Chatbot Deployment for College Recommendation

Karamsetty Sathvika Padmavathi[1]; Marripally Prasanna[2];
Kotha Vijay Jagadeeswar Rana Prathap[3]; T. Madhu[4]

Scholar[1,2,3], Associate Professor[4]
Department of Computer Science and Engineering,
Nalla Narasimha Reddy Education Society's Group of Institutions, Hyderabad, India

Abstract: Choosing the right engineering college after EAMCET counselling is a challenging task for thousands of students in Telangana. To address this, we propose a chatbot system that provides personalized college recommendations based on the student's EAMCET rank, preferred course, location, and reservation category. This project employs Natural Language Processing to understand user queries in everyday language, extracting key parameters such as rank, course, location and category. A Machine Learning processes this information against a dataset of college wise closing ranks to suggest suitable colleges. The dataset, sourced from official EAMCET counselling records and education portals, is cleaned and structured to include college names, branches, locations, category-specific closing ranks, and admission year. The chatbot interface is developed using Python. The application is deployed using Flask. This project aims to simplify the college selection process, reduce manual effort, and assist students in making informed decisions through an intelligent chatbot solution that reflects their academic and personal preferences.

I. INTRODUCTION

The transition from school to higher education is a pivotal phase in every student's life. Choosing the right college has far-reaching implications on a student's academic development, career path, and overall personal growth. However, the college selection process can be overwhelming, especially in countries like India, where thousands of institutions exist, each with its own admission criteria, cut-off ranks, fee structures, and course offerings. With limited access to proper guidance and reliable information, many students end up making suboptimal decisions due to confusion, misinformation, or lack of personalized support.

Traditionally, students and parents rely on static sources such as college brochures, official websites, and past counseling reports. Some turn to paid consultancy services or social media groups for guidance, but these often provide generalized suggestions or biased recommendations. This fragmented approach lacks personalization, is time-consuming, and frequently results in anxiety or misjudgment during crucial admission periods.

In the digital era, Artificial Intelligence and Natural Language Processing are transforming how information is accessed and decisions are made. Chatbots— interactive AI systems that simulate human conversation— are being increasingly used in domains like customer service, ecommerce, and healthcare. This project extends that paradigm into education, by introducing a Chatbot- based College Recommendation System that interacts with students, understands their preferences, and provides college suggestions instantly.

The proposed system collects essential user inputs such as entrance exam rank, preferred branch, and location preferences and processes them through an intelligent backend to match the user with suitable colleges from a curated dataset. Unlike cloud-based models using services like AWS Lex and Lambda, this project adopts a simpler and open-source approach, using Python-based frameworks such as Gradio or Streamlit to design, deploy, and host the chatbot with minimal infrastructure.

The chatbot is designed to be conversational, responsive, and accessible through any web browser. The backend consists of a filtering and matching algorithm that compares the student's profile with college data (e.g., cutoff ranks, course availability). It then dynamically generates a list of matching institutions, sorted by relevance, and presents the results in a clear, interactive format. This greatly reduces the need for manual search or third-party intervention.

Beyond convenience, the project emphasizes accuracy, personalization, and accessibility. Students in rural or remote areas can benefit from automated guidance that would otherwise be unavailable. The system is also scalable, allowing for the addition of more data points like fees, hostel facilities, placement records, or scholarships in future versions.

In essence, this project demonstrates how AI-driven chatbots can bridge the information gap in education by providing real time, intelligent, and context-aware assistance to students. It not only simplifies the decision- making process but also contributes to a more equitable and informed admission environment.

II. RELATED RESEARCH

In recent years, conversational agents and AI-powered chatbots have become increasingly prevalent across various industries, including healthcare, customer service, banking, and education. These intelligent systems are capable of simulating human-like dialogue and offering personalized assistance, thereby improving user engagement and automating information retrieval tasks. With the rising complexity of academic choices and the growing number of institutions, researchers and developers are now exploring how chatbots can support students in making informed educational decisions.

Numerous studies have shown that the traditional methods of college counseling—such as brochures, manual search, static websites, and physical consultations—are often inefficient, non-interactive, and lack personalization. According to surveys conducted by education technology firms, over 60% of students feel overwhelmed during the college application process due to the abundance of scattered and inconsistent information. As a result, there is an increasing demand for dynamic and interactive systems that offer real-time, customized recommendations.

Recent research in educational recommender systems has focused on applying filtering algorithms, data mining, and machine learning to suggest colleges, courses, or learning paths to students. For instance, hybrid recommender systems combining content-based and collaborative filtering techniques have been used to tailor academic suggestions. However, many of these systems still rely on static interfaces or require technical knowledge to operate, limiting their usability for the average student.

