SpeakOne: Multilingual AI Chatbot for Student Support

Abhijit Shankar Jadhav[1], Gajul N.B.[2]

[1]PG Student, [2]Assistant Professor, Department of Computer Applications, SVERI's College of Engineering Pandharpur, India.

E-mail: [1]abhijitsjadhav@coep.sveri.ac.in, [2]nbgajul@coe.sveri.ac.in

Abstract

Higher education organizations face a huge number of repetitive inquiries from students regarding admissions, fee payments, exams, scholarships, placements, and other campus-related aspects. Inquiries are made in various languages, resulting in communication difficulties and increased work pressure on the administrative staff. This paper introduces the concept of SpeakOne, which is an AI chatbot for higher education organizations to answer student inquiries using a single conversational pipeline automatically. The designed system has features including automatic language detection, natural language processing, intent recognition, institutional knowledge base lookup, and an external generative-AI fallback module. Whenever there is a suitable institutional response to an inquiry, it is fetched by the system; otherwise, it is forwarded to an external AI language model. The recognized language is used throughout the processing pipeline to generate appropriate answers in the corresponding language. The system design has included conversation logging, user feedback, analytics, and an administrative dashboard that monitors unanswered questions and manages the knowledge base. The operational system dashboard indicates 15,234 queries, five languages, 95% success indicator for the system as a whole, and 45 unanswered queries; but these numbers refer to the operational system data and not machine learning benchmark-based accuracy levels.

Keywords: Multilingual Chatbot, Language Detection, Intent Recognition, Student Query Automation, Conversational AI, Higher Education Support System, Digital Campus Automation, AI-Based Student Assistance.

DOI: https://doi.org/10.36548/rrrj.2026.2.006
Extended version of article received from 6th **International Conference on Artificial Intelligence, 6G Communications and Network Technologies 2026**
Received: 10.07.2026, received in revised form: 13.08.2026, accepted: 04.09.2026, published: 15.09.2026

1. Introduction

Higher education institutions receive many queries from students regarding admission, fees, examinations, scholarships, placements, academic processes, and campus facilities. These inquiries are traditionally handled by administration departments, phone support centers, websites, or FAQ sections on institution websites. While these methods have proven to provide accurate information, they still entail ongoing participation of staff members and do not always provide instant help outside of office hours. This is why the growing trend of digitalization of academia requires development of conversational agents that could provide timely information to students.

This issue becomes more complicated in multilingual educational environments, where the students could be using English along with the regional languages as means of communication. Chatbot limited to English dialogue would become yet another obstacle to communication with those students who prefer to speak in their native languages, like Hindi and Marathi. Therefore, multi-lingual conversational systems should not only provide simple translations, but also recognize the language of the query, understand what is the student requesting, retrieve the information and generate a response, preserving the linguistic context of the dialogue. For Indian languages, some extra features, like Devanagari script handling and specific language natural language processing, affect the architecture of conversational systems. Recent publications show the rising popularity of multilingual chatbots, Indian language conversational systems, and linguistically aware dialogue systems; however, they usually study each aspect separately or apply in areas different from higher education student assistance.

One additional problem concerns the reliability of the answers produced by the conversation AI system. A general purpose generative language model can answer a variety of questions but using only generated answers to institution-based queries would produce answers that do not correlate with the institution's policies or services. On the other hand, the use of an FAQ or knowledge base provides institution-related information, but such information cannot cover all conceivable questions that students may have. Thus, there should be a middle ground where predefined institution information is preferred, whereas queries not addressed by the knowledge base are answered by generative AI.

To overcome these difficulties, in this paper, we introduce SpeakOne AI Unified Multilingual Chat Engine that is a web-based conversational engine developed to provide automated help to students in institutions of higher education. This conversational engine uses a combination of automatic language detection, natural language processing and intent detection, institution's knowledge base retrieval and a fallback mechanism using external generative AI engines in a unified processing pipeline. Currently, the system works in multiple languages, but the functionality assessment presented in this paper uses English, Hindi and Marathi. The language detected by the engine is kept unchanged through the entire conversation, thus, students can ask questions in their chosen language, and receive answers in the same language. Besides the conversational engine, the system also includes conversation log, feedback, analytics, and knowledge base administration modules.

The key strength of this research lies in bringing together multi-language interaction, institution-specific information retrieval, intent-based query processing, and conditional generative AI response generation within a unified student-support solution. In contrast to a general multilingual chatbot or the traditional FAQ mechanism, the proposed SpeakOne solution unifies the controlled nature of institution-specific knowledge retrieval with the flexibility of an AI-based fallback mechanism. It is designed as a real-world framework aimed at alleviating repetitive administrative tasks and ensuring the accessibility of institution information for students interacting using English, Hindi, Marathi, and other configured regional languages. The rest of the paper describes the related literature, the architecture and processing approach, and implementation and results of the conducted experiments.

2. Literature Review

Conversational systems using multiple languages have been getting more attention as access to artificial intelligence and natural language processing have become easier for real-time interaction between humans and computers. Higher educational institutions may benefit from such systems since students usually look for admission-related information about tests, fee structure, scholarships, and other services offered by institutions. Nevertheless, implementation of multilingualism poses additional difficulties concerning language detection, semantic understanding, intent detection, and response generation. The existing literature provides several approaches to multilingual chatbot development, conversational assistance,

ISSN: 2583-7079

and language-aware interaction, but relatively fewer studies focus specifically on integrating these capabilities into a unified higher-education support system.

