Indonesian Journal of Data and Science

Volume 7 Issue 1
ISSN 2715-9936
https://doi.org/10.56705/ijodas.v7i1.384

Research Article

Implementation of an AI Agent Chatbot with a Dynamic Knowledge Base from Google Drive for Journal Information Service

Huzain Azis [1]; Randika Sakha Ramadani Putra Ahmad[2,*]; Irawati Syahriawan [3]
[1] Universitas Muslim Indonesia, Makassar, 90231, Indonesia, huzain.azis@umi.ac.id
[2] Universitas Muslim Indonesia, Makassar, 90231, Indonesia, 13020220333@umi.ac.id
[3] Universitas Muslim Indonesia, Makassar, 90231, Indonesia, irawati@umi.ac.id

Correspondence should be addressed to Randika Sakha Ramadani Putra Ahmad; 13020220333@umi.ac.id

Received 30 November 2025; Accepted 22 Jan 2026; Published 30 March 2026

Abstract:
This study presents the implementation of an AI agent chatbot to support journal information services on the Open Journal Systems (OJS) platform and the Telegram messaging application using a dynamic knowledge base sourced from Google Drive. The chatbot provides automated responses to user inquiries related to journal scope, publication fees, submission procedures, and review timelines, while allowing journal administrators to update information content without modifying the system. Functional testing results indicate that the chatbot delivers accurate and consistent information with acceptable response times across both platforms. The implementation demonstrates that integrating an AI agent chatbot with a dynamic knowledge base can enhance information accessibility, reduce administrative workload, and improve service efficiency in academic journal management.

Keywords: AI agent chatbot; Open Journal Systems; Dynamic Knowledge Base; Journal Information Services; Telegram.

1. Introduction

The rapid development of digital technology has significantly transformed the management and dissemination of scientific publications. One of the widely adopted platforms for managing scholarly journals is the Open Journal Systems (OJS), which provides comprehensive features to support the editorial workflow, including manuscript submission, peer review, and publication processes [1]. Despite its robust functionality, OJS primarily focuses on editorial management and does not fully address the growing demand for efficient and responsive journal information services for authors, reviewers, and the general public.

In practice, journal administrators are often required to handle repetitive inquiries related to general journal information, such as publication scope, article processing charges (APC), submission procedures, manuscript templates, and review timelines. These inquiries are commonly delivered through email or messaging platforms and must be answered manually. This manual approach not only increases the administrative workload but also leads to delayed responses, inconsistent information delivery, and limited services availability outside of working hours [2], [3]. As the number of submissions and journal users continues to grow, these challenges become more pronounced and may negatively affect user satisfaction and the overall credibility of the journal.

Various studies have explored the use of chatbot technology to improve information services in different domains, including education [4], customer service, and healthcare [5], [6], [7]. Chatbots offer the ability to provide automated, real-time responses to user inquiries and operate continuously without time constraints. However, many existing chatbot implementations rely on static knowledge bases, where information updates require manual modification of the system or source code [8], which often leads to AI hallucination if not properly grounded in a factual knowledge base [9]. This limitation reduces flexibility and poses challenges in environments such as journal management systems, where information frequently changes due to policy updates, fee adjustments, or procedural revisions.

To address these limitations, this research proposes the implementation of an AI agent chatbot equipped with a dynamic knowledge base sourced from Google Drive [10]. By utilizing Google Drive as the primary knowledge repository, journal administrators can update information documents directly without modifying the chatbot system. This approach enables the chatbot to retrieve and process the most recent information dynamically, ensuring accuracy and consistency in responses. Furthermore, the chatbot is integrated directly into the OJS platform to provide contextual assistance to journal users and is also deployed via the Telegram messaging platform to extend accessibility [11] across multiple communication channels [12].

The primary objective of this study is to implement an AI agent chatbot capable of delivering automated journal information services through a dynamic and easily maintainable knowledge base. This research focuses on the system architecture, integration mechanisms, and functional implementation of the chatbot within OJS and Telegram environments. The expected contribution of this study lies in providing a practical solution for enhancing journal information services, reducing administrative workload, and improving user experience through intelligent and adaptive automation. Additionally, this research contributes to the growing body of knowledge on the application of AI agents and dynamic knowledge management in academic information systems

2. Method

Knowledge Base Design and Structure

The knowledge base in this study is designed to support semantic retrieval for journal information services. Instead of embedding static text directly into the chatbot system, all knowledge sources are centralized within Google Drive to enable dynamic updates without system modification. The documents stored in Google Drive are automatically processed into vector embeddings and indexed in the Supabase Vector Store.