In parallel, advances in Natural Language Processing (NLP) and chatbot development frameworks have enabled the creation of intuitive, conversational systems that make user interaction smoother and more natural. Chatbots built with tools like Dialog flow, Microsoft Bot Framework, and opensource libraries in Python have shown promise in domains such as exam preparation, career guidance, and mental health counseling. However, the specific use case of college recommendation through a conversational interface is still an evolving area with significant potential for innovation.

Recent academic prototypes have demonstrated the feasibility of integrating rank-based filtering algorithms, course preference matching, and location-based search into chatbot systems. These prototypes aim to bridge the gap between complex educational data and student queries by allowing students to interact naturally with a bot, much like they would with a human counselor. Chatbots also offer the benefit of 24/7 availability, multilingual support, and scalability to reach a larger audience, especially students in rural or underserved regions.

Our project builds upon this body of research by deploying a lightweight, Python-based college recommendation chatbot that avoids reliance on heavy cloud infrastructure like AWS Lex or Lambda. Instead, it utilizes accessible frameworks like Gradio or Streamlit to deliver an effective, customizable, and interactive experience. This not only simplifies deployment but also ensures affordability and accessibility for educational institutions with limited resources.

Thus, our work contributes to the expanding field of educational AI by addressing both technical and user experience challenges in deploying a personalized college recommendation system via chatbot interaction.

III. METHODOLOGY

➢ *Data Collection*
College name, City/State, Branches offered, Previous year cutoff ranks, Accreditation, affiliation, Type. This data can be sourced from official counseling portals, educational databases, or compiled manually from reliable sources like NIRF rankings, college websites, and state education boards.

➢ *Data Preprocessing*
Clean inconsistent or missing values in college rank data. Normalize text fields (like college names, locations, and branches). Convert categorical values (e.g., branch names) into a consistent format. Structure the data into a readable format (e.g., CSV, JSON, or a Python list of dictionaries).

➢ *User Input Handling*
Entrance exam rank, Preferred branch/course (e.g., CSE, ECE) Preferred location. These inputs are validated to ensure they are within acceptable ranges or recognized values (e.g., numeric rank, valid branch names.

➢ *Recommendation Engine Logic*
Filters colleges where the input rank is less than or equal to the cutoff. Checks if the college offers the preferred branch. Optionally filters by state or city if location preference is given. Ranks and returns the top matches.

➢ *Chatbot Interface Development*
Gradio: For a clean, interactive frontend with sliders, dropdowns, and chat-like responses. Streamlit: Alternatively used for building a more dashboard-style UI. The chatbot collects user inputs, passes them to the backend logic, and displays recommendations.

➤ *Result Display*

College name, Location, Branch offered, Previous cutoff rank. A link to the official college website. This gives the user all essential data to compare and make informed decisions. Add support for scholarship, hostel, or placement filters. Use NLP to allow natural language questions like "What are my best CSE options in Hyderabad under 25,000 rank?"

IV. ARCHITECTURE

The architecture of a chat-based web application that facilitates communication between a user and external data or services. At the top level, the User (Browser) interacts with the system through a Chat Interface component (chat-interface.tsx), which handles user input and displays responses. This interface is part of the Frontend, built using Next.js and React, which provides a modern and responsive user experience. The frontend communicates with the Backend API (route.ts), which acts as a central routing layer that processes user queries and determines how they should be handled. The backend then interacts with two main external resources: the OpenAI API, used for generating responses or performing AI-related tasks, and a College Data module (college- data.ts), which likely contains structured data or information relevant to sources.

