Grenze International Journal of Engineering and Technology, January Issue

Citizen Helpline for Government Schemes and Services

Hiral Kawediya[1], Harsh Chaudhari[2], Tanushri Kankrej[3], Arhum Shah[4], Suhani Mahajan[5] and Jayashree Bagade[6]
[1-5]Department of CE, Software Engineering, Vishwakarma Institute of Technology, Pune, India
Email: harsh.chaudhari24@vit.edu, hiral.kawediya24@vit.edu, tanushri.kankrej24@vit.edu, arhum.shah24@vit.edu, suhani.mahajan241@vit.edu
[6]Department of Information Technology, Vishwakarma Institute of Technology, Pune, India
Email: jayashree.bagade@vit.edu

***Abstract*— This paper introduces an AI citizen helpline that aims to enhance the accessibility of government services by citizens. Spoken speech, natural language processing and a carefully mighty tuned language model trained on official scheme data are incorporated into the system. It aims at harmonizing information retrieval, minimizing user friction, and examining how automated conversational systems would contribute to better public service delivery and efficiency of governance.**

***Index Terms*— Artificial Intelligence, Citizen Services, Conversational AI, Natural language Processing, Voice-to-Voice.**

I. INTRODUCTION

It is challenging to access government schemes and various public services by many people even after their provision. Most people who are living in rural locations with no or minimal internet connectivity, or lack of technical expertise are being impacted. To communicate with government departments, it is possible to use online portals, mobile applications, and informational web sites, however, these devices demand constant internet connection, a smartphone, or at least the familiarity with digital interfaces. As a result, there is a fairly large percentage of the population that is not the rightful beneficiary of benefits and eligibility requirements, along with the application procedures of government welfare programs.
In most cases, traditional helpline services are highly inefficient, in the sense that they are manual, and can work only during fixed working hours and are often characterized by long waiting times as well as provision of unsuitable information. The restrictions listed above weaken the interaction of these services with the customers; there is a lack of transparency, which means that the number of individuals enjoying the fruits of the welfare schemes is reduced. There is a need to have low cost, easily accessible, multilingual, and automated systems to give reliable information on the most popular medium, voice calls.
Our solution to this problem is the Citizen Helpline System, which is an AI-powered application that combines VoIP telephony and Llama-based conversational agent locally deployed. In this manner, verified information on schemes in government is attainable by simply chatting in a normal tone to the agent. The system is entirely inclusive in language terms, it takes into consideration English, Hindu, and Marathi. The proposed system is also unlike the traditional helpline or a chatbot based on the cloud because it performs all the reasoning calculations locally, without any assistance of the cloud or the network, therefore, it is more secure and not reliant on the network connection to reach even remote locations. The system thus simplifies the effort of the users by sending an SMS of their discussion once a call is made.

Grenze ID: 01.GIJET.12.1.168

II. Literature Studies

The introduction of artificial intelligence into the operations of the government has played a significant role in the improvement of government services. These innovations lead to the acceleration of support of citizens, the possibility of accessing information in various languages, and the process of automatic search of information. There is also plenty of publications that indicate that generative AI and conversational interfaces can be the most important factors to redesign the services offered by the public sector and make their provision more natural and user friendly.

The examples given by Lothery et al. [5] on how generative AI may engage citizens in interacting more through their provision of flexible and engaging experiences, hence offering a platform on which AI can be used in governance, makes sense.

Similarly, Jayanti et al. [6], also, presenting his far-fetched illustrative instances, AI has not only facilitated the efficiency of the services but is in the same breath, the driver of the shift in consciousness of intelligent systems as the next wave of public administration.

Numerous reports are devoted to the possibility of using conversational agents which are able to assist the users in identifying whether they are eligible or not and finding their way around numerous government schemes. Indhumathi et al. [7] propose a chatbot-based solution based on the rules and NLP that will calculate the eligibility to the government financial programmes and offer a positive usability experience. Lee et al. [8] elaborated on the further integration of chatbots on the e-governance system. They emphasize better accessibility and faster response, but on the other hand, they admit that current systems do not work with unstructured voice queries and multilingual conversations.

What is more, information regarding the adoption of speech-related technologies to create AI-powered helplines is also highly represented in scientific literature. Among them is the article by Lawrence and Ling-Hwang [9] in which they discussed the efficiency of the services introduced by the current speech recognition tools in numerous sectors, and the field of governance is no exception. These applications, although, in most cases, rely on cloud-based APIs and that is the driving force behind their little application in regions with poor internet connection.

