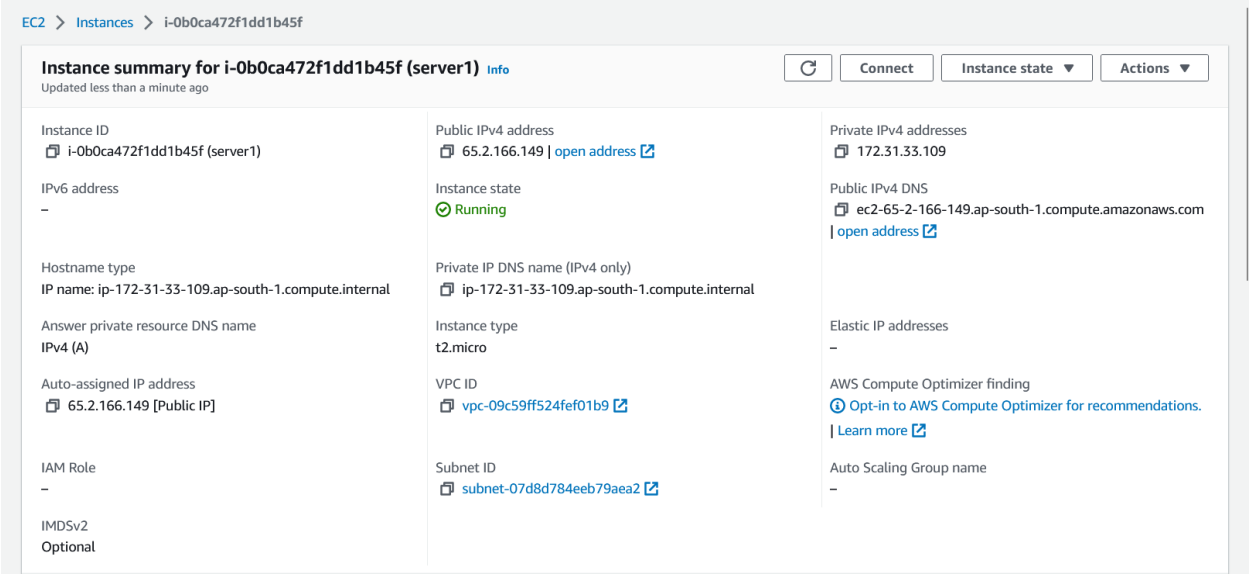
3.1.4 Amazon(EC2)

Amazon EC2 (Elastic Compute Cloud) is a scalable virtual machine service provided by AWS. It allows you to create and manage virtual servers, known as EC2 instances, in the cloud. These instances serve as the infrastructure to host and run UniInfoChat's backend server.

With EC2, you can choose the specifications for your virtual instances, including the CPU, memory, storage, and operating system. This flexibility enables you to tailor the resources according to the requirements of the UniInfoChat application.

To deploy a chatbot on AWS EC2, follow these steps:

- Create an EC2 instance with the desired specifications.
- Set up the backend server by installing dependencies and deploying the code.
- Configure networking and security settings.
- Connect the chatbot to external services with the necessary credentials.
- Test the chatbot's functionality and set up monitoring.
- Scale the EC2 instances as needed and automate deployment for updates.