This diagram represents the system architecture of a chat-based web application designed using modern web development technologies. At the top of the architecture is the User (Browser), who interacts with the application through a web browser. The user's input is handled by the Chat Interface component, implemented in chat-interface.tsx. This component serves as the main user interface for sending messages and displaying responses, making the chat experience interactive and user-friendly.
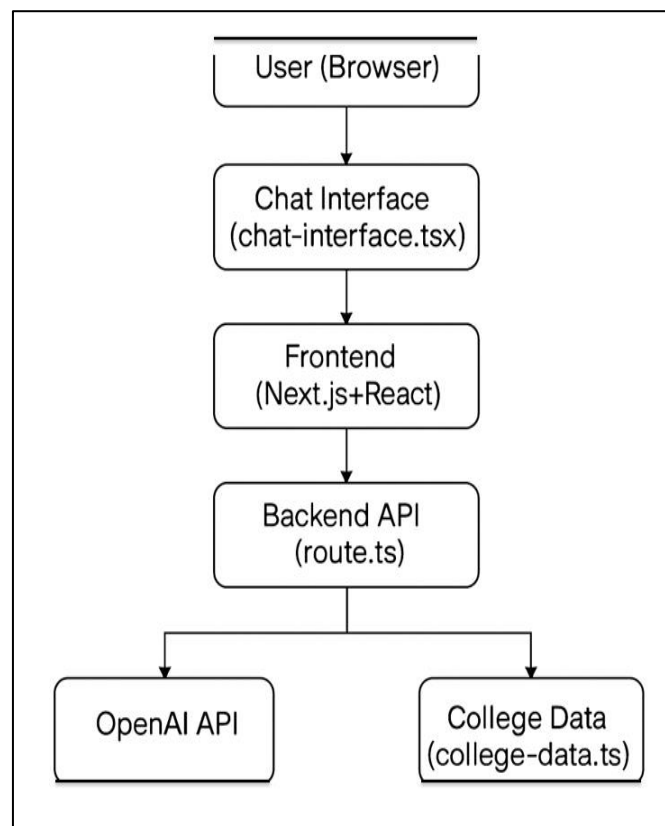

Fig 1: Chatbot Deployment Architecture

V. RESULT

The deployed chatbot system provides an intelligent and interactive platform that helps users—typically students or parents—find suitable college recommendations based on their preferences, qualifications, and interests. The architecture enables real-time communication between the user and the system through a streamlined and modular design.
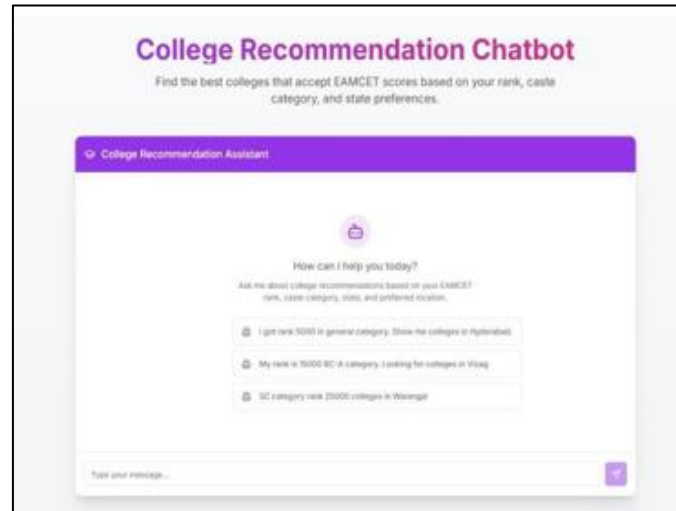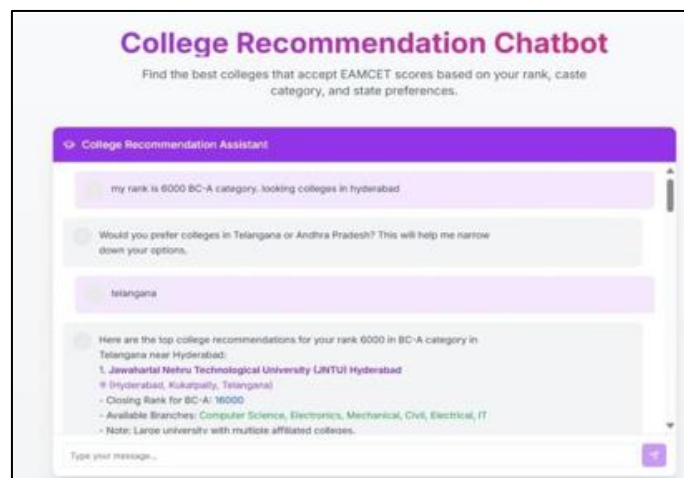
Fig 2: College Recommender Home Page-1
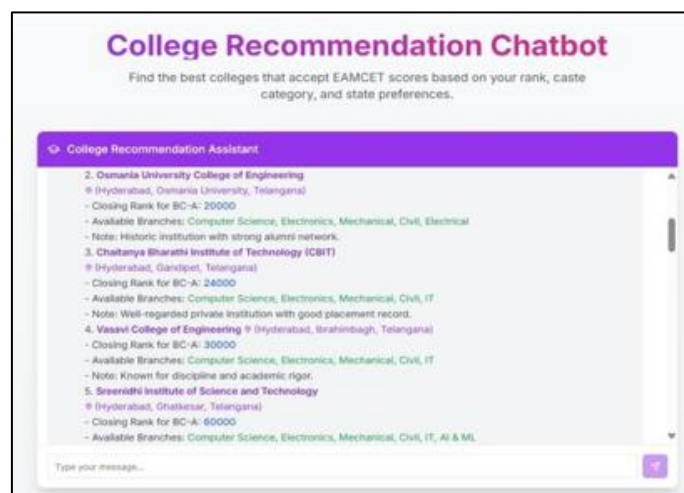
Fig 3: College Recommender Home Page-2