Conversely, the article by Smith et al. [10] that focuses on legal document processing, AI-aided, demonstrates that automated text-processing can be used as a means of reducing the bureaucratic complexity of particularly publicly oriented processes.

Whole articles by Brynjolfsson et al. [11] and Dash [12] clearly describe the topics of how the government is ready to employ AI-driven citizen services. They put emphasis on such problems as data privacy, user trust, and infrastructural issues. The authors all speak in one voice, with the desire to develop cost effective, inclusive and easily scalable solutions that may be effective in situations of multiple languages and limited resources.

A few recent research studies have highlighted this as a promising area for the use of Large Language Models (LLMs).

Gilang et al. [13] study the implementation of KemenkeuGPT into the Indonesian Ministry of Finance and find that the local LLMs can significantly influence the process of making decisions based on the evidence applied in deep technical domains.

Lin [14] posits that with this mechanism of generating models that are using AI, the government chatbots are likely to give bad information to the average citizen into believing that the level of engagement is on the increase. In addition to that, studies on call-center automation [15] and retrieval-based support systems based on the LLM can reduce the response time and increase the quality of services [16].

Nevertheless, these solutions are primarily based on cloud inference and not commonly optimized to support telecom-based interfaces.

The research on Indian government schemes (the example of the EPW reports) contributes important information on the complexity of the program eligibility rules and the necessity of the simplified access to the information [17]. Although much time has been spent on the creation of AI chatbots, speech systems, and LLM-based help, not many have thought of combining offline LLM inference, multilingual speech interaction, and telephony. Besides, in the previous systems, there are no options in the form of automated post-call summary and the possibility to provide on-demand support in poorly connected areas.

Thus, the examined resources define a clear research question, namely there are no existing solutions that will provide an entirely offline, multi-linguistic, voice-based AI-helpline to be accessed when a simple phone call is made, which, consequently, explains the novelty and the significance of the suggested Citizen Helpline framework.

III. SOLUTION METHODOLOGY

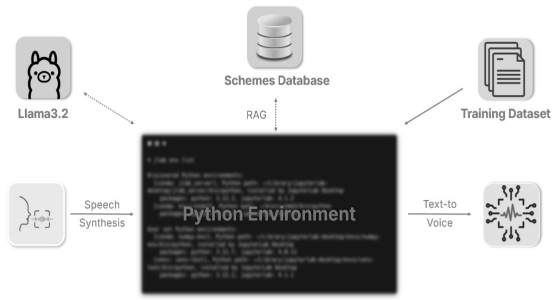

Figure 1. Overview of system components

This proposed Citizen Helpline System will give access to information on the government schemes in real-time on voice and is multilingual and supported by an offline conversational AI-driven agent. It is a localized system that combines both Llama model locally and a VoIP telephony interface to facilitate two-way natural communication without internet occurrences. It is divided into five main modules: VoIP Call Handling Layer, Speech Processing Layer, LLM-based Reasoning Engine, Response Generation Layer, and Post-Call Summarization Module. The general workflow is shown in Fig. 2.

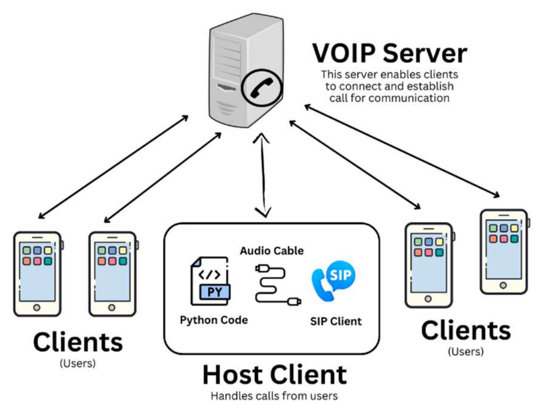

Figure 2. System architecture

A. System Architecture Overview

Essentially, the system works on a tightly integrated pipeline that incorporates a phone call of a user as an input and a generated summary by AI as an output that is provided after the session. The key components include

Asterisk VoIP Server

This server acts as a backbone for telephony. Users can connect with the helpline by dialing the helpline number over VoIP and connecting to this server. The server then routes the audio between the rest of the system.

Speech-to-Text (STT) Module

The module is executed using the Vosk offline ASR engine which can recognize speech in Hindi, English and Marathi and many more languages.

Local Llama Reasoning Engine

The primary concept of a local set-up is the locally fine-tuned Llama model quantized with GGUF inference that is optimized to be efficient to run on both CPU and GPU.

