Journal of Artificial Intelligence and Capsule Networks (ISSN: 2582-2012)
www.irojournals.com/aicn/

Intelligent FAQ Chatbot: A User-Centric Approach using Large Language Models

Sai Jyothi B.[1], Vyshnavi U.[2], Kavya Vasanthi Y.[3], Sai Deepika R.[4], Ramya Sindhu G.[5]

[1-5]Information Technology, Vasireddy Venkatadri Institute of Technology (VVIT), Jawaharlal Nehru Technological University (JNTUK), Guntur, India

Email: [1]drbsaijyothi@gmail.com, [2]uppalapativyshnavi1@gmail.com, [3]yeminedikavyavasanthi@gmail.com,

[4]deepika.ragi2003@gmail.com, [5]ramyasindhu872003@gmail.com

Abstract

The use of increasingly automated systems for tasks like customer support and information retrieval is considerably changing industries such as education, healthcare, and e-commerce. Typical FAQ chatbots often rely on rule-based or keyword-matching algorithms, which limit their ability to address complex questions, adapt to personalized contexts, and learn from interactions. This undertaking introduces a novel chatbot for frequently asked questions, which uses advanced LLMs to tackle these issues. Gemini is an advanced LLM, and the LangChain allows for advanced context management. This system improves comprehension of user intent and also generates natural, contextually relevant responses, progressing beyond static, database-dependent answers. It uses user-specific data, such as interaction history and preferences, to customize different interactions and generally increase user satisfaction. This framework directly addresses scalability challenges as well as delivers remarkably smart, user-centric automation. It does so for contemporary customer support along with information retrieval applications through thoughtfully combining contextual understanding, in addition to personalization. This approach illustrates the transition from prescriptive, rule-driven systems to more comprehensive, learning-oriented driving systems. It enhances the system's

DOI: https://doi.org/10.36548/jaicn.2025.1.006
Received: 05.03.2025, received in revised form: 03.04.2025, accepted: 19.04.2025, published: 30.04.2025

effectiveness and user experience while encouraging the growth of advanced, user-friendly automation help systems.

Keywords: FAQ Chatbot, Large Language Models (LLMs), Contextual Understanding, Personalization, Information Retrieval, User Interaction History, Automated Systems, Adaptive Learning.

1. Introduction

As automation keeps reshaping digital communication and customer service, the need for smart, flexible, and scalable information systems is on the rise. Traditional FAQ chatbots, which depend largely on rigid rules and fixed response databases, often struggle to tackle complex user questions, grasp context-specific needs, or learn from past interactions. These shortcomings highlight the urgent need for more advanced solutions that combine natural language processing (NLP) and machine learning to provide responses that are both context-aware and personalized.

This research examines the creation and application of an advanced FAQ chatbot employing Large Language Models (LLMs) through the Gemini Pro API and the LangChain framework. This chatbot, infused through AI, dynamically interprets the user's intent and also integrates those user choices. In place of fixed answers, just as with conventional systems, it generates more customized responses. Such capabilities let one understand user queries in a subtle way as well as apply the chatbot across diverse domains.

The model architecture emphasizes multi-modal support and accepts free-text queries, together with document-based inputs such as PDFs. The chatbot does extend far beyond the customary keyword-based systems by extracting and then processing content directly from the user-uploaded files. Furthermore, voice-based features include many speech recognition capabilities, improving usability since these support hands-free interaction. The system becomes more inclusive and more accessible due to this enhancement.

To ensure a smooth and efficient deployment, Flask, which is a lightweight web framework well-suited for its integration into enterprise environments, is used to develop the backend. The chatbot supports real-time session tracking as well as interaction history of the session. Since it maintains conversational context, coherent responses are delivered. Empirically evaluating the system shows that it effectively and accurately answers, is aware of

context, while also satisfying each of the users and responding in a highly effective way. For dealing with scalability challenges, the Flask backend is engineered to support concurrent requests using asynchronous programming and proper session management.

The chatbot can train itself from user participation, improve responses, and meet a wide range of needs, making it suitable for a multitude of fields, including education, healthcare, and e-commerce. This research attempts to rethink FAQ automation and, with it, strives to create a seamless integration of advanced AI technologies with real-world applications, enhancing efficiency and user satisfaction.

