International Journal of Electrical and Computer Engineering (IJECE)
Vol. 16, No. 2, April 2026, pp. 1002~1009
ISSN: 2088-8708, DOI: 10.11591/ijece.v16i2.pp1002-1009

Development of BAPOLAIC: AI chatbot for optical character recognition based-document extraction and voice assistant

Ryan Satria Wijaya, Rival Fahreji
Department of Robotics Engineering, Major of Electrical Engineering, Politeknik Negeri Batam, Batam, Indonesia

Article Info

Article history:

Received Apr 21, 2025
Revised Jan 2, 2026
Accepted Jan 16, 2026

Keywords:

AI chatbot
Gemini API
Natural language processing
Optical character recognition
Voice assistant

ABSTRACT

Conventional chatbots often lack integrated functionalities for complex academic tasks, such as multi-format document handling and multimodal interaction. This paper presents the design, implementation, and performance evaluation of BAPOLAIC, a web-based, multimodal AI assistant developed to address this gap. The system architecture integrates optical character recognition (OCR), a dual-strategy natural language processing (NLP) module, and voice assistance, all orchestrated by the Gemini API. Quantitative evaluation confirmed high performance: the OCR module achieved a 98.69% average accuracy, and the retrieval-based NLP path correctly handled 90% of test queries. Furthermore, the API integration demonstrated exceptional efficiency with a median latency as low as 0.06 ms. Task-based evaluations validated BAPOLAIC's effectiveness in performing intelligent functions like summarization and content-based Q&A, with a superior capacity for handling up to 10 consecutive documents. The results validate BAPOLAIC as a successful proof-of-concept for a specialized academic tool, providing a framework for integrating multiple AI technologies to enhance educational productivity.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Ryan Satria Wijaya
Department of Robotics Engineering, Major of Electrical Engineering, Politeknik Negeri Batam
Batam, Indonesia
Email: ryan@polibatam.ac.id

1. INTRODUCTION

Educational institutions are currently undergoing a significant digital transition, with AI-based chatbots playing a pivotal role in enhancing information management efficiency [1]–[3]. The evolution of these systems has progressed from simple rule-based models to complex conversational assistants capable of context understanding via natural language processing (NLP) [4]–[6]. Modern implementations increasingly leverage synergistic technologies, including optical character recognition (OCR) for image data extraction and Web Speech APIs for natural voice interactivity [6]–[8] Creating vast opportunities for dynamic and accessible academic tools.

Despite the expansion of chatbot research in academic contexts, efforts remain fragmented. Existing studies often focus on isolated tasks, such as supporting self-regulated learning [9], addressing administrative FAQs [10], [11], or assisting visually impaired users through discrete OCR integration [12]. While the concept of "document conversations" allows interaction with content [13], most systems are confined to digital text processing and lack robust voice command capabilities. Consequently, a clear research gap exists for an integrated platform capable of: i) processing diverse physical and digital formats, including handwriting, ii) providing seamless multimodal voice-text interaction, and iii) offering intelligent analysis such as summarization within a single cohesive system. This study addresses this gap by proposing Batam

Polytechnic AI Chatbot or BAPOLAIC, a versatile web-based assistant. The primary contribution of this research is a novel system architecture that seamlessly orchestrates Tesseract OCR [14], NLP, Web Speech API, and the Gemini API to develop a comprehensive AI assistant [15]. Unlike conventional chatbots, BAPOLAIC functions as a central processing brain that extracts information from printed documents and images, ensuring accessibility and accurate responsiveness. This paper details the system's development methodology and quantitative evaluation, offering a framework for future document management solutions in education.

2. METHOD

This study applied a design and creation methodology to the systematic development and quantitative evaluation of the BAPOLAIC system [16], [17]. To ensure a structured and rigorous approach, the research process was executed in four sequential phases. It commenced with requirement planning and literature review, where the research gap and system functions were defined, followed by system design and architecture, covering both the high-level and component-level design of the platform. The third phase is implementation and modular development, constituted the core development stage of the prototype. Finally, the process concluded with testing and quantitative evaluation, where each module's performance was quantitatively assessed against predefined metrics. The subsequent sections detail the technical implementation of the platform.