- This model interprets the natural language queries by users on government schemes.
- The model comprehends the given text besides retrieval in the datasets using language constructs.
- The system causes the network not to interrupt the running of the system, thereby enhancing security significantly, as well as enabling the user to get government services anywhere that he or she is located.
- This model is multilingual, so it can process English text as well as text in many other Indian languages.

Text-to-Speech (TTS) Module

Among the sources that offer this module is pyttsx3 that is utilized in a light and offline process.

- A solution to the eventual answer is finally provided through human voice simulation.
- The Asterisk audio path ensures short-time playback.

Conversation Logging & Summarization Module

- Detailed records, dialogue between two parties concerning the period during which a call is being made.
- After the discussion, the resultant logs are logged, and the resultant logs are subsequently summarized by the LLM using a prompt of how one wants to summarize the logs.
- Succinct is written in a language the conversation happened.
- Delivered to the caller in SMS/VoIP messaging API.

B. Detailed Workflow

Step 1: Call Initialization: Asterisk establishes a SIP session, and a user dials the support number. The system activates STT, TTS and the Llama inference engine. The Python backend receives the on-the-fly Webmaster packets at the time as a new audio packet.

Step 2: Speech Recognition & Language Identification: Vosk identifies user speech. A language identification (LID) module that identifies the language as Hindi, Marathi or English is also applied to the text obtained.

Step 3: Query Interpretation & Response Generation: The query is then input in the Llama model together with the domain-specific prompt template.

Step 4: Voice Response Playback: The response of the system is translated to the voice and language of the user with the help of pyttsx3. Sound is relayed back to the user using Asterisk which enables users to have comfortable and nearly delay free conversation flow.

Step 5: Conversation Summarization

When the user disconnects, the system:

- Compiles the dialogue log
- Uses Llama with a summarization prompt
- Translates the condensed summary
- Sends the final SMS through the messaging module.

C. Algorithmic Description

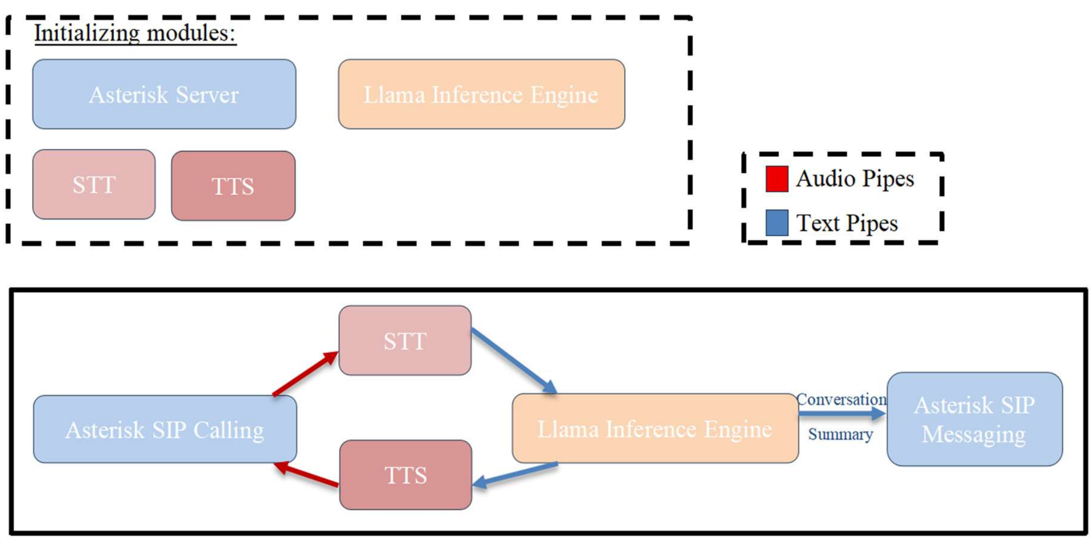

Figure 3. Architecture Diagram

D. Technical Advantages of the Proposed Architecture

Fully Offline Operation

- Ensures privacy (no cloud usage)
- Reliable for remote, low-bandwidth regions

Multilingual End-to-End Voice Interface

- Removes literacy barriers
- Supports broader citizen access

Modular & Scalable Architecture

- Components (STT, LLM, TTS) can be independently upgraded
- Supports future integration of more Indian languages

Automated Post-Call Knowledge Delivery

- Stores are necessary to store information.
- Improves usability by elderly or digital challenged users.