The TalkGenie is a multilingual chatbot with intent and sentiment analysis capability which stresses the importance of intent and sentiments in addition to the understanding of language used in the questions. Saranya et al. [1]. However, this research brings to light the importance of intent recognition in matching the queries posed by students to the services offered at institutions but does not coincide with the main focus of this research. The ChatSense chatbot, developed by Saravana et al. [2], is designed to improve communication by overcoming the obstacles posed by language barriers. However, the chatbot fails to provide any institution-specific responses. On the other hand, SpeakOne incorporates an institution-specific knowledge base and also uses an AI model from the outside for any fallbacks. The paper written by Qin et al. [3] discussed an AI-based speaking assistant that was designed to help non-native speakers communicate using multiple languages in real time. Though the application of such a speaking assistant cannot be used in the field of higher education administration, it emphasizes the importance of multilingual processing in SpeakOne.

Rodríguez [4] explored phraseological interpreting enhanced by technologies via live-streaming platforms, analyzing its pedagogical applications rather than the issue of automated queries from institutions. This research promotes language technology in general education applications but cannot help with intent-based chatbots for institutions. The work of Liang et al. [5] examines multi-agent debate techniques to improve reasoning skills of large language models. According to their findings, the interactions between agents may influence the nature of the generated response. Even though this problem is not a concern in the context of SpeakOne, it can justify the usage of generative AI as an additional data source for institutions. Priya et al. [6] presented a cross-language chat app called Cross-Language Connect. It performs similarly to SpeakOne in terms of multilingualism, but its primary function is the user-to-user interaction, not institutionally-oriented assistance.

Kapse and Parihar [7] have stressed the importance of multilingual chatbots for Indian languages, thus directly linking them to SpeakOne's aims of supporting English, Hindi, Marathi, and other local languages. They demonstrate that multilingualism alone is not enough for an effective institutional response, thus underscoring the importance of language, intent detection, and domain knowledge used by SpeakOne. Goutham et al. [8] have explored the issue of multilingual chat in real time via chat translation, stressing the importance of quick reactions. But their interest in translation is different from SpeakOne's aim to detect institutional intent and provide a customized answer. Pedregosa et al. [9] have described scikit-learn, which is a machine learning toolkit useful for implementation of NLP modules but not comparable to chatbots research. Similarly, Mane et al. [10] conducted a survey on Devanagari language chatbots, which is useful for solving problems related to script handling and response generation in the Indian environment.

Suryawanshi et al. [11] conducted research on text-to-speech conversion and its relevance in language-specific systems, while Mishra et al. [12] designed a chatbot in Hindi for health-related questions. Finally, George et al. [13] conducted a survey on Indian-language chatbot development, focusing on issues that go beyond just translation. Thus, from the above-mentioned literature, it is clear that there are a number of challenges faced when designing a chatbot for the Indian languages, such as script handling, language understanding, and domain adaptation. The goal of SpeakOne is to combine these multilingual aspects with intent recognition and knowledge base

2.1 Research Gap

There are some important features that have been highlighted by the reviewed literature including multilingual conversation [1], [2], [7], [8], [9], language assistance in real time [3], technology-assisted multilingual communication in education [4], and response-generation using AI [5]. However, these features have not been analyzed together.

There is therefore an existing gap in the merging of multilingual language detection, intent recognition, knowledge retrieval for institutions, and generative-AI fallback into one higher education support system. Multilingual chatbots could lower the barrier of language but do not necessarily provide institution-specific information. On the other hand, knowledge-based support systems could provide institution-specific information but only if the student's query is already contained in the database.

This is achieved by the SpeakOne AI platform through its hybrid system design, which includes the identification of the language and intention behind a student's question and the subsequent attempt at finding a preset response from the institution's knowledge base. In cases where the desired response cannot be retrieved, an AI language model outside the system is employed for the generation of the desired response. Finally, the generated response is rendered in the appropriate language. The system therefore brings together the best of institutional knowledge retrieval and AI generative systems as well as conversation logging, analytics, feedback, and knowledge base administration.

Therefore, the literature indicates the relevance of multilingual conversational AI, but this paper is focused on such AI within the institution-oriented multilingual student help framework.

3. Proposed Work

The current system developed as a hybrid conversational system that could automate routine questions posed by students in higher education institutions. The system tries to answer the question raised by searching for the required information from the knowledge base specific to the institution. The external AI language model is called upon only if a proper predefined answer cannot be found. This allows the separation of the institutional information search from the generation of the answer, hence allowing control over answering the questions that are repetitive but also having the ability to answer those that are not present in the knowledge base.