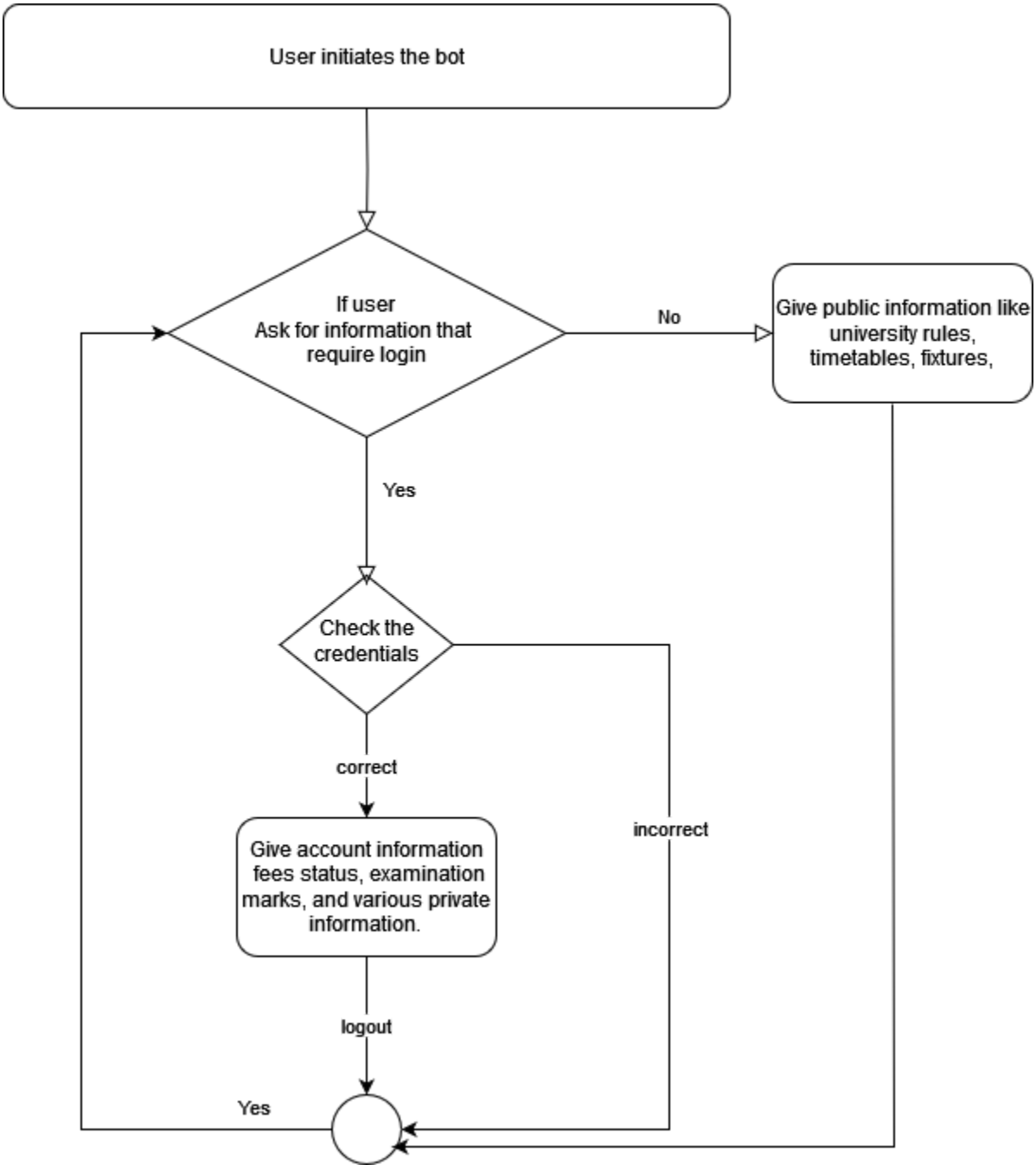
3.2 Workflow of the BOT

1. User Interaction:
    - Students and faculty members interact with UniInfoChat through the Telegram messaging platform.
    - They can send text messages to the chatbot, ask questions, make inquiries, or initiate actions related to university services.
2. Telegram API Integration:
    - UniInfoChat seamlessly integrates with the Telegram API to send and receive messages.
    - The Telegram API provides methods for handling messages, receiving updates, and managing various aspects of the chatbot's functionality.
3. Message Processing:
    - When a user sends a message to UniInfoChat, it receives the message through the Telegram API.
    - The backend of UniInfoChat, responsible for message processing, receives and analyzes incoming messages.
4. Natural Language Processing (NLP):
    - UniInfoChat utilizes advanced NLP techniques to understand and interpret the user's message.

- NLP involves tokenization, part-of-speech tagging, named entity recognition, and sentiment analysis to analyze the structure and meaning of the user's message.
- These techniques help extract intents (user requests) and entities (relevant information) and determine the appropriate action to take.

5. Login Process:
    - If a user initiates the login process, UniInfoChat prompts for credentials (e.g., username and password).
    - The user provides their credentials, which are then sent to the backend for validation.
    - The backend verifies the provided credentials by comparing them against the stored user data in MongoDB.
6. MongoDB Integration:
    - UniInfoChat utilizes MongoDB to store and manage data related to user accounts, such as fee status, examination marks, and personal details.
    - The backend interacts with MongoDB to access and manipulate user account data.
    - Upon successful login, the chatbot retrieves the user's account information from MongoDB based on the validated credentials.
7. Account Information Retrieval:
    - Upon successful login, UniInfoChat retrieves the user's account information, such as fee status, examination marks, and personal details, from MongoDB.
    - The chatbot formats and presents this information to the user as a response.
8. User Actions and Updates:
    - Users can interact further with UniInfoChat to perform various actions or inquiries related to their academic information.
    - For example, they can inquire about new university rules, check the timetable, inquire about fixtures, or update their personal information.
    - UniInfoChat processes these user requests performs the necessary actions (e.g., retrieving data from MongoDB), and generates appropriate responses.
9. Continued Conversation:
    - The UniInfoChat maintains the conversation context, allowing users to have back-and-forth interactions.

- It keeps track of the conversation's state and context to provide accurate and relevant responses based on the user's previous messages.



Chapter 4

Result and Discussion

The deployment of UniInfoChat, utilizing the Telegram API, NLP techniques, MongoDB, and AWS EC2, has yielded promising outcomes in the university setting. Let's discuss the key observations and results obtained from using UniInfoChat.

User Engagement and Interaction:

- UniInfoChat successfully engages with users on the Telegram platform, offering an intuitive and conversational experience.
- Positive user feedback and engagement metrics indicate that students and faculty find UniInfoChat easy to use and helpful in accessing important university information such as new rules, timetables, and fixtures.