IV. RESULTS AND ANALYSIS

The performance of the Citizen Helpline system was evaluated using two key indicators including (i) end to end response latency. (ii) accuracy of speech recognition. To make real-time telephone-based conversational systems usable, the following metrics are required:

A. Speech Recognition Performance

There was the Speech-to-Text (STT) module that was deployed with the offline Vosk engine and achieved a Word. 10% error rate in testing in English, Hindi and Marathi voice queries. For telephony-grade voice applications, background noise, variation of accent, and compressed audio usually reduce STT. accuracy the WER of 10% is normally acceptable.

The obtained performance indicates that offline STT is sufficiently reliable for practical implementation when compared to industry standards, where the average telephone-line WER for multilingual audio ranges between 8 and 15%. Transcription errors were most common in:

- Proper nouns
- Region-specific scheme names
- Pronunciation variations in Hindi and Marathi

Because the Llama model frequently made up for these mistakes by deducing the intended meaning from incomplete text inputs, user comprehension remained high. Even in cases where upstream STT contains small errors, LLM-based semantic correction remains robust.

B. End-to-End Response Time

From the time a user speaks until they hear the synthesized response, the system showed an average response time of about ten seconds. Included in this total latency are:

- Speech-to-Text decoding (Vosk)
- Llama model inference for response generation
- Text-to-Speech output (pyttsx3)
- Minor processing delays in the VoIP audio pipeline

For an offline, multilingual conversational agent, the measured latency is acceptable, especially since all computation takes place locally without cloud acceleration. Such latency is similar to many real-world IVR systems and government helplines, where response wait times are often longer than five to fifteen seconds.

However, there were a few delays when dealing with:

- Long, multi-sentence user queries
- Domain-specific prompts requiring extended reasoning
- Tasks involving translation between Hindi/Marathi and English

In spite of this, there were no call drops or buffering delays, and user interactions continued to be clear and continuous. Response time may be further decreased by optimizing TTS speed and Llama inference efficiency (e.g., via quantization or GPU acceleration).

C. Overall System Behavior

The system successfully delivered:

- Multilingual discussions on the regular phone lines.
- Context-aware scheme advice using Llama reason.
- The speech feedback is consistent also in the background noise that is moderate.
- Automated conversation summaries delivered correctly after each call.

More importantly, since all the processing is done offline, the system was remarkably reliable even when not in operation of internet connectivity, which is essential to implement in underserved and rural regions.

TABLE I. RESULTS OBTAINED FROM INFERENCE

Metric	Description	Measured Result	Remarks
Speech Recognition Accuracy (WER)	Word Error Rate of Vosk STT across English, Hindi, and Marathi	10% WER	Acceptable for telephony audio; most errors in scheme names and accents
Average End-to-End Response Time	Time from user speaking to system replying	~10 seconds	Includes STT, Llama inference, translation, and TTS; suitable for offline IVR
STT Processing Time	Time for speech-to-text conversion	1–2 seconds	Varies with input length and background noise
Llama Inference Time	Time for generating textual response	6–7 seconds	Based on local CPU/GPU inference without cloud acceleration
TTS (Speech Output) Time	Conversion from generated text to audio playback	1 second	pyttsx3 generally low latency
Languages Supported	Native speech recognition + translated LLM reasoning	English, Hindi, Marathi	Enables multilingual citizen access
Conversation Summary Accuracy	Accuracy & relevance of LLM-generated SMS summary	High (manual verification)	Summaries captured essential points correctly

D. Discussion

The analysis shows some significant conclusions:

- Robustness: Llama model was able to recover user intent even with inaccurate STT, reducing the impact of minor transcription errors.
- Latency Trade-offs: A response time of approximately 10 seconds is a tradeoff between offline processing and accuracy. The need to sacrifice speed to gain privacy and accessibility is at the expense of the system to cloud-based chatbots.
- Multilingual Strength: The translation layers allowed Hindi and Marathi to be supported, reach to the users in a big way.
- Practical Deployment Suitability: Testing proves that the system is suitable to government. Where accuracy and clarity are more valued than very low latency, helpline settings.
- Recommendations on Improvement:

The following could improve performance:

- GPU BOOSTING FOR LLAMA
- QUICKER NEURAL TTS MODELS

V. FUTURE WORK

The viability of an independently deployable, multilingual, LLM-based conversational interface for public service delivery is evidenced by the proposed Citizen Helpline system. More work can be done on its performance, scalability and societal impact. The future work might focus on the following issues:

- Integration of Neural Speech Models: Substituting pyttsx3 and Vosk with lighter neural TTS and ASR models, like Whisper-lite, FastSpeech2, and VITS can significantly enhance the quality of speech, the strength of accents, and minimize latency.
- GPU-Optimized or Edge-Optimized Llama Deployment: The present-day inference is dependent on the CPU Quantized GPU engines such as TensorRT and can be used in the future GGUF and has backends based on a GPU, or an edge accelerator to reduce Llama response time to less than 7 seconds to under 2 seconds.
- Expansion to Additional Indian Languages: Adding Indian-language ASR/TTS models, including IndicSpeech and OpenSLR corpora, will make the system more inclusive and accessible to areas where Hindi is not spoken.
- Connectivity to Actual Government Databases: To verify scheme eligibility, application status or to assist users who require document submission facilities API secure or offline-cache can be enhanced.
- Quick-tempered Human-Agents and Smart Routing of Queries: To ensure accuracy but reduce the use of manual labour, improvement of future versions can be done by adding hybrid models where complex or ambiguous questions are relayed to the competent human operators.

- Improved Summarization and Record-Keeping: Citizen can be improved with multi-format outputs such as email summaries, WhatsApp messages or downloadable PDFs accessibility even more.
- Dashboards for Analytics and Administration: Creating dashboards for government administrators can assist in identifying scheme-awareness trends, regional service gaps, and frequently asked questions.
- Mobile App Integration: Although the system only uses voice now, a companion mobile app might enable chat and voice hybrid interaction, increasing accessibility for smartphone users.
- Improvements in Security and Compliance: In order to fully comply with data-protection frameworks in government deployments, future work can implement differential privacy, encrypted call logging, and zero-trust authentication.

Due to these advancements, the system will be enabled to evolve into a scalable digital-governance component suitable for statewide or national deployment.

VI. Conclusion

A fully offline multilingual Citizen Helpline was described above, capable of delivering information related to government schemes with the help of real-time voice interactions. The integration between the Asterisk VoIP server and Llama model running locally ensures privacy without any need for connectivity hence filling up a major lacuna present in current mechanisms of delivery of government services. English, Hindi, and Marathi speech interactions have been supported by the utilization of offline STT and TTS engines while leveraging AI-driven reasoning that returns context-aware responses. In addition, this system automatically summarizes a call which helps a user keep important details for future use.

Experimental results show about a 10% Word Error Rate in recognizing speech and about 10 seconds of end-to-end average latency, which validates the feasibility of deploying offline conversational AI in telephony environments. This is extremely constrained at the moment, but is a viable structure, in the direction of scaling citizen access to government welfare schemes even in the hinterlands where internet penetration also leaves much to be desired. In general, the suggested system is a chance at becoming more visible, open, and citizen-oriented governance. With future enhancements in speech models, wider language support, improved inference efficiency, and integration with real government databases, the system can significantly strengthen digital public service infrastructures.

References

[1] Official Python Documentation: https://docs.python.org/3/

[2] Python text to Speech Module Documentation: https://pypi.org/project/pyttsx3/

[3] Python Vosk Module Documentation: https://alphacephei.com/vosk/

[4] Asterisk VoIP Documentation: https://www.asterisk.org/get-started/

[5] EJ Lothery et al. "Generative Artificial Intelligence as an Enabler for Citizen Engagement," ProQuest (2024)

[6] P. Jayanti et al., “Transforming Citizen Services: How AI Can Empower Government Agencies,” Blue Whale App, 2024.

[7] Indhumathi et al., “AI-Based Chatbot for Government Scheme Eligibility,” Int. Res. J. Modernization in Engineering, Technology, and Science, 2024.

[8] Y. Lee et al., “Chatbots for E-Governance,” Smart Governance Review, 2024.

[9] R. Lawrence and J. Ling-Hwang, “Speech Recognition: A Game Changer,” 2024.

[10] Smith et al., “AI in Legal Document Processing,” Journal of Government Technology, 2023.

[11] E. Brynjolfsson, M. McAfee, H. Margetts, and S. Dorobantu, “Artificial Intelligence for Citizen Services and Government,” OECD/WEF Reports, 2024.

[12] J. Dash, “How AI Can Reform and Transform Citizen Service Delivery,” CSM Review, 2024.

[13] F. Gilang et al., “KemenkeuGPT: Leveraging a Large Language Model for Indonesia’s Government Financial Data,” 2024.

[14] Z. Lin, “Effective Use of Generative AI in Government Chatbots,” 2024.

[15] S. Patel et al., “Voice-Based AI in Call Center Customer Service: Evidence from a Field Experiment,” 2023.

[16] M. Singh and A. Verma, “LLM-Driven Chatbots for Retrieval-Based Government Services: A Technical Review,” 2023.

[17] “Government Schemes in India: A Comprehensive Overview,” Economic & Political Weekly (EPW), 2023.