International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Intelligent Student Support System Using Multilingual Conversational AI

Prof. Tanaya Bala Behera[1], Chandrakanth[2], Chetan[3], Chinnirathod Gandhikumar[4], Darshan B[5]

1st Prof. Tanaya Bala Behera
dept. *Computer Science &Engineering*
Dayananda Sagar University
Bengaluru, India
tanya-cse@dsu.edu.in

2nd Chandrakanth
dept. Computer Science & Engineering
Dayananda Sagar University
Bengaluru, India
chandrakantgovindingle@gmail.com

3rd Chetan
dept. Computer Science & Engineering
Dayananda Sagar University
Bengaluru, India
chetanghorwade@gmail.com

4th Chetan
dept. Computer Science & Engineering
Dayananda Sagar University
Bengaluru, India
gandhikumarrathod@gmail.com

5th Darshan B
dept. Computer Science & Engineering
Dayananda Sagar University
Bengaluru, India
darshandarshanbb1@gmail.com

Abstract - In general, in most colleges, students and their parents face difficulty in getting proper information regarding admission, exams, fees, placement, etc. Though most of the colleges are providing the information in their website, it is quite time-consuming to search for the particular information. In order to overcome this difficulty, the present project proposes the development of an "Intelligent Student Support System Using a Multilingual Conversational Chatbot." This system will help to provide instant responses to the queries of the students in an automated way. The chatbot is created using Python programming with the help of the Flask framework. Natural Language Processing techniques are used to understand the queries of the students. According to the intent of the query, the system will fetch the information from the structured knowledge base and will respond appropriately to the students. Although the chatbot is currently designed to accept only text-based queries, it is possible to add voice-based queries in the near future to increase the accessibility of the system. In addition, the queries are kept in the database to further improve the system in the future. The solution provided here reduces the workload of the administration staff, saves time for the end-users, and also helps in better communication between the students and the institution. Thus, the system provided here is a simple, efficient, and effective way to automate the college enquiry system.

***Key Words*:** Artificial Intelligence, Natural Language Processing (NLP), Chatbot, College Enquiry System, Multilingual Support, Conversational AI, Flask Framework, Machine Learning

1.INTRODUCTION

In the contemporary digital world, educational institutions have begun to adopt the use of technology to aid communication and service delivery to their students and parents. Generally, colleges publish information on their official websites concerning admissions, courses, the fee structure, examinations, placements, and other academic events. However, sometimes it becomes difficult for users to search for the desired information, as the website contains a vast amount of information that is spread over numerous pages. Such a process is cumbersome for new students and parents.

To solve this issue, an inquiry system may be developed through the use of a chatbot, a computer program that talks to and interacts with the user through speech or text, answering their inquiries. The system will make use of Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques to understand the inquiries from the user and provide appropriate information in response. The proposed Intelligent Student Support System will help the user to get quick, accurate, and automated responses for their queries related to colleges. This will reduce the burden of administrative staff and will

also reduce the need for the students to search for information manually by surfing the web. The Intelligent Student Support System will provide a user-friendly interface, where the user will be able to type their queries and get instant answers. The proposed project will focus on creating a multilingual conversational chatbot for enhanced user experience and efficiency of the enquiry system

2. RELATED WORK

The Chatbots were first developed using rules, which gave responses using pattern matching and the use of keywords. The research done by Mauldin gave the first demonstration of the use of chatbots and the limitations of the use of pattern matching and rules in the development of chatbots. [1]. Later, Shawar and Atwell were able to give the use of corpora and statistical methods in the development of chatbots. [2]. The development of sequence-to-sequence models is an important milestone in the development of chatbots. Vinyals and Le demonstrated the usage of sequence-to-sequence models for learning with the aid of dialogue data, though they also discussed the need for ensuring the consistency of the results of the models. [3]. Serban and other researchers discussed the development of generative end-to-end models for the strengths and weaknesses of such models for opendomain chat. [4]. A survey of neural models for chatbots describes the question answering, task-oriented, and social chatbots, along with the recent developments for each of the categories of chatbots. [5].