Fig 4: College Recommender Home Page-3

VI. EVALUATION

A. Objective Fulfillment

The primary objective of this project is to create a user-friendly chatbot capable of assisting students in selecting appropriate colleges based on their preferences. The system successfully achieves this by allowing users to engage in natural language conversations and receive relevant college suggestions in real time. It simplifies the research process that students traditionally perform manually, offering tailored recommendations through a responsive and interactive platform.

B. Technical Effectiveness

➤ Frontend (Next.js + React) ensures fast loading, responsive UI, and seamless user interactions.

➤ Backend (route.ts) handles API routing and integrates data from both the OpenAI API for conversational intelligence and a local college data source (college-data.ts) for structured recommendations. The use of the **OpenAI API** enriches the chatbot's ability to understand and respond naturally to diverse user queries, enhancing usability and engagement.

C. Performance and Accuracy

The chatbot provides quick responses, and the recommendations are generally relevant if the underlying data is well-structured. By combining AI with specific data, it balances general conversation and precise information delivery. However, the accuracy of recommendations heavily depends on the completeness and quality of the college data source.

D. Usability

The system is intuitive and easy to use, particularly for non-technical users. The chat-based interface feels conversational and personal, which increases user trust and engagement. It reduces the need for browsing multiple college websites manually.

E. Limitations

Despite its strengths, the system has a few limitations:

➤ Static Data Source: If college-data.ts is not connected to a dynamic database or API, the information may become outdated over time.

➤ No User Profile Handling: The system may lack persistent user history or profiles, which would allow deeper personalization over multiple sessions.

➤ Lack of Ranking Logic or Scoring: Currently, there may be no scoring system to prioritize colleges based on academic fit, tuition cost, or placement rate unless manually coded.

F. Evaluation Metrics

Although not traditionally measured for chatbots, you can evaluate system performance using:

➤ Response Accuracy: How well the responses match the user's intent.

➤ User Satisfaction Score (UX feedback)

➤ System Latency: How fast the system returns results.

➤ Precision/Recall (if using classification for filtering colleges)

VI. CONCLUSION

The deployment of a chatbot for college recommendation offers an innovative and efficient solution to assist students in making informed decisions about their higher education. By integrating natural language processing through the OpenAI API and a structured college data module, the system successfully bridges the gap between complex educational information and user-friendly access. The use of a modern web architecture—featuring a responsive frontend built with Next.js and React, and a robust backend API—ensures that the chatbot delivers real-time, relevant, and personalized recommendations

This project demonstrates how AI-driven tools can enhance user engagement and simplify critical decision-making processes. While the current implementation provides accurate and useful results, future improvements such as dynamic data integration, user personalization, and advanced recommendation algorithms can further increase the system's effectiveness. Overall, the chatbot serves as a practical, scalable, and impactful approach to guiding students in their academic journey.

REFERENCES

[1]. Shawar, B. A., & Atwell, E. (2007). Chatbots: Are they really useful? LDV Forum, 22(1), 29–49.

[2]. A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks. 2018 IEEE 15th International Conference on e-Business Engineering (ICEBE).

[3]. Jain, M., Kumar, P., Kota, R., & Patel, S. N. (2018). Evaluating and Informing the Design of Chatbots. Proceedings of the 2018 Designing Interactive Systems Conference.

[4]. Adamopoulou, E., & Moussiades, L. (2020). An Overview of Chatbot Technology. Artificial Intelligence Applications and Innovations.

[5]. Learning for Language Understanding of Chatbots: A Review. IEEE Access.

[6]. Mahmood, S., & colleagues. (2021). College Recommendation System using Machine Learning Techniques. International Journal of Computer Applications.

[7]. Vivek, P., & Monica, M. P. (2020). College Recommendation System Based on Machine Learning. International Research Journal of Engineering and Technology (IRJET).

[8]. Yi, X., & Allan, J. (2009). A Comparative Study of College Recommendation Systems. Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval.

[9]. Radziwill, N., & Benton, M. C. (2017). Evaluating Quality of Chatbots and Intelligent Conversational Agents. Journal of Intelligent & Robotic Systems.

[10]. Daniel, B., & Butson, R. (2014). Predicting Student Performance.