2.1. System architecture

The foundation of BAPOLAIC is a modular three-tier client-server architecture that strictly separates the presentation layer, application logic, and external services to facilitate scalability [18]. As illustrated in Figure 1, the operational flow begins on the client frontend, developed using HTML, CSS, and JavaScript, which serves as the primary interface for user interaction. Crucially, to support multimodal functionality, the frontend utilizes the browser's native Web Speech API to capture and transcribe voice commands on the client-side before transmission, thereby reducing server latency. The inputs are routed to the backend server [19]. powered by a Python Flask micro-framework, which orchestrates the core business logic. Depending on the input type, the Flask application directs data to specific internal modules: the OCR module (Tesseract and OpenCV) for document digitization, the NLP module for intent analysis, or the voice module for generating audio responses. For complex cognitive tasks, the backend acts as an API layer, formatting pre-processed data into structured prompts and transmitting them via secure HTTPS requests to the external Gemini API [20]. Upon receiving the generative response, the backend processes the text and, if required, utilizes the google text-to-speech (gTTS) library to synthesize audio, delivering a complete multimodal response back to the user.

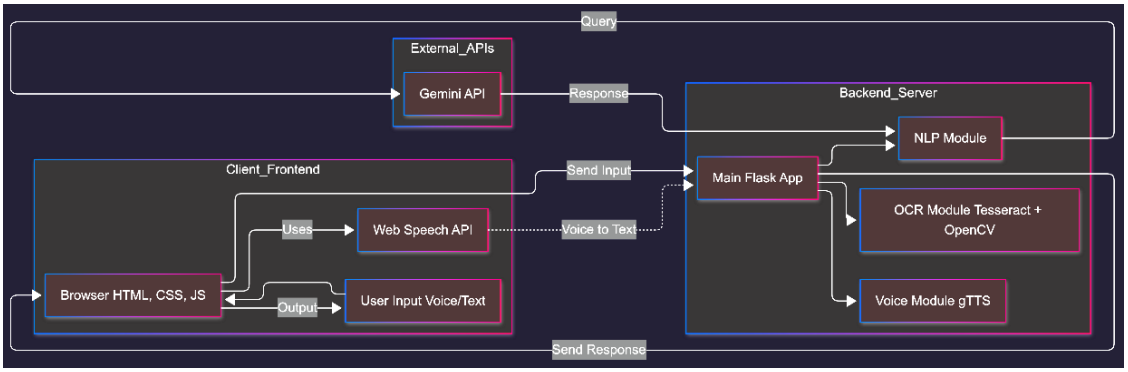

Figure 1. BAPOLAIC system architecture diagram

2.2. Implementation of core technologies

The system implementation integrates four critical modules to enable seamless multimodal interaction. First, the OCR module, following the pipeline illustrated in Figure 2, functions as the primary data ingestion point for digitizing text from diverse formats, including PDFs and handwriting. It utilizes the Tesseract OCR engine (v5.4.0) implemented via Python's *pytesseract* library, selected for its robust accuracy on printed text [21]. To maximize recognition quality, a mandatory pre-processing pipeline using OpenCV executes grayscaling, binarization, noise reduction, and deskewing prior to extraction [22]. Following

ingestion, the NLP module processes the text using a dual-strategy approach as depicted in Figure 3. For standard FAQs, a retrieval-based path employs the cosine similarity algorithm (1) to match user queries against a TF-IDF vectorized knowledge base [23]–[26].

$$Cosine\ Similarity(A,B) = cos(\theta) = \frac{A.B}{\|A\|\|B\|} = \frac{\sum_{n=1}^{i} A_i B_i}{\sqrt{\sum_{n=1}^{i} A_i^1}\ \sqrt{\sum_{n=1}^{i} B_i^2}} \quad (1)$$

For complex reasoning tasks, such as document summarization or open-ended Q&A, the system activates a generative path powered by the Gemini API [27]. In this workflow, the backend dynamically constructs a structured prompt containing the user's query and the OCR-extracted context [28], transmitting it via secure HTTPS requests to the *Gemini-pro* model for deep semantic analysis [29], [30]. Finally, the voice interaction module manages the audio interface. It utilizes a client-server architecture where the browser's Web Speech API performs real-time speech-to-text (STT) on the client side to minimize latency [31], while the server-side Python *gTTS (Google Text-to-Speech)* library converts the AI's textual response into synthesized speech for playback [32].