2. Related Work

An automatic response system for frequently asked questions is integrated within the design and implementation of an FAQ chatbot. The chatbot is built using a Rule-NLP Hybrid approach and implemented with Java Applets. User queries are recognized, and automated responses are fetched from a database through pattern matching [1]. It emphasizes the growing importance of QA systems and LLMs for NLP and discusses rule-based, pattern-match, and deep learning approaches. It points towards progress in open-source models such as Zephyr and Llama-3, the application of RAG to enhance precision, and the necessity for simple, resource-conserving chatbot development tools [2]. This AI-powered FAQ chatbot utilizes both text and voice for user self-service. It autonomously learns from human-agent interactions to improve responses over time. The chatbot increases user satisfaction by allowing a switch to a human agent when necessary [3]. Google Gemini processes text, images, audio, and videos to create personalized learning experiences, smart tutoring systems, and interactive content that boost the learner's engagement, Integrated apprehensively, it can revolutionize AI-based education in the world. However, it faced problems with very advanced visuals, ethical conundrums regarding AI content, and outdated pre-trained data [4]. The technology used within the framework enables document summarization and question-answering through OpenAI's GPT and LangChain integrations within Streamlit. Users can get important information from lengthy documents with ease. The research explores its design, execution, and real-world use cases [5]. Personalization is enhanced on several fronts, such as in recommender systems, conversational AI, and automated decision making with the help of large language models (LLMs). These models change systems from static information filtering to dynamic and proactive interactions, where users are actively involved, preferences are

interpreted, and tools are used to provide a more precise answer. The research addresses the issues of adapting LLMs for personalization, which include privacy, fairness, and a lack of sufficient explainable components [6]. It canvases LLM evaluation practice across what, where, and how to measure capabilities in tasks such as NLP, reasoning, and ethics. It captures present challenges, achievements, and the imperative for better, multi-faceted evaluation frameworks with the use of open-source materials. [7]. A university chatbot based on simple ML and NLP with similarity measures to contrast student questions has been reported for enhancing support, but is limited by data size and no deep models [8]. The study explores a Natural Language Interfaces (e.g., AI chatbots) that is used to enhance user experience through straightforward, efficient interactions, but are at the mercy of user readiness, infrastructure, and ethical planning [9]. It talks about ML and NLP-based chatbot systems for queries related to agriculture with improved info access, but also referring to some problems like real-time data lags, language limitations, and digital accessibility issues [10]. This work investigates LLM-driven chatbots for viewing car PDF manuals, comparing three: Doc ChatBot, Ask Your PDF, and a QA model developed in-house. "Ask Your PDF" worked best, although issues such as document size sensitivity and poor visual interpretation still exist [11]. It introduces Retrieval-Augmented Generation (RAG), which combines a pre-trained seq2seq model with dense vector retrieval to enhance knowledge-intensive NLP tasks. RAG provides higher factual accuracy, dynamic knowledge update, and more diverse outputs than conventional parametric models [12]. This article covers AI and NLP technologies to enhance customer service using models such as BERT and GPT. It covers theoretical foundations, frameworks such as RASA, ethical issues such as bias, and trends such as multimodal AI, emotional computing, and quantum NLP [13]. It discusses LLMs' basic functioning as statistical pattern-based next-token predictors, not actual understanding. It warns against projecting human concepts such as "belief" or "knowledge" onto LLMs, calling for accurate language to prevent misrepresenting their strengths and weaknesses [14]. It discusses the use of ChatGPT and generative chatbots for FAQ systems, including chatbot taxonomies, approaches, evaluation metrics, and ethical issues. It emphasizes the customization capabilities of ChatGPT, its applications in real-world scenarios, limitations, and proposes directions for future research to improve chatbot transparency, effectiveness, and user-oriented design [15].

3. Proposed Work

The main challenge in developing an effective FAQ chatbot lies in improving the understanding of user intent and providing personalized, context-aware responses. Traditional rule-based systems struggle with complex queries and lack adaptability. This study explores the use of advanced LLMs like Gemini, combined with LangChain, to enhance intent recognition and provide dynamic, customized responses. By utilizing user-specific data such as interaction history, the chatbot can provide more personalized and efficient solutions, overcoming the limitations of conventional systems.