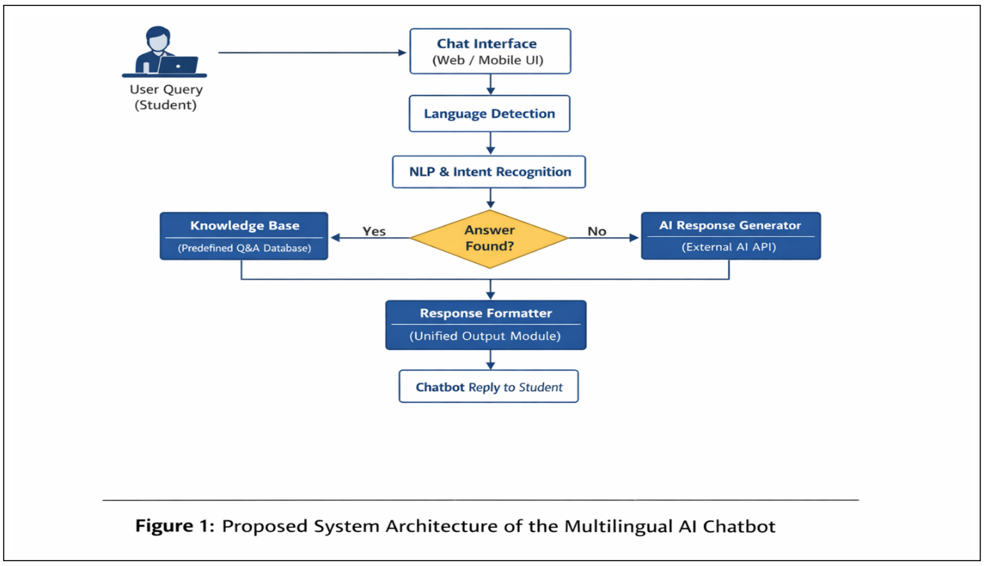

Figure 1: Proposed System Architecture of the Multilingual AI Chatbot

Figure 1. Flowchart of the proposed system

3.1 System Objectives

The system is built to fulfill five major functional requirements. Firstly, it allows students to ask questions via a web-based conversational interface. Secondly, it detects the language of the input query so that any further processing and answering can be done in a proper linguistic environment. Thirdly, it finds the intent behind the question asked and then retrieves a proper institutional response based on that intent. Fourthly, in case the response needed is not found in the knowledge base, then the query is directed to an external AI language model for generating a response. Lastly, the response is formatted and given to the user in the relevant language.

3.1 Overall System Architecture

The architecture consists of six functional processing stages:

1. Chat interface
2. Language detection
3. NLP and intent recognition
4. Institutional knowledge-base retrieval
5. External AI response generation
6. Response formatting and delivery

The query from a user is provided through the chatbot interface. This query is processed by the language detection engine, which recognizes the language used in the query. The language recognition result and query are then given to the NLP and intent recognition engine. The intent representation derived is then used to look up in the organization's knowledge base.

The system uses a conditional decision mechanism after knowledge-base retrieval:

$$R(q) = \begin{cases} R_{KB}(q), & \text{if a suitable knowledge-base answer exists} \\ R_{AI}(q), & \text{otherwise} \end{cases}$$

where q represents the student query, R_{KB} denotes a response retrieved from the institutional knowledge base, and R_{AI} denotes a response generated through the external AI language-model API.

This architectural condition ensures that not all the questions are unnecessarily sent to the generative model located outside. Questions related to institutional information are answered via the predefined content, while questions that fall outside of the known information base are answered using the fallback approach. The generated answer is then processed by the formatting module and provided to the student.

3.2 Multilingual Query Processing

The multilingual component is responsible for identifying the language of the incoming query and maintaining the language context throughout the response-generation process. The current implementation supports English, Hindi, Marathi, and additional regional-language configurations. The functional testing carried out in this study, however, was based on queries in English, Hindi, and Marathi. These queries included the typical requirements of students with regard to issues related to admissions, fees, examinations, scholarships, placement and facilities at the campus. The evaluation was done based on the functional aspects of multilingual query processing, intent recognition, knowledge base lookup, fallback responses and same-language response generation. This current study does not make any claims about a separate labeled benchmark data set or statistical accuracy per language due to the lack of such benchmarking data set and analysis in this prototype study. The system is intended to provide a same-language response rather than translating every interaction into a single internal language. In case of Hindi and Marathi languages, the submitted text is sent to the language detector module where the appropriate language is detected. This information on the detected language, along with the original query, is kept and forwarded to the NLP and intent detection modules. The process of processing of the query is done through the same pipeline as in the case of English, and the detected language is kept for generating responses. Hence, the Hindi and Marathi queries are not considered as different chatbot systems; they just go through the general pipeline.

Let the incoming query be represented as

$$q = (x, l)$$

where x is the textual query and l is its detected language. The language-processing stage determines

$$\hat{l} = D(x)$$

where $D(\cdot)$ represents the language-detection operation and $\hat{l}$ is the detected language.

The detected language is kept as context in the subsequent processing steps. Therefore, in case the input is in Hindi, then the response generation should keep the output in Hindi; English and Marathi inputs are responded in English and Marathi respectively. This is shown by the automatic detection and same-language responses that can be done on the supported languages according to the language support table in the manuscript.

Language preservation is especially critical in the proposed application since what is sought is not only multilingual information access but also multilingual interaction. Hence, the chatbot sees language detection as part of the query-processing pipeline and not translation.

3.3 NLP and Intent Recognition

Following language detection, the query is processed by the NLP and intent-recognition module. The objective of this stage is to transform an unrestricted question by students into an intent that can be correlated with institutional knowledge. In the case of Hindi and Marathi language, the submitted text is initially fed into the language detection module, which recognizes the language of input. This is followed by forwarding both the original text and language recognized to the common natural language processing and intent recognition pipeline. The language recognized is maintained till knowledge base retrieval and response generation to provide the output in the corresponding language.

For a query (x), the intent-recognition process can be represented as

$$\hat{c} = I(x, \hat{l})$$

where $I(\cdot)$ denotes the intent-recognition process, $\hat{l}$ is the detected language, and $\hat{c}$ represents the identified query category.

The intended application contains recurring administrative categories such as:

- admission-related queries;
- fee-related queries;
- examination-related queries;
- scholarship-related queries;
- placement-related queries; and
- campus-facility queries.