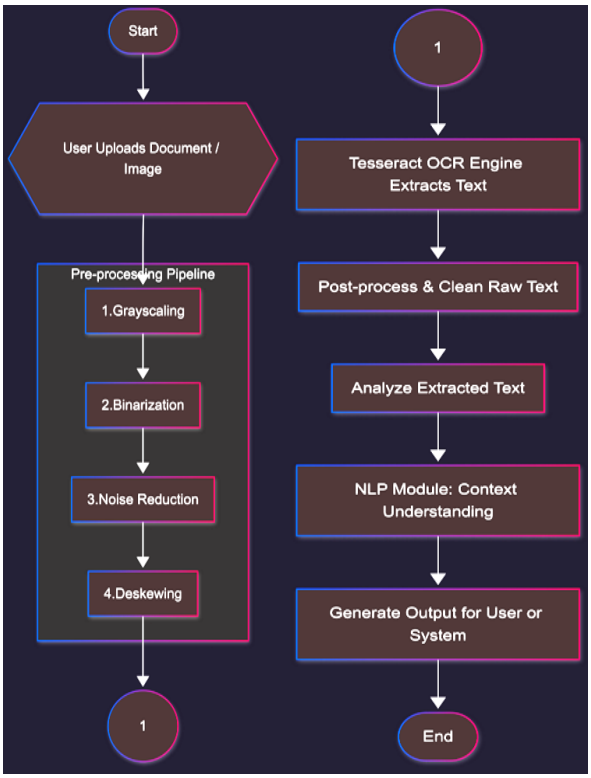

Figure 2. OCR module processing pipeline

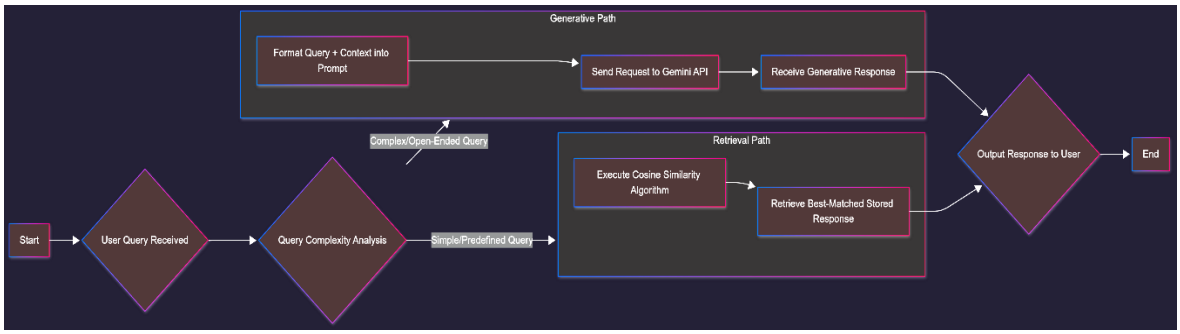

Figure 3. Flowchart of the NLP dual-strategy approach

3. RESULTS AND DISCUSSION

This chapter presents a comprehensive performance evaluation of the BAPOLAIC system, structured to empirically validate each of the core contributions outlined in the introduction. The analysis begins with a quantitative assessment of the foundational modules, including the OCR accuracy, the NLP module's reliability, and the Gemini API integration's performance. Following this, a task-based evaluation of the system's advanced intelligent and multimodal functionalities is presented to demonstrate its practical capabilities. The chapter then contextualizes these findings through a comparative analysis against an industry benchmark, and concludes with an overall discussion of the results, the study's limitations, and directions for future work.

3.1. Performance evaluation of the OCR module

The OCR module's accuracy was quantified by comparing extracted text against manually verified ground truth data across diverse document formats. Performance was measured using the character error rate (CER) calculated as the Levenshtein distance representing the sum of substitutions (S), deletions (D), and insertions (I)—divided by the total number of characters in the ground truth text (N), as shown in (2). The final accuracy percentage was then derived from the CER, as defined in (3).

$$CER = \frac{S+D+I}{N} \quad (2)$$

$$Accuracy = (1 - \text{CER} \times 100\%) \quad (3)$$

The raw text output from the Tesseract module was captured *before* any processing by the Gemini API. A hypothetical raw output with minor deviations could be: "*...mutasi dalm Algoritma Genetka...*". Comparing this raw output to the ground truth reveals two-character errors (a deletion in "dalm" and a substitution in "Genetka"), resulting in a Levenshtein distance of 2. Applying (2) and (3):

$$CER = \frac{2}{206} = 0.0097$$

$$Accuracy = (1 - 0.0097 \times 100\%) = 99.03\%$$

As detailed in Table 1, the module demonstrated robust performance with an aggregate average accuracy of 98.69%. The highest accuracy of 99.76% was achieved on printed PDF documents, while handwritten text yielded a competitive 99.63% accuracy despite its inherent morphological complexity. These results confirm that the integration of the Tesseract engine with the OpenCV pre-processing pipeline (specifically binarization and deskewing) provides a reliable foundation for data ingestion, significantly outperforming standard benchmarks which typically range between 71% to 98% [33], [34]. While the overall performance is strong, the slight variance in accuracy across formats suggests that user verification remains a recommended step for critical data extraction.