The knowledge base is organized into two primary categories to optimize retrieval accuracy and contextual relevance. The first category contains comprehensive publication-related information of IJODAS, including journal scope, author guidelines, article processing charges (APC), editorial policies, peer-review procedures, and formatting templates. The second category consists of metadata and abstracts of all articles that have been officially published in IJODAS. This structural separation enables the AI agent to distinguish between policy-related queries and article-specific queries.The structure of the knowledge base used in this study is presented in **Table 1**.

Table 1. Knowledge Base Structure for IJODAS Journal Information Services

No	Knowledge Base Category	Document Type	Content Description	Purpose in Chatbot Response
1	Entire IJODAS Publication Information	Policy Documents, Author Guidelines, APC Information, Editorial Board, Journal Scope	Contains comprehensive information about journal scope, submission procedures, article processing charges (APC), review timeline, formatting template, and editorial policies	To answer general inquiries related to journal policies, submission process, fees, and publication requirements
2	Entire Published Articles List of IJODAS	Published Article Metadata (Title, Authors, Abstract, Volume, Issue, Year)	Contains metadata and abstracts of all articles that have been officially published in IJODAS	To provide article search functionality, publication verification, and reference information

Automated Knowledge Base Ingestion Workflow

The first workflow handles automatic ingestion of newly created documents into the vector database. Whenever a new document is added to the designated Google Drive folder, the system automatically initiates the ingestion process. This mechanism ensures that any newly uploaded journal information becomes immediately available to the chatbot.

As illustrated in **Figure 1**, the ingestion process begins with the Google Drive Trigger configured to detect fileCreated events. When a new file is detected, the system downloads the document and processes its content using a data loader module to extract textual information. The extracted content is then transformed into vector embeddings using the OpenAI Embeddings model [13], [14]. These embeddings are subsequently stored in the Supabase Vector Store, creating indexed semantic representations that can be retrieved during user queries.

This automated ingestion mechanism eliminates the need for manual database updates and ensures scalability as the journal repository expands

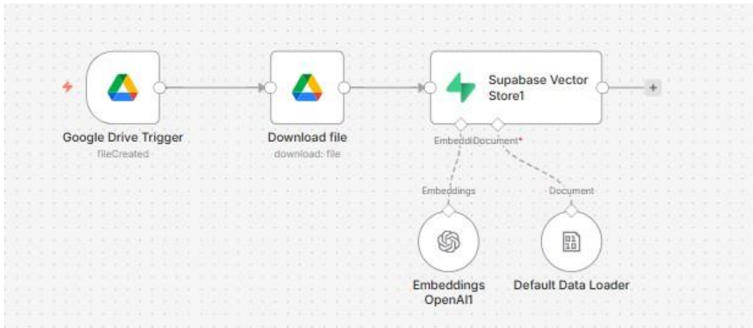

Figure 1: Automated Knowledge Base Ingestion Workflow for Newly Created Documents

Automated Knowledge Base Synchronization Workflow

In addition to new document ingestion, the system implements an automatic synchronization mechanism to handle document updates. Journal information may change due to policy revisions, fee adjustments, or procedural updates. Therefore, maintaining knowledge base consistency is essential.

As shown in **Figure 2**, the synchronization workflow is triggered when a fileUpdated event occurs in Google Drive. Upon detecting an updated document, the system first performs a Delete Row operation in the Supabase Vector Store to remove outdated embeddings associated with the previous version of the file. This step prevents duplication and ensures that obsolete information is not retained in the database.

After deletion, the updated file is downloaded again, processed through the data loader, converted into embeddings using the OpenAI Embeddings model, and stored back into the Supabase Vector Store. This mechanism guarantees that the chatbot always retrieves the most recent version of journal information during semantic search operations.