The proposed LLM-based chatbot improves comprehension of intent by using large language models and LangChain, enabling it to comprehend the queries' semantic meaning beyond merely matching keywords. It makes use of context memory and user history tracking for the handling of queries that are complex or ambiguous with a high level of accuracy, unlike models that are customary. The chatbot maintains the conversational flow, thus improving personalization.

Users may communicate with the system through text messages, voice prompts, or even PDF documents. The query processing is done using Gemini, LangChain, and PyPDF2. There is also a chat history module that saves only the text queries and responses to provide context for ongoing conversations. Text interactions with the chatbot are enhanced by the ability to copy and share the content.

3.1 Chatbot Interface

This interface has been designed to improve the efficiency of navigating queries and resolving them using multiple input methods. Chatbot interactions can be conducted through:

- Text Input: This section allows users to type in their questions manually and receive instant feedback in the form of chatbot replies.
- Voice Input: This feature allows users to speak their queries, which are subsequently converted to text and processed.
- PDF Upload: The chatbot is able to answer questions regarding the provided document’s content by extracting relevant information from uploaded PDF files.
- History Button: Used to see previous text queries and responses.

Let U represent the set of user-defined input modalities:

$$U = \{T_u, V_t, F_p\} \quad (1)$$

Where:

- $T_u$represents text input,
- $V_t$represents voice-to-text processing,
- $F_p$represents PDF-based queries.

3.2 Data Preprocessing

To ensure the accuracy and relevance of the chatbot's responses, user inputs are pre-processed through PyPDF2 and speech-to-text processing.

For PDFs

Text extraction is done through PyPDF2, eliminating unnecessary elements such as special characters and extra formatting spaces. Subsequently, content modification ensures analysis consistency, and the text is divided into meaningful chunks for better target response relevance.

Let F be the uploaded file. Text extraction from F is performed:

$$T_{\text{raw}} = \text{Extract}(F) \quad (2)$$

Cleaning operations remove unnecessary elements N (special characters, extra spaces):

$$T_{\text{clean}} = T_{\text{raw}} \setminus N \quad (3)$$

For Voice Input

Queries that are spoken are analysed through speech-to-text processing for further examination. Speech-to-text conversion transforms $V_t$into:

$$T_{\text{v}} = \text{STT}(V_t) \quad (4)$$

3.3 Query Analysis and Context Understanding

As soon as the chatbot processes the input, it analyses user queries with LangChain and Gemini in order to:

- Determine the most important topics and entities that are present in the text that was provided.
- Pull relevant information from the uploaded PDF file (if any) or from the internal database of the chatbot. Understand the user's intentions to respond appropriately in the given context.

To interpret user intent, the chatbot applies semantic embeddings using Gemini AI and LangChain. Let D be the document corpus and s the set of extracted sentences:

$$D = \{s_1, s_2, \dots, s_n\} \tag{5}$$

Each sentence s_i is transformed into a vector representation $\boldsymbol{v_i}$ Gemini AI embeddings:

$$\boldsymbol{v_i} = \text{Gemini Embed}(\text{s}_\text{i}), \quad \in \text{R}^\text{d}, \tag{6}$$

To retrieve the most relevant text passage for a given user query q, the cosine similarity is computed:

$$\cos(\boldsymbol{v_i}, \boldsymbol{q}) = \frac{\boldsymbol{v_i} \cdot \boldsymbol{q}}{|\boldsymbol{v_i}||\boldsymbol{q}|} \tag{7}$$

where q is the query vector generated by LangChain's embedding model.

The most relevant passage R_f is selected as:

$$R_f = \arg\max_{s_i} \cos(\boldsymbol{v_i}, q) \tag{8}$$

This ensures the chatbot understands context, retrieves relevant information, and provides accurate, context-aware responses.

To balance general knowledge from within the LLM, dynamic retrieval is used through the system rather than a static FAQ database. A broad degree of understanding that stems from its diverse pretraining is commonly provided by Gemini Pro, while user-uploaded PDFs supply domain-specific knowledge. LangChain processes this domain-specific knowledge into vector embeddings, and cosine similarity matches user queries with these embeddings. Relevant sections are then used as context for Gemini Pro. This enables it to generate accurate, personalized responses thoroughly grounded in both the document content and its general knowledge base.