Table 1. OCR module performance on various document types

No.	File Name	Document Type	Total Character	Character Error (Levenshtein)	Calculated Accuracy (%)
1	SoalGA	DOCX	206	2	99.03%
2	Soal Olimpiade MTK	PDF	828	2	99.76%
3	Tulis Tangan 2	PNG	1876	12	99.63%
4	Daftar Nama	JPG	291	8	97.26%
5	DE1	PNG	2088	26	98.76%
6	SoalGA2	PDF	475	10	97.9%
7	File ScreenShot	PNG	330	5	98.49%

3.2. Performance evaluation of the NLP module

The performance of BAPOLAIC's dual-strategy NLP module was evaluated on two fronts: the quantitative accuracy of its retrieval-based path and the qualitative efficacy of its generative path. The quantitative accuracy of the retrieval-based path, which is responsible for handling frequently asked questions (FAQs), was benchmarked against a curated set of 10 common user queries, with the results detailed in Table 2. Accuracy for this test was calculated using the standard formula.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Queries} \times 100\% \quad (4)$$

As shown in Table 2, the retrieval-based path correctly responded to 9 out of the 10 test cases. Based on (4), the accuracy for the retrieval path was therefore calculated to be 90%. Qualitatively, the generative path proved highly effective, successfully providing coherent and accurate explanations for complex, open-ended topics (e.g., particle swarm optimization) that were not present in the predefined knowledge base. A 90% accuracy for the retrieval path indicates a high degree of reliability for handling the most common user inquiries, ensuring users receive fast and consistent responses. The 10% error rate was attributed to a single failure case (Query 8), where the system correctly identified the intent ("Inquire about tuition fees") but failed to retrieve the specific information, as it was not yet present in the system's knowledge base. This highlights a key distinction between successful intent matching and final answer provision. This dual-strategy architecture provides a significant advantage over single-strategy chatbots by balancing computational efficiency with semantic flexibility. This hybrid model is increasingly recognized as a best practice for developing scalable and responsive conversational agents [35]. The 90% accuracy of BAPOLAIC's retrieval path is a strong result for its specific task of intent matching and is comparable to the high-performance benchmarks seen in specialized AI-based models. Consequently, users of BAPOLAIC are provided with an optimized interaction: near-instantaneous answers for common questions and in-depth, intelligent responses for novel topics. This confirms the robustness and efficiency of the NLP module as the core of the user interaction.

Table 2. NLP retrieval-path performance on a curated test dataset

No.	User Query (Input)	Expected Intent	Predicted Intent (BAPOLAIC)	Result (Correct/Incorrect)
1	"Who is your developer?"	Ask about creator identity	Ask about creator identity	Correct
2	"Who is Director Polibatam"	Ask about Director identity	Ask about Director identity	Correct
3	"Tell me about BAPOLAIC"	Ask about BAPOLAIC's function	Ask about BAPOLAIC's function	Correct
4	"What are the features available?"	Ask about system features	Ask about system features	Correct
5	"What programs are available?"	Ask about available study programs	Ask about available study programs	Correct
6	"Explain about the PSO"	Request explanation of PSO	Request explanation of PSO	Correct
7	"Tell me the location of Polibatam campus"	Ask for campus location	Ask for campus location	Correct
8	"How much is the tuition fee?"	Inquire about tuition fees	Inquire about tuition fees	Incorrect
9	"What is the function of OCR in BAPOLAIC? "	Ask about OCR function	Ask about OCR function	Correct
10	"Tell Me about the Polibatam"	Ask about Polibatam	Ask about Polibatam	Correct

3.3. Performance evaluation of Gemini API integration

The Gemini API, serving as BAPOLAIC's core reasoning engine, was benchmarked throughout January 2025 to evaluate speed, reliability, and scalability. As visualized in Figure 4, the horizontal X-axis represents time, while the vertical Y-axis measures specific metrics including requests/second, latency in milliseconds, and error percentages. The evaluation revealed a distinct performance trade-off: *cloudaicompanion.googleapis.com* endpoint achieved a superior median latency of only 0.06 ms but encountered a notable 13.33% error rate during traffic spikes. In contrast, *generativelanguage.googleapis.com* endpoint maintained perfect reliability (0% error rate) with a significantly higher median latency of 0.92 ms.