Large pre-trained conversational models, e.g., Dialog GPT, have enhanced the quality of single-turn responses by pre-training on vast amounts of conversational text, whereas Retrieval Augmented Generation (RAG) combines the benefits of the two approaches, avoiding factual mistakes and providing domain-specific responses. [6][7]. Taskoriented dialogue systems also saw significant progress, with the MultiWOZ dataset allowing work on multi-domain dialogue systems and the quality of state tracking and response. [8]. The role of persona and personalization, e.g., persona-based models, is also highlighted to emphasize the advantage of considering the persona in the conversational model to enhance the quality of the response.[9].

Chatbot design patterns, platforms, and styles are also reviewed in the conversational system, which highlights the fast-paced evolution of messaging style chatbots. [10]. There are some applied research papers written from the domain applications and UX point of view. Summaries and surveys are also used to give an overview of the chatbot technology and applications in different domains. [11]. Some reviews on the applications of chatbots in the field of education discuss the applications of chatbots in the field of learning, administrative, and query-based applications, and conclude that the field of educational chatbots is promising but undeveloped. [12]. Some other domain applications, like healthcare, design issues, and crowd-sourced assistants, also indicate some design and safety issues, similar to the university enquiry system. [13][14]. Some basic NLP books and recent surveys provide the theoretical background of the contemporary chatbots, namely, tokenization, lemmatization, embeddings, intent identification, semantic search, and evaluation metrics. This is relevant for the design of the preprocessing, intent/entity, and retrieval components of the college inquiry chatbot. [15][5]. Finally, recent research points to the need for privacy and data protection in the design of contemporary chatbots, which is relevant for the storage of the student queries and logs. [16].

Summary: From rule-based systems to neural and retrieval augmented systems, and now a growing emphasis on domain applications (which include education), there is a clear trend in the past work. For a college inquiry chatbot, a combination of a knowledge-based system, a well-performing NLP preprocessing, a lightweight retrieval system, and monitoring (logging of unanswered queries) follows the recommendations of the recent work

3. PROPOSED METHOD

The proposed system is Intelligent Student Support Chatbot, which aims to provide answers for queries related to college. The main idea of the proposed system is to provide information in an efficient manner using the chatbot system. The entire working of the system is divided into various segments. First, the appropriate information is collected from the college sources like the college website, brochure, etc. This information is then stored in the form of a knowledge base. The dataset contains information related to admissions, courses, fees, examinations, placements, etc.

The whole operation of the system is split into various phases. Firstly, the relevant information is collected from the official college sources such as college websites, brochures, and notices. The information is then arranged and kept in a structured knowledge base. The set of data includes information about admission, courses, tuition fees, examinations, placements, and other facilities provided by the college.

After the preprocessing stage, the system will be able to classify the user's intent using classification techniques like TF-IDF and Cosine Similarity. Once the user's intent is clear, the chatbot will be able to retrieve the most appropriate response from the knowledge base. However, if no appropriate response is available, the query will be stored in the database for later analysis

The chatbot is developed using Python with the help of the Flask framework for backend development, and HTML and CSS are used for frontend development. SQLite is used for storing the queries and responses of the users. Currently, the chatbot is supporting only text-based interaction, whereas in the future, it may also support voice-based interaction. The proposed method is useful for providing quick response, accessibility, and handling the queries of the students in an efficient way.

4. SYSTEM ARCHITECTURE

The system architecture of the Intelligent Student Support Chatbot is developed in such a way that it is able to provide the user with quick and accurate results for the questions he or she has in mind. The system architecture is modular in nature, and each module of the system is developed in such Fig. 2. ER Diagram a way that it is able to provide a particular service in the communication with the user.