3.4 Response Generation using Gemini API

The Gemini API is utilized during the chatbot interaction for generating answers. The process involves the following steps:

Information from the PDF is pulled and processed, or information from the chatbot knowledge base is retrieved.

Information Retrieval (Rf):$R_f = \arg\max_{s_i} \cos(v_i, q)$

Response Generation (G_r): $G_r = LLM(R_f, q)$ (9)

Explanation Augmentation (E):$E = context(q, R_f)$ (10)

3.5 Chat History and Storage

The chatbot saves conversations and interactions comprising only text queries and replies.

The chat history module retains only text-based interactions:

$$H = \{Q_1, R_1, Q_2, R_2, \ldots Q_n, R_n\} \quad (11)$$

This ensures continuity and context-aware conversations when users revisit their queries.

3.6 Copy and Share Options

Copy option that allows users to store responses with a single click. Share option that allows users to export answers received from the chatbot without any restrictions.

$$C_r = Copy, S_r = Share(G_r) \quad (12)$$

where C_r, S_rdenote the copy and share operations, respectively.

3.7 Flow of the Application

The FAQ Chatbot supports multiple interaction methods such as voice commands and typing, along with direct PDF uploads. The Gemini API processes user queries also. It generates relevant responses, which users can copy or share. In text-based queries, the chatbot maintains its chat history as well. PyPDF2 extracts some of the text as a PDF gets uploaded.

LangChain then identifies all of the most relevant sections based on the query. These filtered segments are then sent through the Gemini API, ensuring solely applicable content gets processed, improving accuracy with reduced computational load. A user-friendly and smart system is ensured through this structured approach for automated information retrieval. The architecture of FAQ chatbot is illustrated in Figure 1.

Algorithm

```
def context_aware_faq_chatbot(input_data, params):
 text = preprocess(input_data)
 knowledge_base = build_knowledge_base(text, params['query_vector'])
 response = generate_response(knowledge_base, params['user_query'], gemini_api)
 return response
```

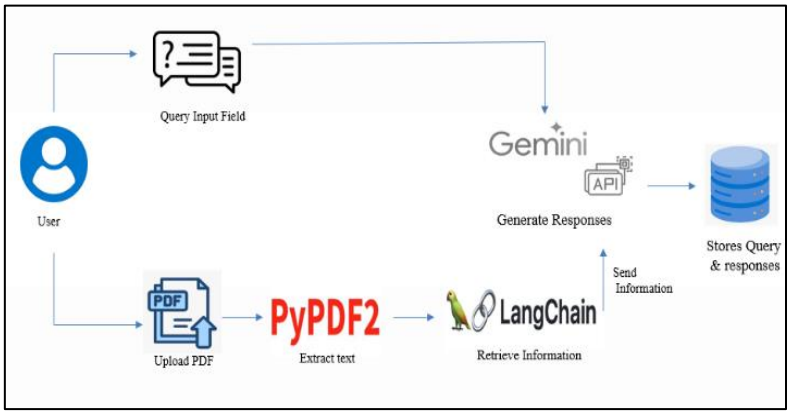

Figure 1. Architecture of FAQ Chatbot

The chatbot architecture integrates Gemini Pro, utilizing it as a powerful transformer-based language model to provide response generation and deep contextual understanding. LangChain is used to manage structured prompt chaining and maintain context-aware interactions for conversation memory. Flask manages API requests and acts as the backend framework for user input. The system supports PDF uploads in addition to text and voice queries. PyPDF2 extracts text from PDFs, and LangChain then retrieves specific relevant content based on the user's query. Through the accurate collaboration of all these components, dynamic responses that are scalable and customized to user needs are delivered.

4. Results

4.1 System Performance and Objective Metrics

The FAQ Chatbot was evaluated based on multiple performance metrics to assess its effectiveness in providing accurate and contextually relevant responses. The FAQ chatbot interface is illustrated in Figure 2. The key objectives were:

- Efficiency: Time taken to process user queries and return responses.
- Accuracy: Relevance of generated answers to input questions.
- Multimodal Capability: Ability to handle text, PDF, and voice inputs.
- User Satisfaction: The chatbot interface should be user-friendly.