Traffic analysis during this period indicated that peak daily usage reached only 6.07% of the free tier limit, confirming the system's substantial scalability for academic environments. To prioritize a responsive user experience, the *cloudaicompanion* endpoint was selected as the primary interface, supplemented by a backend retry mechanism to mitigate intermittent failures and achieve a practical balance between speed and operational stability. These results, combined with the system's ability to handle up to 10 consecutive documents, validate BAPOLAIC as a robust and specialized tool compared to general-purpose AI benchmarks.

3.4. Overall discussion, limitations, and future work

The empirical results collectively validate BAPOLAIC as a high-performing multimodal AI assistant tailored for academic environments. The quantitative findings—including a 98.69% average OCR accuracy, 90% NLP retrieval accuracy, and stable low-latency API integration—demonstrate the successful implementation of the proposed architecture. This work effectively addresses the research gap by providing a holistic tool capable of multi-format document processing and seamless multimodal interaction. The primary contribution lies in the successful integration of disparate technologies into a single, cohesive, and user-centric platform. Despite these outcomes, several limitations remain. First, evaluation was conducted on a controlled dataset; thus, the OCR module requires further stress-testing against a broader corpus of complex handwriting styles. Second, a formal large-scale user acceptance study using frameworks such as the

technology acceptance model (TAM) is yet to be conducted to gauge broader end-user perceptions. Finally, the current prototype lacks persistent conversation memory to maintain context across different user sessions. Future research will focus on addressing these constraints. The immediate roadmap involves expanding test datasets for handwriting recognition and implementing a conversation memory module to facilitate more contextual multi-turn dialogues. Subsequently, formal usability studies will be conducted with students and faculty at Politeknik Negeri Batam to validate practical utility and gather qualitative feedback. These steps are essential for transitioning BAPOLAIC from a functional prototype into a production-ready tool for enhancing academic productivity.