Questions on admission process will be matched with the intent regarding admission while the questions on exam schedules will be matched with the intent related to exams. The intent model helps in interpreting the input question provided by the user in its original form at the retrieval stage.

The distinction between the language detection and the intent recognition is crucial since the two queries could express the same institutional intent through different languages. Therefore, the system could recognize the interaction in terms of the query content, language detection, and the intent, rather than recognizing each language as a separate chatbot.

3.4 Institutional Knowledge-Base Retrieval

The knowledge base represents the controlled information repository for commonly asked organizational questions. It holds pre-defined question-answer information related to academic and administrative services. The current system was tested using frequently asked questions covering areas such as admissions, fees, examinations, and scholarships.

For an identified intent $\hat{c}$ the retrieval process searches the institutional knowledge base for an appropriate response:

$$R_{KB} = K(\hat{c}, x, \hat{l})$$

where $K(\cdot)$ denotes knowledge-base retrieval operation. Therefore, the retrieval process considers the semantic meaning and the intention of the request while keeping the language in which it is made. In case there exists a relevant institutional answer, the retrieved answer is transferred directly to the response-formatter module. This system enables the deterministic way of responding to the information already kept by the institution.

Moreover, the knowledge base approach offers administrative benefits. Information of the institution can be changed independent of the conversation system, hence allowing frequent updating of information without changing the entire structure of the chatbot.

3.5 External AI Fallback Mechanism

It is also a disadvantage of a fixed knowledge base because not all formulations can be included there. Thus, SpeakOne uses the external AI language model to serve as a backup tool.

The decision process can be represented as

$$A(q) = \begin{cases} K(q), & \text{if } K(q) \text{ returns a suitable answer} \\ G(q, \hat{l}, \hat{c}), & \text{if no suitable answer is retrieved} \end{cases}$$

where $K(q)$ represents knowledge-base retrieval and $G(\cdot)$ represents external AI response generation.

If the knowledge base does not have an appropriate answer, the query is directed to the external AI language-model API, and an answer is generated, which is then provided to the response formatting phase.

The dual nature of this system gives two different response streams:

Known institutional query

$$\text{Query} \rightarrow \text{Intent} \rightarrow \text{Knowledge Base} \rightarrow \text{Response}$$

Unresolved query

$$\text{Query} \rightarrow \text{Intent} \rightarrow \text{Knowledge Base} \rightarrow \text{AI API} \rightarrow \text{Response}$$

It is important to distinguish this because it will ensure that the proposed system is not classified as an interface to a generic AI model. While the knowledge base is the main resource of answers, generative AI expands the scope of the system in addressing unanswered questions.

3.6 Response Formatting and Same-Language Delivery

The responses created through the knowledgebase and external AI system can come through various processing paths. For that reason, the response formatting component offers a standardized output interface.

The response formatter receives:

$$R \in \{R_{KB}, R_{AI}\}$$

along with the detected language $\hat{l}$, and produces the final chatbot response:

$$R_{\text{out}} = F(R, \hat{l})$$

where $F(\cdot)$ represents response formatting. Formatter guarantees that replies from both processing paths have the same conversational formatting and appear in the respective language of the user. Therefore, the interaction with the user is uniform regardless of which source the reply came from: the organizational knowledge base or an external AI. The architectural design of the solution makes this step explicit between reply generation and delivery.

3.7 Conversation Logging and Administrative Management

In this system, logging conversations, gathering feedback, analysis, and managing knowledge base are some of the support services provided.

For each conversation, the system would be able to link information such as:

$$L_i = \left(q_i, \hat{l}_i, \hat{c}_i, R_i, f_i\right)$$

where q_i is the submitted query, $\hat{l}_i$ is the detected language, $\hat{c}_i$ is the recognized intent, R_i is the delivered response, and f_i represents the associated feedback information when provided.

The logged data can be used to detect frequently asked information, queries that need further solving, and places where knowledge base needs extension. It is supposed that the administrative dashboard should be user-friendly to allow non-technical users, like volunteers, to maintain the knowledge base.

The administration dashboard of the manuscript includes four different tabs with information about overall, FAQ, analytics, and settings. There are some statistics shown on the prototype, including the total number of queries, the number of supported languages, success rate, and unanswered queries.

3.8 End-to-End Processing Algorithm

The complete processing procedure can be summarized as follows:

Algorithm: Multilingual Student Query Processing

Input: Student query q.
Output: Response R_{out}

1. Receive the student query through the chatbot interface.
2. Detect the language $\hat{l} = D(q)$.
3. Process the query using the NLP pipeline.
4. Determine the query intent $\hat{c} = I(q, \hat{l})$.
5. Search the institutional knowledge base using the query and identified intent.
6. If a suitable institutional response exists, retrieve R_{KB}.
7. Otherwise, forward the unresolved query to the external AI language-model API.
8. Generate R_{AI} using the query context and detected language.
9. Pass the selected response through the response-formatting module.
10. Return the response in the corresponding language.
11. Record the interaction and associated feedback/analytics information.

Therefore, the suggested system does not depend on only one mechanism for generating responses. The system utilizes language-aware query processing, institution identification based on intention, conditional generative-AI, and formatting of responses in one pipeline.

4. System Implementation

The proposed work was deployed as an online chat engine which combines multilingual query processing, institution knowledge retrieval, external AI-generated responses, response formatting and administration. The process is done in accordance with the flow of operations as outlined in the suggested architecture, whereby a student's query will go through language identification and intent recognition, after which the system will decide whether the response is available within the institutional knowledge database or needs to be generated externally by AI.