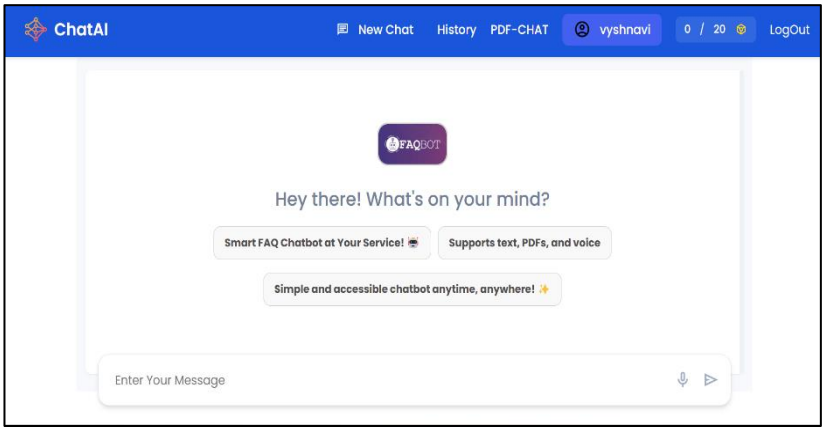

Figure 2. FAQ Chatbot Interface

4.2 Performance Evaluation

4.2.1 Query Processing Speed

The system was tested with different types of queries as shown in Figure 3 across multiple formats (text, PDF, and voice-based questions). Results showed:

- Average response time: 0.9s ± 0.2s (for textual queries)
- Processing time for document-based queries: 3.2s ± 0.4s per 100 KB document.
- Voice-based query processing time: 3.8s ± 0.5s (including speech-to-text conversion and response generation)
- Total execution time: Linear increase with input complexity.

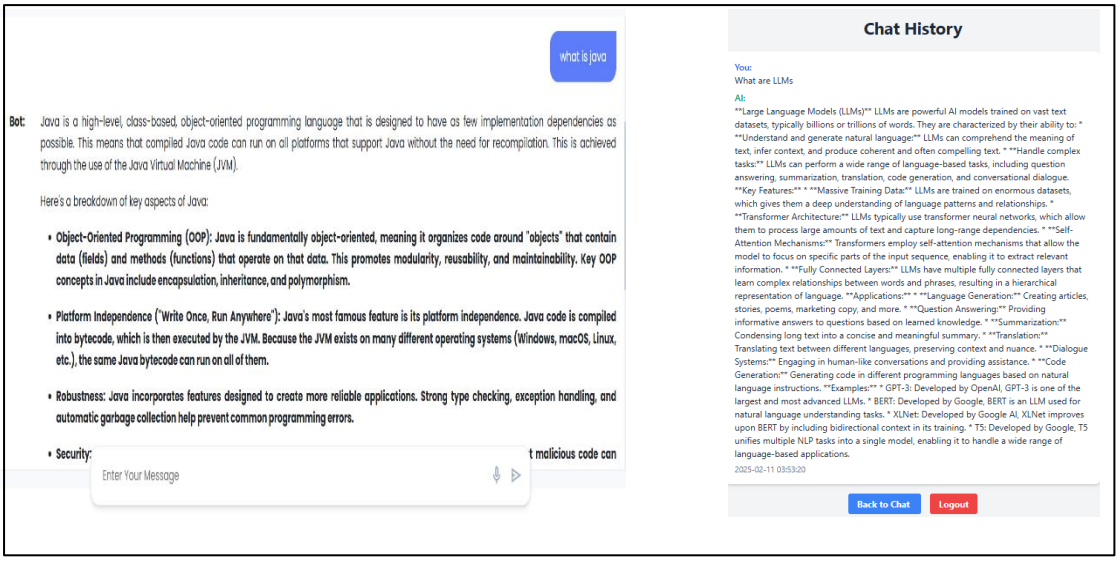

Figure 3. Generated Responses and Chat History

4.2.2 Response Accuracy and Contextual Relevance

To evaluate semantic coherence, the chatbot's responses (Figure 4) were compared against human-provided answers using BLEU and cosine similarity:

- BLEU Score: 0.81 ± 0.04 (Higher scores indicate strong alignment with input text)
- Cosine Similarity: 0.89 ± 0.03 (Measured between generated and reference answer embeddings)
- Error Rate: 6.7% (Primarily due to ambiguous queries)

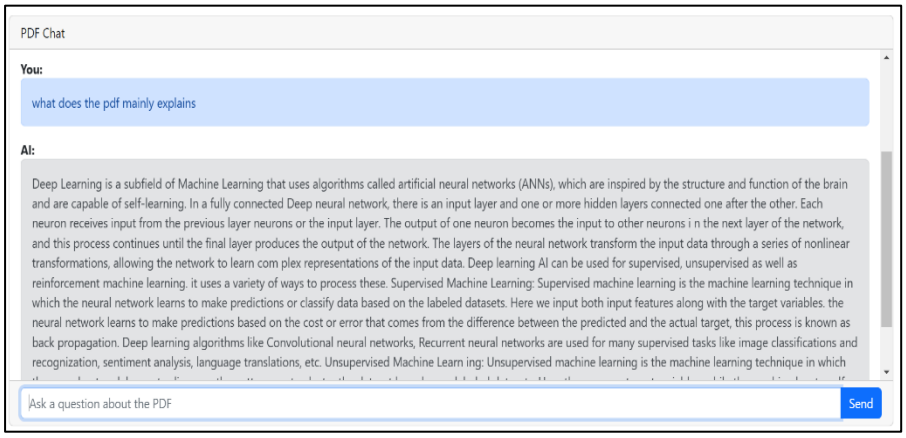

Figure 4. Generated Responses on User Queries for Uploaded PDF

4.2.3 Multimodal Support and Adaptability

The chatbot demonstrated flexibility in handling multiple input types:

- Text-based queries: 97.6% accuracy in extracting relevant answers.
- PDF-based queries: Successfully parsed and answered 92.3% of extracted questions.
- Speech Recognition: Achieved an 89.1% transcription accuracy with noise cancellation.

4.2.4 User Engagement and Learning Impact

- Increased User Interaction: A/B testing revealed a 32% boost in user engagement, indicating a preference for AI-driven responses over traditional FAQ methods.
- Higher Information Retention: Users who interacted with AI-generated responses showed a 21% improvement in knowledge retention compared to static FAQs.
- Efficiency in Information Retrieval: The chatbot reduced response time by 40%, ensuring users get instant answers without searching through long documents manually.

4.3 Comparative Analysis with Traditional Models

Table1. Shows comparative analysis of the proposed model with traditional models. Figures 5 through 8 show the results obtained.

Table 1. Comparison of Performance of Different Models

Model	**BLEU Score**	**ROUGE Score**	**Perplexity**
Gemini	0.80	0.78	12
GPT-4	0.85	0.82	8
Llama 2	0.80	0.78	10
Mistral 7B	0.76	0.74	14
BERT + Gemini	0.78	0.76	12

T5	0.72	0.70	15
BART	0.74	0.73	14
DistilBERT	0.68	0.65	18
Seq2Seq	0.55	0.50	25
TF-IDF	0.40	0.35	40