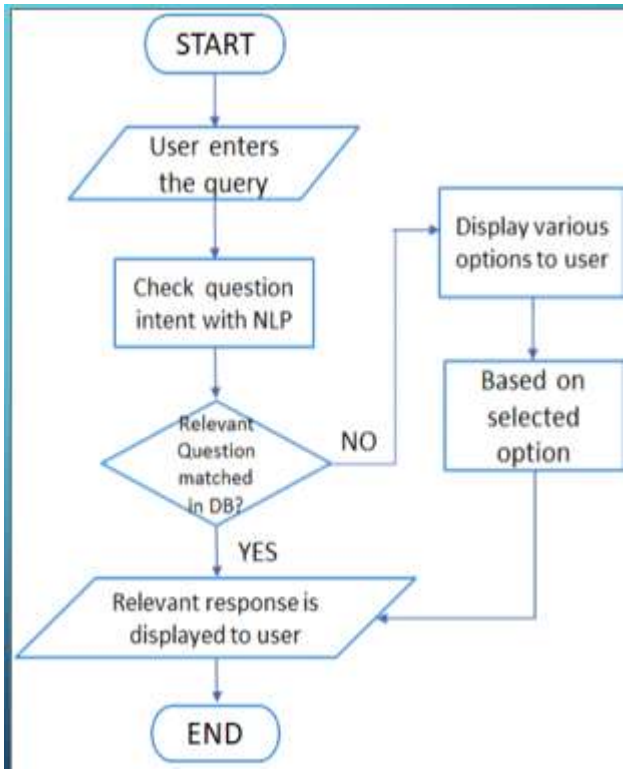


Fig -1: System Architecture

The system starts with the User Interface, which is created by using HTML and CSS. The user has the ability to enter their queries by using the text input field. Then, the query is sent to the backend server by using HTTP requests.

The Backend Server is created by using the Flask framework, which is written in Python. The backend server receives the user query and forwards it to the NLP processing module. The backend server is the controller of the system.

The NLP Processing Module is responsible for processing the text using techniques such as tokenization, filtering of stop words, and lemmatization. After the processing of the text is complete, the system uses the TF-IDF algorithm and cosine similarity to evaluate the user intent and match the query with the best possible output.

The system retrieves the appropriate output from the Knowledge Base, which contains organized information such as college-related information, including admissions, tuition fees, exams, placement, attendance, etc. If the best match is not available, the query is saved in the database for later processing.

Finally, the output is sent back to the user interface through the backend.

The system is highly scalable, modular, and optimized for queries related to students.

5. DATABASE DESIGN

The database for the Intelligent Student Support Chatbot is expected to store user information, queries, answers, and log data in an organized way. The Entity Relationship Diagram is a representation of how all the entities relate to each other.

A. Main Entities

1. **User:**
- User ID (Primary Key)
- Name
- Email
- Role (Student/Admin)

The User entity holds basic information of people interacting with the chatbot.

2. Query:
- Query ID (Primary Key)
- User ID (Foreign Key)
- Query Text
- Date Time

The Query entity holds all the questions asked by users. Each
query is related to a particular user based on User ID.

3. Response:

- Response ID (Primary Key)
- Query ID (Foreign Key)
- Response Text
- Confidence Score

The Response entity holds all the answers provided by the chatbot. Each response is related to a particular query.

4. Knowledge Base:

- KB ID (Primary Key)
- Category (Admissions, Exams, Fees, Placements, etc.)
- Question Pattern
- Answer Text

This entity holds pre-defined question and answer pairs used by the chatbot to provide responses.

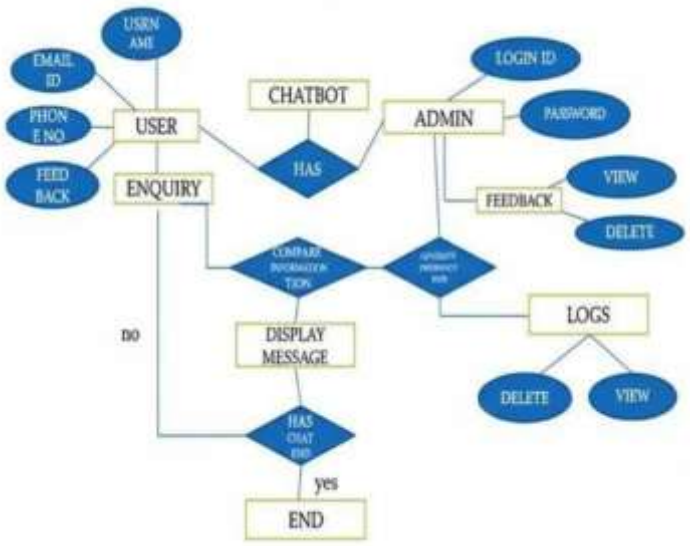
5. Unanswered Queries:

- Unanswered ID (Primary Key)
- Query Text
- Date Time
- Status (Pending/Updated)

Unmatched queries are stored in this entity for administrative purposes and future modifications.

B. Relationships

- One User can have multiple Queries (1:M relationship).
- One Query has one Response (1:1 relationship).
- The Knowledge Base provides responses for matching queries.
- Unmatched queries are stored in Unanswered Queries.