4.1 Implementation Environment

The designed system consists of two main interfaces that will be available to users: student interface in form of chatbot and administrative interface. Chatbot interface is meant to submit queries and get their answers. Administrative interface has the capability of management of knowledge base of chatbot and system activity control. The screenshots from the implementation of the manuscript show both interfaces, including chatbot interaction interface and administrative dashboard with query, language, success rate, and unanswered query information. The implementation is organized around the following functional components:

Table 1. Components and its implementation role

Component	Implementation role
Chat interface	Accepts student queries and displays responses
Language detection	Determines the language of the submitted query
NLP and intent recognition	Determines the semantic category of the query
Knowledge base	Provides predefined institutional answers
External AI model	Handles queries for which no suitable knowledge-base answer is available
Response formatter	Produces a consistent response in the detected language
Conversation logging	Records interactions for subsequent analysis

Analytics/dashboard	Provides administrative monitoring and management

This modularity in design allows separation of understanding the query, searching for information, generating responses, and managing processes independently without having to change the entire process of conversation.

4.1 Student Query Interface

This is the portal from which the conversation begins. Students will pose their queries in natural language via the chatbot input box, from there, the query will enter the multilingual processing workflow. This portal will be designed such that users do not have to select the language manually prior to posing their query. The language configuration includes English, Hindi, Marathi, and other regional languages.

4.2 Multilingual Processing Implementation

Multilingual preprocessing: The query from the student is taken as text and fed into the language detection step before processing the intent. The detected language is saved as an attribute of the interaction and used later on in the response-processing stage. Queries in English, Hindi, and Marathi are processed using the exact same pipeline steps for natural language processing and intent recognition. Textual queries written in Devanagari script, like Hindi and Marathi, are kept as Unicode text through the conversation pipeline so that the original script can be retained for generating responses in the output stage. The system does not necessarily translate the input query into one particular language output.

4.3 Intent Processing

The intent processing component will correlate the free-form questions asked by students with pre-defined academic and administrative classes. The current implementation of the application caters to admission, fees, exams, scholarships, placements, and facilities on campus. The intent is then fed into the knowledge base retrieval component.

4.4 Knowledge-Base Implementation

The knowledge base for institutions was designed to be the main provider of predefined academic and administrative answers. The retrieval first strategy verifies whether there is a predefined answer and only then applies the external AI model. Such an approach is more suitable for the institutional knowledge base as it is possible to keep the most commonly asked information as predefined answers. Consequently, updating this type of information will not require any change in the whole conversational system.

4.5 External AI Fallback Implementation

The external AI response generator gives a secondary channel for the generation of responses. In case no response can be found from the knowledge-base retrieval phase, the query will be passed on to the external AI response generator, where a response will be created. The response will then be sent to the response formatting module.

4.6 Conversation Logging and Feedback

Conversation logging and user feedback facilities have been put into place. The interaction logs can be used for spotting commonly asked information and any unresolved queries. The data logged includes the query made, the language detected, the intention detected, the response provided, and the feedback. This provides a framework for detecting information gaps and updating the institutional knowledge base.

4.7 Administrative Dashboard

The use of an administrative dashboard was employed to aid the monitoring and management of the chatbot. Information is provided regarding the activities of the system, the languages it supports, response handling, and unanswered queries. Other sections available include that for frequently asked questions, analytics, and system settings. The dashboard thus allows for management whereby the information about the interactions can be analyzed and unanswered queries evaluated for future knowledge-base updates.

5. Results and Discussion

Performance of the multilingual processing feature was investigated on the basis of queries in English, Hindi, and Marathi. For each language, the system has been tested functionally in terms of language detection, intent processing, knowledge base lookup, fallback response generation, and response generation in the same language. The system maintained the detected language as contextual information through the processing chain such that the final response was delivered in the detected language. Nonetheless, separate performance values for English, Hindi, and Marathi have not been stated due to the absence of benchmark dataset in the current research. Hence, the findings of the current research can only be considered multilingual system observations.

5.1 Deployed System Home Page

A unique home page for the system is provided where the user can access the chatbot to begin communication. As seen in Table 2, the system has the capability of automatically detecting the language used and responding in the same language. However, it was specifically tested with English, Hindi, and Marathi languages, apart from others.

Table 2. Multilingual interaction support

Language	Language code	Automatic detection	Same-language response	Script
English	EN	Yes	Yes	Latin
Hindi	HI	Yes	Yes	Devanagari
Marathi	MR	Yes	Yes	Devanagari
Additional regional languages	KN/TE	Yes	Yes	Native script

The interface introduces SpeakOne AI as a multilingual platform aimed at helping students, and gives direct access to the chatbot features. The home page also includes system indicators such as language support exceeding five languages, over 10,000 questions answered, an indicator of a 95% success rate, and 24/7 availability. All these indicators are provided by the deployed interface as system indicators. The home page interface is illustrated in Figure 2.

The homepage serves as the starting point for the conversation interface while delivering information about the main functionalities of the proposed system. Specifically, the language capability and availability functionalities map to the intended aim of creating an easy access for students' assistance without the need for any intervention on behalf of the administrators. This interaction can be facilitated via the use of language detection, NLP, intent recognition, knowledge-base lookup, and conditional AI response generation.