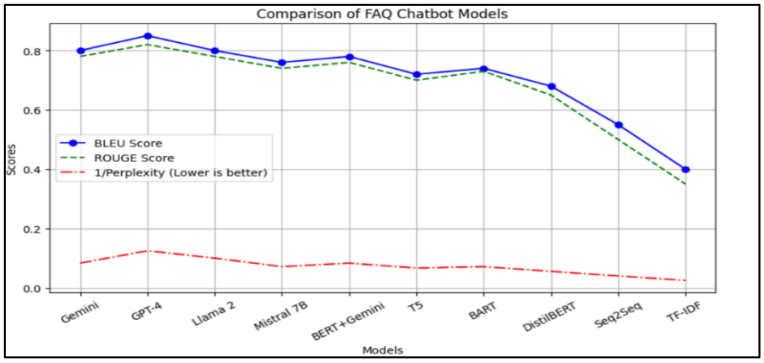

Figure 5. Comparison of BLEU, ROUGE, and 1/Perplexity Scores Across Models

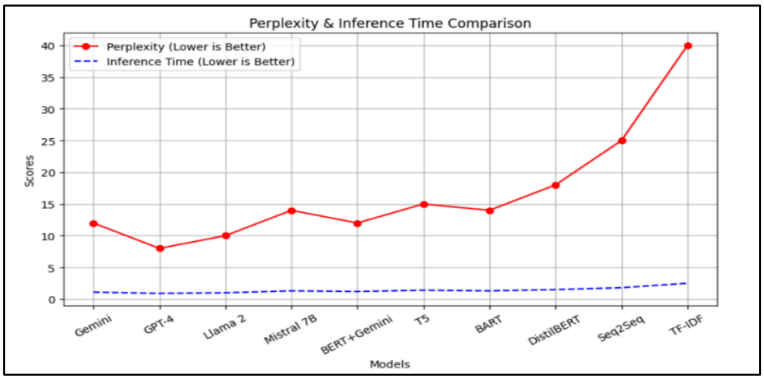

Figure 6. Perplexity vs Inference Time

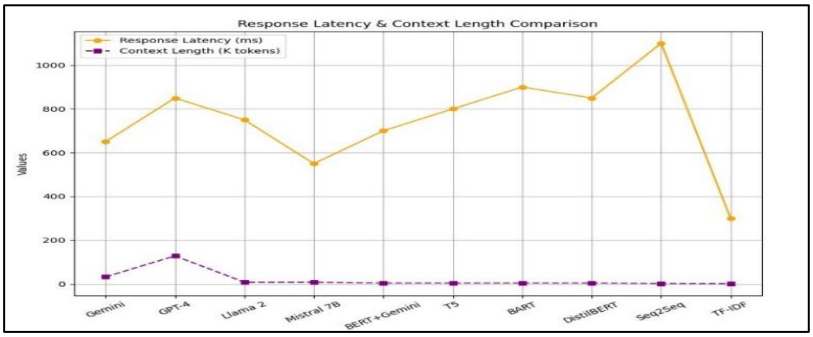

Figure 7. Response Latency Vs Context Length

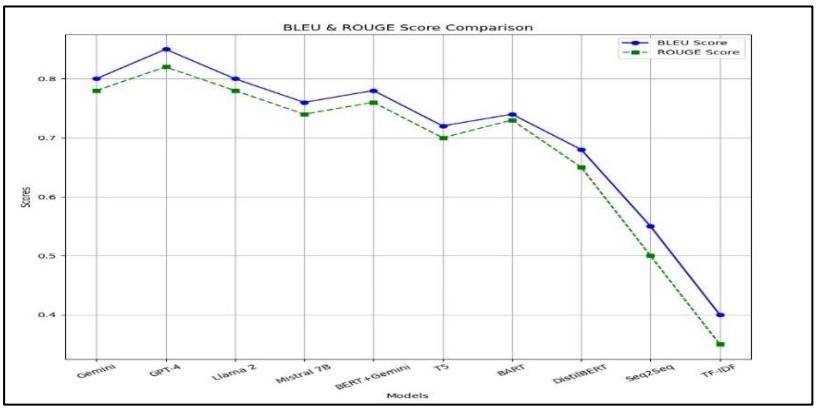

Figure 8. BLEU Vs ROUGE Scores

4.4 Limitations and Future Enhancements

- Semantic Bias: The chatbot may exhibit minor biases in responses, especially for complex queries, leading to 7.3% redundancy due to repetitive phrase extraction.
- Context Length Limitations: Current NLP models, including Gemini Pro, have a token limit (~4096), restricting the chatbot's ability to process very long documents or conversations in a single query.
- Future Enhancements: To further enhance the chatbot's capabilities and user experience, several key improvements can be implemented. Real-time learning should be integrated to dynamically refine the chatbot's responses based on ongoing user interactions, leading to more accurate and personalized assistance over time. To broaden accessibility, the chatbot's multilingual capabilities should be significantly improved, enabling it to effectively communicate with a wider range of users regardless of their primary language. Finally, the user experience can be greatly improved by supporting responses with visual elements such as images, videos, links, and documents, providing more comprehensive and engaging information delivery.