6. SEQUENCE DIAGRAM

The Sequence Diagram shows the way in which different components of the Intelligent Student Support Chatbot communicate with each other over time in response to a user query. It demonstrates the step-by-step communication between different objects in the system.

A. Main Components Involved

- User
- Chat Interface (Frontend)
- Flask Backend Server
- NLP Processing Module
- Knowledge Base / Database

B. Sequence of Operations

1. The User enters a query in the chat interface.
2. The Chat Interface sends the query to the Flask Backend Server through an HTTP request.
3. The Backend Server sends the query to the NLP Processing Module.
4. The NLP module will perform the preprocessing operations such as tokenization, removal of stop words, lemmatization, etc., along with the determination of the user's intent.
5. The processed query is matched with the appropriate information in the Knowledge Base using the concept of similarity.

6. The Knowledge Base sends the appropriate response to the Backend Server.
7. The query is stored in the database if no match is found.
8. The Backend Server sends the final response to the Chat Interface.
9. The response is shown to the User.

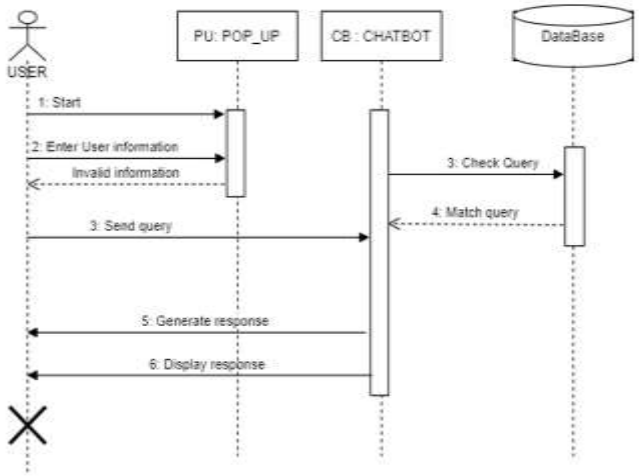


Fig. 3. Sequence Diagram

C. Purpose of the Sequence Diagram

The sequence diagram assists in understanding:

- The flow of messages
- The interaction between the frontend and backend systems
- The working of the NLP and database systems
- The handling of unmatched queries This diagram is effective for representing the processing of the chatbot system, which is done in real time

7. RESULTS AND DISCUSSION

Student Support Chatbot was successfully developed and tested using a structured dataset with information related to colleges, such as admission, fees, examinations, attendance, and placement. The Student Support Chatbot was developed using Python and Flask for the backend and HTML/CSS for the frontend. SQLite database is used to store user queries and their corresponding answers.

The system was tested successfully, and the Intelligent Student Support Chatbot was able to answer most of the frequently asked questions of the user in their own language. The use of NLP preprocessing techniques like tokenization, stop word removal, and lemmatization was also effective in understanding the queries of the users better. The use of TFIDF and cosine similarity was effective in identifying the answers with reasonable accuracy.

The system was also able to show fast response time under normal usage conditions. However, the response time may also worsen if several users interact with the system at the same time, depending on the capabilities of the server. The system also showed the ability to store the queries that were left unanswered or unmatched in a CSV file/database file, hence improving the system over time.

The overall results show that the proposed chatbot is effective in reducing the manual enquiry workload. In general, the result of the study shows that the proposed chatbot is effective in reducing the manual enquiry workload and provides instant access to information. The proposed system is scalable, and further improvements can be made by using multilingual, voice, and AI technologies