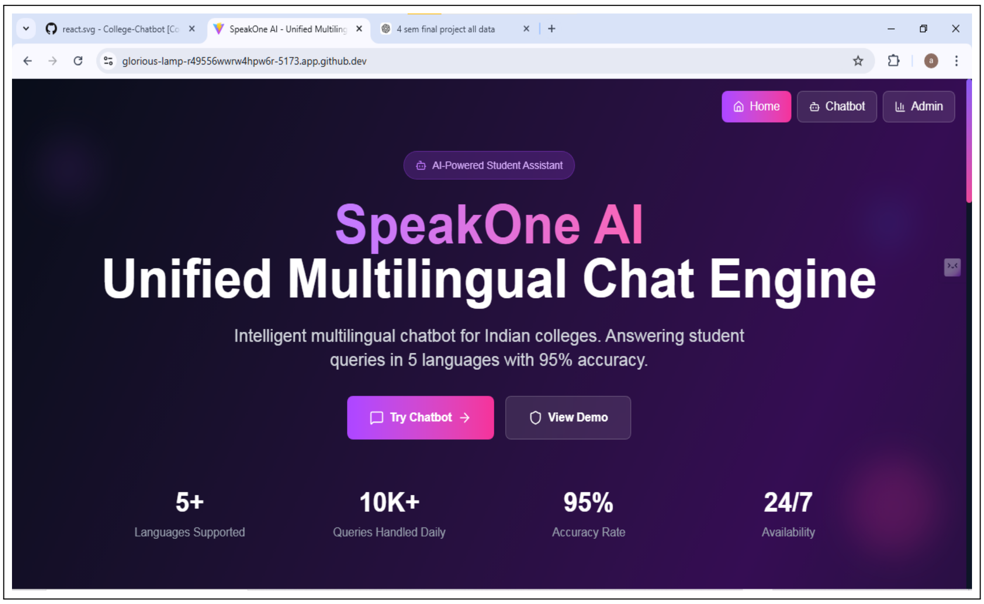

Figure 2. Home page of the deployed SpeakOne

5.2 Student Chatbot Interface

This is the user interface through which natural language queries are presented to the student chatbot. Figure 3 illustrates how the interface consists of an area for conversations where the interactions are displayed and another where the student submits the queries.

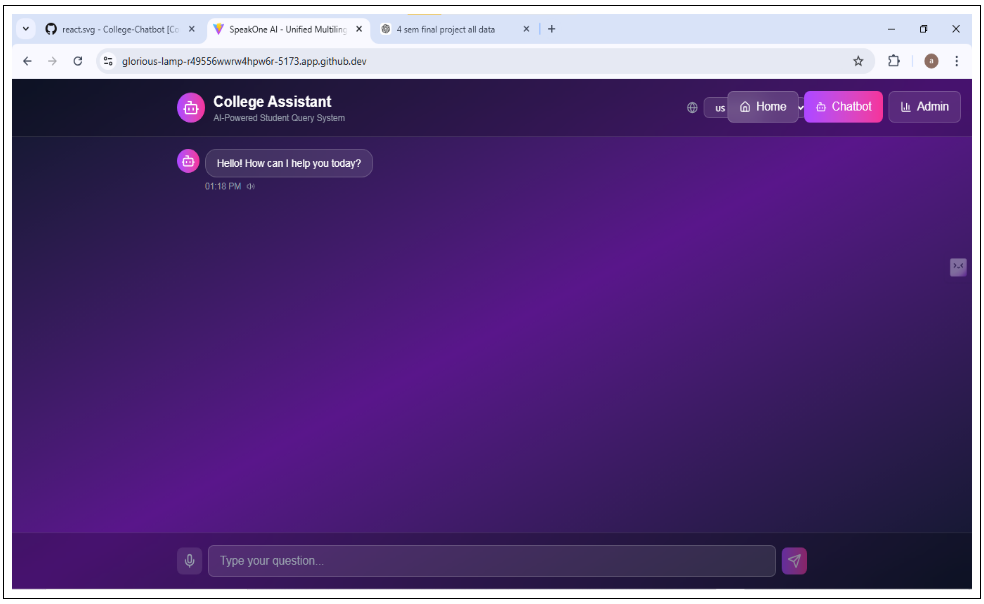

Figure 3. Student chatbot interface for multilingual query interaction

The query that is submitted undergoes processing by the multilingual pipeline that is explained in Section 3. Initially, the system detects the language of the query and then conducts NLP and intent recognition. The detected intent is further used for searching in the institution's knowledge base. In case there is a suitable answer available, the predefined institutional response is returned; otherwise, the query is directed to the external AI response generator.

The configuration of languages, which is employed in the system, facilitates automatic language detection and responses in English, Hindi, and Marathi languages with the regional language configurations also recognized by the system. Thus, students can use the system without having to choose a specific chatbot for different languages.

Testing was performed with the academic and administrative queries such as admissions, fees, examinations, scholarships, placements, and campus facilities. These are the primary applications domains of the chatbot that is suggested.

5.3 Multilingual Query Processing and Response Generation

Multilingual capabilities were tested with questions posed in English, Hindi, and Marathi languages. Language detected in the query is stored by the system as contextual information and remains so during subsequent processing and response generation. Hence, the language of the query itself dictates the language of the response.

In case of queries that exist in the institutional knowledge base, the system uses retrieval approach to generate responses. In the absence of a suitable response, external AI language model comes into play.

This dual response approach is vital since the chatbot does not rely solely on question-answer pairs. Institutional knowledge base serves as a reliable source of institutional data for commonly asked questions, while AI response approach expands its scope to questions not found in the database.