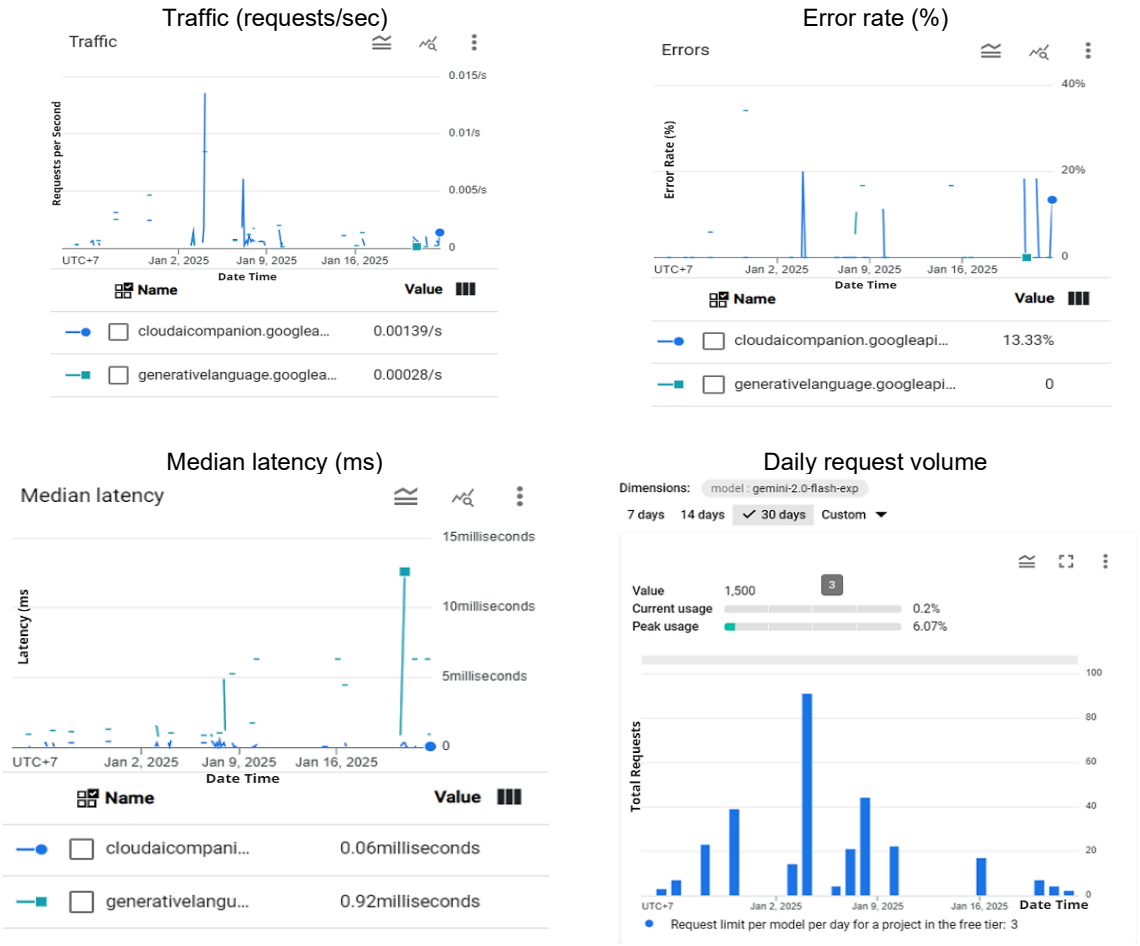

Figure 4. Gemini API performance metrics over a 30-day test period

4. CONCLUSION

This research successfully designed, implemented, and evaluated BAPOLAIC, a multimodal AI assistant engineered to bridge functional gaps in conventional academic chatbots. The empirical evaluation validated the system's efficacy, highlighted by a 98.69% average OCR accuracy and a 90% success rate for the retrieval-based NLP module. Furthermore, the integration of the Gemini API provided a highly responsive reasoning engine, while comparative analysis confirmed BAPOLAIC's specialized advantage in multi-document management for educational workflows. These findings demonstrate that the system successfully integrates its core technologies into a cohesive, high-performance platform that meets the initial research objectives. The primary contribution of this work is a validated framework for developing specialized, multi-functional AI assistants in academic environments. While the current prototype serves as a robust proof-of-concept, limitations such as the absence of persistent conversation memory were identified. Consequently, future research will focus on implementing memory features to enable contextual multi-turn dialogues and conducting large-scale user acceptance studies using the Technology Acceptance Model (TAM). Such developments will be instrumental in transitioning BAPOLAIC from a functional prototype into a production-ready tool with significant potential to enhance academic productivity.

FUNDING INFORMATION

This research was supported by Politeknik Negeri Batam through internal research funding under the 2025 Research Grant Scheme.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Ryan Satria Wijaya	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
Rival Fahreji		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that there is no conflict of interest regarding the publication of this paper.

DATA AVAILABILITY

Data availability is not applicable to this paper, as no new data were created or analyzed in this study.

REFERENCES

[1] M. Steidl, V. Golendukhina, M. Felderer, and R. Ramler, "Automation and development effort in continuous AI development: a practitioners' survey," in *2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, Sep. 2023, pp. 120–127, doi: 10.1109/SEAA60479.2023.00027.

[2] G. Santoso, J. Setiawan, and A. Sulaiman, "Development of OpenAI API based chatbot to improve user interaction on the JBMS website," *G-Tech: Jurnal Teknologi Terapan*, vol. 7, no. 4, pp. 1606–1615, Oct. 2023, doi: 10.33379/gtech.v7i4.3301.

[3] R. Kulkarni and D. R. C. Jaiswal, "A survey on AI Chatbots," *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 9, pp. 1738–1744, Sep. 2023, doi: 10.22214/ijraset.2023.55911.

[4] S. U. Nze, "AI-powered chatbots," *Global Journal of Human Resource Management*, vol. 12, no. 6, pp. 34–45, Jun. 2024, doi: 10.37745/gjhrm.2013/vol12n63445.

[5] D. P. Malik, "Enhancing human-robot interaction through advanced natural language processing techniques," *International Journal of Scientific Research in Engineering and Management*, vol. 08, no. 06, pp. 1–5, Jun. 2024, doi: 10.55041/IJSREM35803.