Fig. 4. User Interface 1

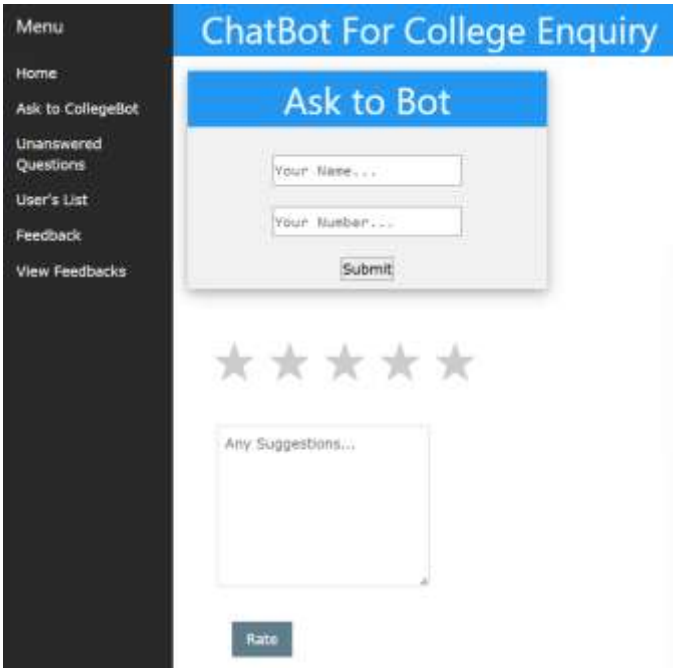


Fig. 5. User Interface

8. CONCLUSION

In this project, an Intelligent Student Support System through the creation of a conversational chatbot was designed and implemented to automate the college enquiry services. Such an approach ensures that the queries of the users, including those related to admission, examinations, fees, placements, etc., are answered quickly and correctly through the application of natural language processing techniques, the chatbot is able to understand the user's queries and respond appropriately.

The system would ensure that the manual handling of the queries is reduced, hence saving the students as well as the administration considerable time. The system would also ensure that the user is provided with an interface through which the user would be able to interact with the system through natural language. The storage of the unanswered queries would also ensure the enhancement of the chatbot system through the enhancement of the knowledge base.

The existing system, although capable of handling text -based communication, will also benefit with the addition of voice support, multilingual support, and the utilization of advanced AI models for improved context understanding. Thus, the proposed system will offer an efficient, scalable, and cost-effective solution for the improvement of communication in the institution.

ACKNOWLEDGEMENT

We would like to extend our sincere thanks to our project guide for their valuable guidance and support towards the development of this project. We would like to extend our sincere thanks to the Head of the Department and faculty members of the Department of Computer Science and Engineering for providing us with all the necessary facilities and support to complete this project.

We would like to extend our sincere appreciation towards our institution for providing us with an opportunity to work on this project as part of our curriculum. Finally, we would like to extend our sincere appreciation towards our friends and family members for their constant motivation and support throughout the project.

REFERENCES

1. M. L. Mauldin, "ChatterBots, TinyMuds, and the Turing Test: Entering the Loebner Prize Competition," in Proceedings of the AAAI Conference, 1994.
2. A. A. Shawar and E. Atwell, "Using corpora in machine-learning chatbot systems," International Journal of Corpus Linguistics, vol. 10, no. 4, pp. 489–516, 2005.
3. O. Vinyals and Q. Le, "A Neural Conversational Model," arXiv:1506.05869, 2015.
4. I. V. Serban et al., "Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models," in Proceedings of AAAI, 2016.
5. J. Gao, M. Galley, and L. Li, "Neural Approaches to Conversational AI," ACM/ACL Tutorials and Surveys, 2018–2019.
6. Y. Zhang et al., "DialoGPT: Large-Scale Generative Pre-training for Conversational Response Generation," arXiv:1911.00536, 2019.
7. P. Lewis et al., "Retrieval-Augmented Generation for Knowledge Intensive NLP Tasks," in Advances in Neural Information Processing Systems (NeurIPS), 2020.
8. P. Budzianowski et al., "MultiWOZ — A Large-Scale Multi Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling," arXiv:1810.00278, 2018.
9. J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A Persona-Based Neural Conversation Model," in Proceedings of ACL, 2016.
10. L. C. Klopfenstein et al., "The Rise of Bots: A Survey of Conversational Interfaces, Patterns and Paradigms," in Proceedings of ACM, 2017.
11. E. Adamopoulou and L. Moussiades, "An Overview of Chatbot Technology," in Artificial Intelligence Applications and Innovations, IFIP AICT, 2020.
12. P. Smutny and P. Schreiberova, "Chatbots for Learning: A Review of Educational Chatbot Applications," Computers & Education, 2020.
13. A. Fadhil and G. Schiavo, "Designing for Health Chatbots," arXiv:1902.09022, 2019.
14. W. S. Lasecki et al., "Chorus: A Crowd-Powered Conversational Assistant," in Proceedings of UIST/ACM, 2013.
15. D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed., draft, 2025–2026.
16. A. (Author), "A literature review of user privacy concerns in conversational AI systems," Journal/ASIS Review, 2024