Table 3. Representative multilingual student query-response examples

Language	Example student query	Example response
English	“What documents are required for admission?”	“The required admission documents include the prescribed application form, academic certificates, identity proof, and other documents specified by the institution.”

Hindi	" प्रवेश के लिए कौन-कौन से दस्तावेज़ आवश्यक हैं?''	"प्रवेश के लिए निर्धारित आवेदन पत्र, शैक्षणिक प्रमाणपत्र, पहचान प्रमाण तथा संस्थान द्वारा निर्दिष्ट अन्य दस्तावेज़ आवश्यक हैं।''
Marathi	"प्रवेशासाठी कोणती कागदपत्रे आवश्यक आहेत?''	"प्रवेशासाठी निर्धारित अर्ज, शैक्षणिक प्रमाणपत्रे, ओळखपत्र आणि संस्थेने निर्दिष्ट केलेली इतर आवश्यक कागदपत्रे आवश्यक आहेत.''
English	"When is the examination schedule released?"	"The examination schedule is announced through the institution's official academic notifications."
Hindi	"परीक्षा की समय-सारणी कब जारी की जाती है?''	"परीक्षा की समय-सारणी संस्थान की आधिकारिक शैक्षणिक सूचना के माध्यम से जारी की जाती है।''
Marathi	"परीक्षेचे वेळापत्रक कधी जाहीर केले जाते?''	"परीक्षेचे वेळापत्रक संस्थेच्या अधिकृत शैक्षणिक सूचनांद्वारे जाहीर केले जाते.''

Examples of the same query in English, Hindi and Marathi languages are illustrated in Table 3. These examples give an indication of the functionality of the proposed system. They are only meant to show how the interaction is made and not to replace the quantitative evaluation in terms of language.

5.4 Administrative Dashboard and System Usage

The admin part offers monitoring and management capabilities to the deployed chatbot. The dashboard gives an insight into the activities of the system, including number of requests, languages supported, success details, and unanswered requests. The dashboard also offers different dashboards for FAQs, analytics, and system configurations.

As depicted in Figure 4, the dashboard provides information that includes 15,234 queries overall, 5 language types supported, 95 percent success rate, and 45 queries without answers. This information serves as an indicator on the usage and query answering process in the system. Additionally, the dashboard is integrated with the conversation logging system, and interaction logs can be analyzed for the most repeated queries and unanswered queries, which can later be used for updating the institutional knowledge base.

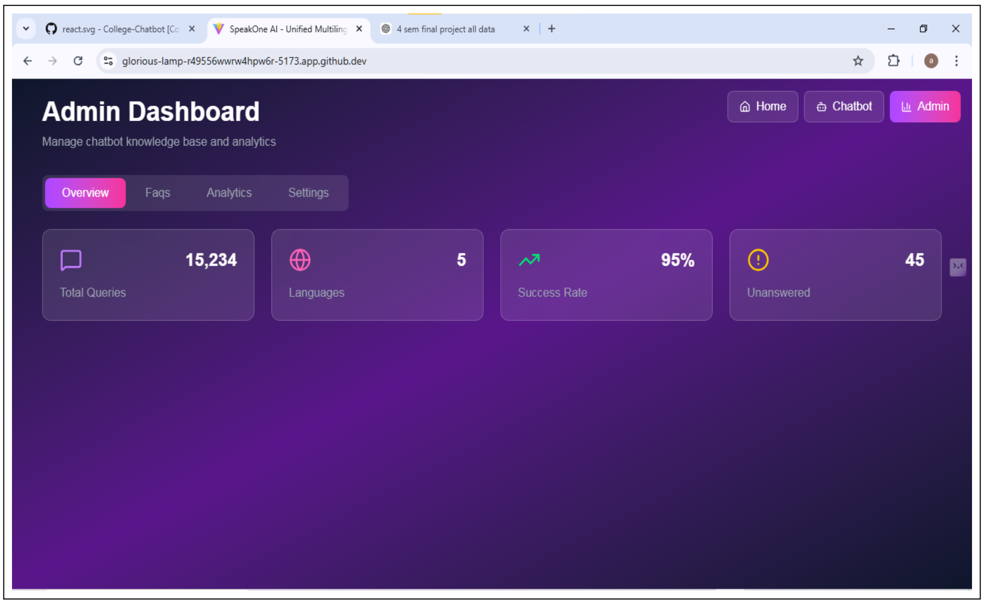

Figure 4. Administrative dashboard showing chatbot usage and system statistics

5.5 Query Handling Performance

The developed system is intended for processing frequent academic and administrative inquiries with minimal human input. The knowledge base functions as the main response provider for the predefined institutional information, whereas any unanswered inquiries are passed on to the external AI language model.

The system dashboard shows 15,234 total inquiries, which suggests that there is significant number of interactions in the developed application. The success rate of 95% represents the system-level success measure shown in the application. Nonetheless, it must be considered as the system statistic from the dashboard, and not the machine-learning accuracy measure, since the current paper does not show the test dataset.

Likewise, the dashboard contains 45 unanswered questions. Such questions denote cases in which the existing system could not give an adequate answer; hence, they present knowledge base augmentation possibilities.

5.6 Unanswered and Complex Queries

While the system is efficient in processing the queries for which it was intended, some limitations were noted during testing. In particular, mixed-language and complex queries are harder to process than direct queries based on predetermined intentions of the institution.

Mixed language queries could include vocabulary from different languages while complex queries could have multiple needs or intentions inside a single interaction. This could have implications for language recognition and intention detection.