[6] R. J. Ong, R. A. A. Raof, S. Sudin, and K. Y. Choong, "A review of chatbot development for dynamic web-based knowledge management system (KMS) in small scale agriculture," *Journal of Physics: Conference Series*, vol. 1755, no. 1, p. 012051, Feb. 2021, doi: 10.1088/1742-6596/1755/1/012051.

[7] M. Fahad, A. Akbar, S. Fathima, D. Mohammed, and A. Bari, "Windows based ai-voice assistant system using GTTS," *Mathematical Statistician and Engineering Applications*, vol. 72, no. 1, pp. 1572–1580, 2023.

[8] F. B. Safir, A. Q. Ohi, M. F. Mridha, M. M. Monowar, and M. A. Hamid, "End-to-end optical character recognition for Bengali handwritten words," in *2021 National Computing Colleges Conference (NCCC)*, Mar. 2021, pp. 1–7, doi: 10.1109/NCCC49330.2021.9428809.

[9] D. H. Chang, M. P.-C. Lin, S. Hajian, and Q. Q. Wang, "Educational design principles of using AI chatbot that supports self-regulated learning in education: goal setting, feedback, and personalization," *Sustainability*, vol. 15, no. 17, p. 12921, Aug. 2023, doi: 10.3390/su151712921.

[10] C. M. Kumar, M. Fardeen, P. Rohit, and M. Rakesh, "AI based chatbot to answer FAQs," *Interantional Journal of Scientific Research in Engineering and Management*, vol. 07, no. 03, Mar. 2023, doi: 10.55041/IJSREM18072.

[11] P. K. H M, "AI based multilingual college enquiry voice bot using Python," *International Journal for Research in Applied Science and Engineering Technology*, vol. 12, no. 4, pp. 4300–4305, Apr. 2024, doi: 10.22214/ijraset.2024.60961.

[12] T. Khete and A. Bakshi, "Autonomous assistance system for visually impaired using tesseract OCR & gTTS," *Journal of Physics: Conference Series*, vol. 2327, no. 1, p. 012065, Aug. 2022, doi: 10.1088/1742-6596/2327/1/012065.

[13] M. ter Hoeve, R. Sim, E. Nouri, A. Fourney, M. de Rijke, and R. W. White, "Conversations with documents: an exploration of document-centered assistance," in *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval*, Mar. 2020, pp. 43–52, doi: 10.1145/3343413.3377971.

[14] V. Khatri, S. Tiwari, P. Ninama, and V. Pandet, "Developing a resume app using Google Gemini Pro," *International Journal of Worldwide Engineering Research*, vol. 2, no. 4, pp. 4–6, 2024.

[15] K. Joshi, "Study of Tesseract OCR," *GLS KALP: Journal of Multidisciplinary Studies*, vol. 1, no. 2, pp. 41–50, Mar. 2024, doi: 10.69974/glskalp.01.02.54.

[16] S. Jatmika, "Cognition-based document matching within the chatbot modeling framework," *Journal of Applied Data Sciences*, vol. 5, no. 2, pp. 613–627, May 2024, doi: 10.47738/jads.v5i2.209.