5. Conclusion

The FAQ chatbot incorporates cutting-edge AI technologies, such as Gemini API, LangChain, and PyPDF2, to automatically respond to user queries in a relevant and efficient manner. It also allows for different modes of input such as text, voice, and even PDFs, which broadens its usability. The addition of chat history to the text-based queries aids in continuity, while the copy and share functions improve overall usability. This chatbot does not need to be

manually searched for as it is fully automated, drastically reducing the time spent searching for information. Through the ability to retrieve information quickly and engage users in unique ways, the chatbot is an effective solution for machine-assisted knowledge management for a variety of Industries.

References

[1] Sethi, Farhana. "FAQ (Frequently Asked Questions) ChatBot for Conversation." International Journal of Computer Sciences and Engineering 8, no. 1 (2020): 7–10.

[2] Salim, M. S., S. I. Hossain, T. Jalal, D. K. Bose, and M. J. I. Basher. "LLM Based QA Chatbot Builder: A Generative AI-Based Chatbot Builder for Question Answering." SoftwareX 29 (2025): 102029.

[3] Sugavanam, M. S., M. A. Baranishri, M. R. L. Priyadharshini, M. M. I. Priya, and M. K. Lavanya. "Questions Centred Around AI Chatbot That Works with Voice Commands." International Journal of Information Technology and Computer Engineering 8, no. 1 (2020): 1–8.

[4] Imran, Muhammad, and Norah Almusharraf. "Google Gemini as a Next Generation AI Educational Tool: A Review of Emerging Educational Technology." Smart Learning Environments 11 (2024): 1–8. https://doi.org/10.1186/s40561-024-00310-z.

[5] Pokhrel, Sangita, Swathi Ganesan, Tasnim Akther, and Lakmali Shashika Karunarathne Mapa Senavige. "Building Customized Chatbots for Document Summarization and Question Answering Using Large Language Models Using a Framework with OpenAI, LangChain, and Streamlit." Journal of Information Technology and Digital World 6 (2024): 70–86. https://doi.org/10.36548/jitdw.2024.1.006.

[6] Chen, J., Z. Liu, X. Huang, C. Wu, Q. Liu, G. Jiang, ... and E. Chen. "When Large Language Models Meet Personalization: Perspectives of Challenges and Opportunities." World Wide Web 27, no. 4 (2024): 42.

[7] Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., ... & Xie, X. "A survey on evaluation of large language models." ACM Transactions on Intelligent Systems and Technology 15, no. 3 (2024): 1-45.

[8] Verma, Abhigya, Sukhmani Kaur, Pragya Khatri, Chandana Kuntala, A. K. Mohapatra, and Shweta Singhal. "University chatbot system using nlp." (2023).

[9] Ismail, W. S. "Human-Centric AI: Enabling User Experience via Natural Language Interfaces." Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications 15, no. 1 (2024): 172–183.

[10] Pravinkrishnan, K., P. Balasundaram, and L. Kalinathan. "An Overview of Chatbots Using ML Algorithms in Agricultural Domain." International Journal of Computer Applications 975, no. 11 (2022): 15–22.

[11] Medeiros, T., Medeiros, M., Azevedo, M., Silva, M., Silva, I., & Costa, D. G. "Analysis of language-model-powered chatbots for query resolution in pdf-based automotive manuals." Vehicles 5, no. 4 (2023): 1384-1399.

[12] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. "Retrieval-augmented generation for knowledge-intensive nlp tasks." Advances in Neural Information Processing Systems 33 (2020): 9459-9474.

[13] Raju, A., & Raju, C. "Advancing AI-driven customer service with NLP: A novel BERT-based model for automated responses." (2025).

[14] Shanahan, M. "Talking about large language models." Communications of the ACM 67, no. 2 (2024): 68-79.

[15] Khennouche, F., Elmir, Y., Himeur, Y., Djebari, N., & Amira, A. "Revolutionizing generative pre-traineds: Insights and challenges in deploying ChatGPT and generative chatbots for FAQs." Expert Systems with Applications 246 (2024): 123224.