The noted limitations imply that expanding the number of queries in the training/query database and better understanding of multilingual languages will prove useful in developing future versions of the system. The current paper proposes similar strategies to solve the problem.

5.7 Discussion

This demonstration serves to illustrate how the proposed hybrid chatbot architecture works functionally. It combines the use of a multi-language user interface, language detection, intent determination, institutional knowledge search, external AI fallback, response generation, logging of conversations, and administrative monitoring. The implemented interfaces presented in Figures 2–4 highlight the three basic layers of operation, which are user access, student interaction, and administration. It is evident that the architecture proposed can effectively handle multi-language student query interaction and provide response through a controlled path in the knowledge base. The external AI fallback allows providing more responses even when there are no pre-determined institutional answers available.

The administrative dashboard further serves as another illustration of operational use of the system by collecting system statistics and displaying them as query handling statistics. The administrative dashboard shows the following system indicators at the dashboard level: 15,234 total number of queries; five supported languages; a success-rate indicator set at 95%; and 45 unanswered queries. It is important to clarify the relationship between the success rate indicator shown on the dashboard and the unanswered queries since there is no information on whether these indicators refer to the same query population. Furthermore, there is a numerical relationship which needs to be checked prior to submitting the paper. The dashboard shows the following indicators: 15,234 total number of queries and 45 unanswered queries. If ”success” means ”not unanswered,” these numbers result in much more percentage than 95%. Hence, the authors need to clarify definitions of success rate, accuracy, and unanswered queries.

6. Conclusion

This research explains the development and deployment of the SpeakOne AI Unified Multilingual Chat Engine, a hybrid system that tries to automate common queries made by students in higher education institutions. This system tackles the difficulties associated with supporting multiple languages through automatic language identification and intent recognition using NLP. This system is a balance between institutional knowledge retrieval and AI fallback responses for external queries. According to the results of the deployed system dashboard, there were a total of 15,234 queries, with a 95% success-rate indicator; however, the reported test was done only on English, Hindi, and Marathi queries.

References

[1] Saranya, Dhulipala Saisubrahmanya, Dara Anirudh, Koppula Pooja Reddy, NSSS Girish Kumar, and M. A. Jabbar. ”TalkGenie: Intent and Sentiment-Aware Multilingual Chatbot.” In 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT), IEEE, 2025, 1-6.

[2] Saravana, M. K., Arman Mohammed, Sovan Pattanayak, Devendra Kumar Paswan, and Yuvraj Dadhich. ”ChatSense–A Multilingual Chatbot.” International Journal of Scientific Research in Science and Technology 12, no. 2 (2025): 1242-1250.

[3] Qin, Peinuan, Zicheng Zhu, Naomi Yamashita, Yitian Yang, Keita Suga, and Yi-Chieh Lee. ”AI-based Speaking Assistant: Supporting Non-Native Speakers' Speaking in Real-Time Multilingual Communication.” Proceedings of the ACM on Human-Computer Interaction 9, no. 7 (2025): 1-28.

[4] Rodríguez, Pablo Ramírez. ”Pedagogical Perspectives on Technology-Enhanced Phraseological Interpreting: A Case Study of Yandex Live Stream System.” Quaderns de Filologia-Estudis Lingüístics 30 (2025): 207-222.

[5] Liang, Tian, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. ”Encouraging Divergent Thinking in Large Language Models Through Multi-Agent Debate.” In Proceedings of the 2024 conference on empirical methods in natural language processing, 2024, 17889-17904.

[6] Priya, S. Sathiya, D. Manivannan, T. Kujani, and S. Bharathi. ”Cross-Language Connect: A Chat App for Smooth Multilingual Communication.” In 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), IEEE, 2024, 159-164.

[7] Kapse, Avinash S., and Vaishali Datta Parihar. ”Advanced Multilingual Chatbot for Indian Language Support.” Journal of Computer Science and Technology 5, no. 01 (2026): 478-491.

[8] Goutham, R., S. Hamjath, and N. Sabeer Ahamed. ”Real-Time Multilingual Chat Translator.” In 2026 8th International Conference on Intelligent Sustainable Systems (ICISS), IEEE, 2026, 1389-1395.

[9] Fabian, Pedregosa. ”Scikit-learn: Machine Learning in Python.” Journal of machine learning research 12 (2011): 2825.

[10] Mane, Deepak, Mayur Patil, Vrushali Chaudhari, Rutuja Nayakavadi, and Sanjana Pandhe. ”A Survey on Chatbot in Devanagari language.” In Proceedings of the 6th International Conference on Advance Computing and Intelligent Engineering: ICACIE 2021, Singapore: Springer Nature Singapore, 2022, 341-354.

[11] Suryawanshi, S., R. Itkarkar, and D. Mane. ”High Quality Text to Speech Synthesizer Using Phonetic Integration.” International Journal of Advenced Research in Electronics and Communication Engineering (IJARECE) Volume 3 (2014).

[12] Mishra, Ritwik, Simranjeet Singh, Jasmeet Kaur, Pushpendra Singh, and Rajiv Shah. ”Hindi Chatbot for Supporting Maternal and Child Health Related Queries in Rural India.” In Proceedings of the 5th Clinical Natural Language Processing Workshop, 2023, 69-77.

[13] George, Athira Susan, Gowri Muralikrishnan, Leya Rachel Ninan, Pranav S. Varrier, and L. K. Dhanya. ”Survey on the Design and Development of Indian Language Chatbots.” In 2021 International Conference on Communication, Control and Information Sciences (ICCISc), vol. 1, IEEE, 2021, 1-6.