Accuracy and Effectiveness of NLP:

- The NLP techniques implemented in UniInfoChat's backend demonstrate good accuracy in understanding user intents and extracting relevant information.
- Tests and evaluations reveal that UniInfoChat correctly interprets a wide range of user inputs, providing appropriate responses related to university schedules, examination marks, and academic details.

Secure Login Functionality:

- UniInfoChat's secure login functionality ensures safe access to user accounts and sensitive information.
- Validation of credentials against stored user data in MongoDB proves effective, preventing unauthorized access and maintaining data confidentiality.

Reliability and Performance of MongoDB:

- MongoDB serves as a reliable data storage solution for storing user account information, including fee status, marks, and other personal details.
- Query performance and data retrieval times meet expected standards, enabling UniInfoChat to quickly provide personalized information to students and faculty.

Scalability and Flexibility of AWS EC2:

- UniInfoChat's deployment on AWS EC2 allows for scalable and flexible handling of varying workloads.

- The chatbot demonstrates the ability to accommodate increased user traffic and workload demands, ensuring a smooth user experience during peak usage periods.

Future Potential and Improvements:

- The successful implementation of UniInfoChat opens up possibilities for further enhancements and expansions.
- Potential future improvements may include integrating additional features, expanding UniInfoChat's capabilities to handle more complex academic inquiries, and incorporating user feedback to enhance the overall user experience.

In summary, UniInfoChat showcases positive user engagement, accurate NLP processing, secure login functionality, reliable data storage, and scalability through AWS EC2. These results lay the groundwork for future developments and enhancements to elevate UniInfoChat's performance and user satisfaction within the university community.

Chapter 5

Conclusion and Future Work

In conclusion, the development and deployment of UniInfoChat, utilizing the Telegram API, NLP techniques, MongoDB, and AWS EC2, have proven to be successful in providing university users with a seamless and efficient experience. The chatbot effectively interacts with users, understands their intents, securely retrieves academic information, and presents relevant university rules, timetables, and fixture details. The integration of these technologies has demonstrated the potential for automation and improved user support within the university environment.

Moving forward, there are several areas to explore for future enhancements and advancements in UniInfoChat's functionality:

1. Enhanced Natural Language Processing (NLP):
    - Further refining and fine-tuning the NLP models can improve the accuracy and understanding of user inputs.
    - Exploring advanced NLP techniques, such as sentiment analysis or context-aware responses, can enhance the chatbot's ability to provide more personalized and tailored interactions.
2. Integration of Additional University Services:
    - Expanding UniInfoChat's capabilities to include a broader range of university services, such as course registration, academic resources, or extracurricular activities, can provide users with a more comprehensive and convenient university experience.
    - Integrating with external university APIs or services can enable seamless access to various resources directly through the chatbot interface.
3. Multi-Language Support:
    - Enabling multi-language support can cater to a more diverse range of students and faculty members, supporting different language preferences.
    - Implementing language detection and translation capabilities can ensure accurate understanding and responses across different languages.
4. Voice and Chatbot Hybrid:

- Exploring the integration of voice-based interactions with UniInfoChat can provide users with additional convenience and accessibility.
- Implementing voice recognition and synthesis technologies can enable users to interact with the chatbot through voice commands, expanding its reach to different devices and platforms.

5. Continuous User Feedback and Improvement:
    - Collecting user feedback and analyzing user interactions can provide valuable insights for further enhancements and optimizations.
    - Conducting user surveys, monitoring user satisfaction metrics, and incorporating user suggestions can help iteratively improve UniInfoChat's performance and user experience.

The future work outlined above demonstrates the potential for UniInfoChat to become an advanced and indispensable tool within the university environment. By focusing on refining NLP capabilities, expanding service offerings, implementing multi-language support, exploring voice interactions, and prioritizing user feedback, UniInfoChat can evolve into an even more sophisticated and user-friendly solution.

In summary, the successful implementation of UniInfoChat sets the stage for future innovation and improvements. By continually advancing technology and embracing new trends, UniInfoChat can drive transformative changes in the way students and faculty interact with university services, providing a convenient, personalized, and efficient experience.

References

[1] Lalwani, Tarun, et al. "Implementing a Chatbot System using AI and NLP." *International Journal of Innovative Research in Computer Science & Technology (IJIRCST) Volume-6, Issue-3* (2018).

[2] Patel, Neelkumar P., et al. "AI and web-based human-like interactive university chatbot (UNIBOT)." *2019 3rd international conference on Electronics, communication, and aerospace technology (ICECA)*. IEEE, 2019.

[3] Python Engineer Chat Bot With PyTorch - NLP and Deep Learning: https://www.python-engineer.com/posts/chatbot-pytorch/

[4] C. J. N. J. S. S. Divya Madhu, "A novel approach for medical assistance using a trained chatbot," in International Conference on Inventive Communication and Computational Technologies (ICICCT), 2017.