[17] R. T. Alegado, R. G. Luciano, G. M. Alcantara, A. L. Laureta, B. C. Y. Abando, and J. P. S. Makidang, "Design and development of chatbot request system," *International Journal of Science and Applied Information Technology*, vol. 11, no. 3, pp. 1–5, Jun. 2022, doi: 10.30534/ijsait/2022/011132022.
[18] B. Ayan Ahamad, S. Vikas S., and G. Madhusudan Ramdas, "AI based smart robot (chatbot) using Python," *International Journal of Innovative Science and Research Technology (IJISRT)*, pp. 569–572, Mar. 2024, doi: 10.38124/ijisrt/IJISRT24MAR521.
[19] V. Jonatan and A.-A. Igor, "Creation of a chatbot based on natural language processing for WhatsApp," *Electrical and Electronics Engineering: An International Journal (ELELIJ)*, vol. 12, no. 2/3/4, pp. 11–24, 2023, doi: 10.14810/elelij.2023.12402.
[20] P. Bansal, "Prompt engineering importance and applicability with generative AI," *Journal of Computer and Communications*, vol. 12, no. 10, pp. 14–23, 2024, doi: 10.4236/jcc.2024.1210002.
[21] H. Madhavi, J. Cherian, Y. Khamkar, and D. Bhagat, "Low-resource language processing: an OCR-driven summarization and translation pipeline," *arXiv:2505.11177,* May 2025,
[22] P. Sisodia, "Optical character recognition development using Python," *Journal of Informatics Electrical and Electronics Engineering (JIEEE)*, vol. 4, no. 3, pp. 1–13, 2023, doi: 10.54060/jieee.2023.75.
[23] Z. B. Reza *et al.*, "RAG for effective supply chain security questionnaire automation," Dec. 2024, *arXiv:2412.13988*.
[24] E. Yang, J. Amar, J. H. Lee, B. Kumar, and Y. Jia, "The geometry of queries: query-based innovations in retrieval-augmented generation for healthcare QA," in *Proceedings of Machine Learning Research*, Jul. 2025, pp. 1–27.
[25] K. Juvekar and A. Purwar, "COS-Mix: cosine similarity and distance fusion for improved information retrieval," *arXiv: 2406.00638*, Jun. 2024.
[26] Zaid Mundher, Wissam Khalf Khater, and Laith Mohammed Ganeem, "Adopting text similarity methods and cloud computing to build a college chatbot model," *Journal of Education and Science*, vol. 30, no. 1, pp. 117–125, Mar. 2021, doi: 10.33899/edusj.2020.127244.1079.
[27] R. Islam and I. Ahmed, "Gemini-the most powerful LLM: myth or truth," in *2024 5th Information Communication Technologies Conference (ICTC)*, May 2024, pp. 303–308, doi: 10.1109/ICTC61510.2024.10602253.
[28] M. S. Ravindra and K. T. K. Kumar, "JavaScript powered web based speech to text converter," *International Journal for Multidisciplinary Research (IJFMR)*, vol. 5, no. 2, pp. 1–14, 2023.
[29] N. Rachmat and D. P. Kesuma, "Implementation of Gemini LLM in developing Android-based chatbot applications," (in Indonesian: Implementasi LLM Gemini pada pengembangan aplikasi chatbot berbasis Android) *Jurnal Ilmu Komputer (JUIK)*, vol. 4, no. 1, p. 40, Feb. 2024, doi: 10.31314/juik.v4i1.2831.
[30] P. Leutz-Schmidt *et al.*, "Performance of large language models ChatGPT and Gemini on workplace management questions in radiology," *Diagnostics*, vol. 15, no. 4, p. 497, Feb. 2025, doi: 10.3390/diagnostics15040497.
[31] N. Bumann, "Automated chatbot using speech-to-text and text-to-speech with mobile app integration," 2023.
[32] L. Padilha, A. C. C. Souza, and F. C. M. Souza, "A chatbot system to support visually impaired students: a pilot experiment," in *Anais da VI Escola Regional de Engenharia de Software (ERES 2022)*, Oct. 2022, pp. 151–160, doi: 10.5753/eres.2022.227985.
[33] M. Muthusundari, A. Velpoorani, S. Venkata Kusuma, T. L, and O. k. Rohini, "Optical character recognition system using artificial intelligence," *LatIA*, vol. 2, p. 98, Aug. 2024, doi: 10.62486/latia202498.
[34] N. Kotwal, A. Sheth, G. Unnithan, and N. Kadaganchi, "Optical character recognition using tesseract engine," *International Journal of Engineering Research & Technology (IJERT)*, vol. 10, no. 9, pp. 381–385, 2021.
[35] L. Benaddi *et al.*, "Seq2Seq model-based chatbot with LSTM and attention mechanism for enhanced user interaction," *arXiv: 2501.00049*, Dec. 2024,.

BIOGRAPHIES OF AUTHORS

Ryan Satria Wijaya is an assistant professor and a lecturer at Politeknik Negeri Batam specializing in robotics, motion control, computer vision, and artificial intelligence, has contributed to various research projects and publications in these fields. His work includes studies on deep learning algorithms for object detection, path planning in service robots, and the development of interactive learning media for vocational education. His research aims to advance technological applications in robotics and AI, enhancing both academic understanding and practical implementations. He can be contacted at ryan@polibatam.ac.id.

Rival Fahreji is an undergraduate student in robotics engineering at Politeknik Negeri Batam, specializing in AI, machine learning, LLMs, embedded systems, and web application development. In this study, he contributed to methodology, data analysis, and system implementation. His research experience includes OCR-based document processing for academic services, human-machine interfaces for AGVs, participation in the Indonesian Robotics Contest, and hockey-playing robot design. He can be contacted at rivalfahreji91@gmail.com.