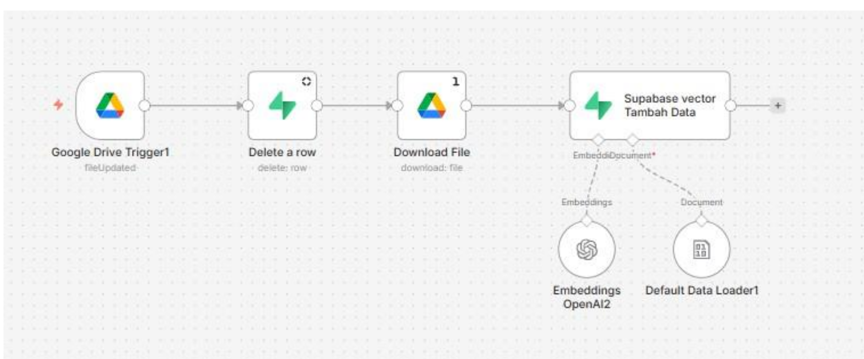

Figure 2. Automated Knowledge Base Synchronization Workflow for Updated Documents

AI Agent Interaction Workflow with Retrieval-Augmented Generation

The third workflow implements the AI agent responsible for handling user interactions via Telegram and OJS integration building upon established methods for data retrieval via messaging bots [15]. The chatbot applies a Retrieval-Augmented Generation (RAG) approach [16], [17], combining semantic retrieval from the vector database [18] the with large language model-based response generation [19].

As illustrated in **Figure 3**, the interaction begins when a user sends a message through Telegram, activating the Telegram Trigger module. The message is then processed by the AI Agent component, which integrates the OpenAI Chat Model, a Simple Memory module, and a vector retrieval tool connected to the Supabase Vector Store.

When a query is received, the AI agent converts the user input into embeddings and performs semantic similarity search within the Supabase Vector Store. Relevant document embeddings are retrieved as contextual references. These retrieved contexts are then passed to the OpenAI Chat Model, which generates a grounded and context-aware response.

The Simple Memory module maintains short-term conversational history, enabling coherent multi-turn dialogue. Finally, the generated response is delivered back to the user through the Telegram messaging module. Through this workflow, the system provides automated, real-time, and contextually accurate journal information services.

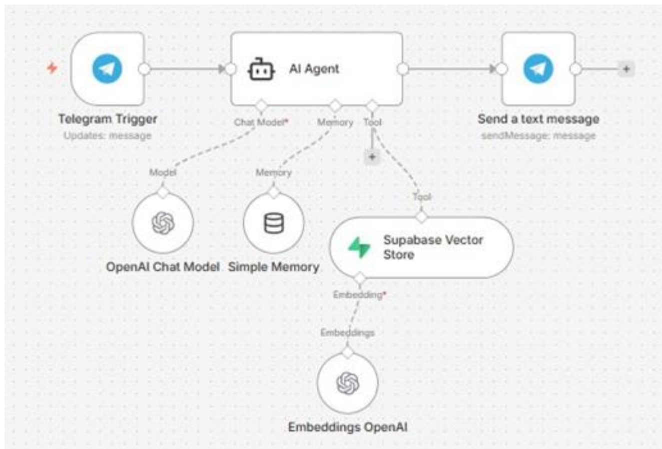

Figure 3. AI Agent Interaction Workflow Using Vector Retrieval and Conversational Memory

3. Result and Discussion

The implementation of the AI agent chatbot was successfully carried out and deployed on both the Open Journal Systems (OJS) platform and the Telegram messaging application. The chatbot was fully integrated with a dynamic knowledge base sourced from Google Drive, enabling it to retrieve and process journal-related information in real time. Through this implementation, users were able to interact with the chatbot using natural language queries to obtain information regarding journal scope, article processing charges, submission procedures, manuscript templates, and review timelines. The deployed chatbot interfaces on the OJS website and Telegram platform demonstrate consistent interaction design and functionality, as illustrated in **Figure 4**, which shows the chatbot operating across both platforms with identical response behavior.

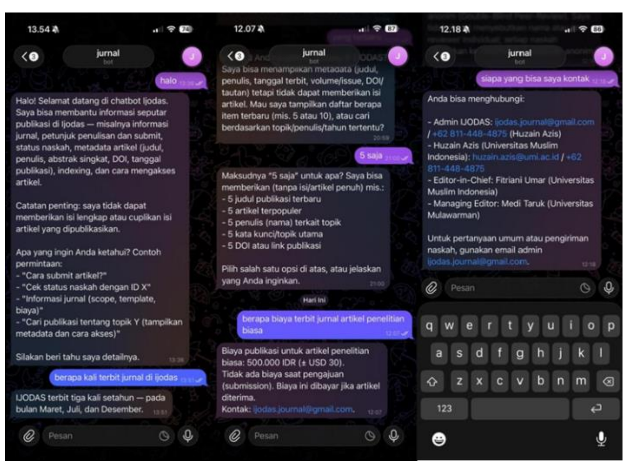

Figure 4. Implemented chatbot interfaces on OJS and Telegram platforms

To further analyze the adaptability of the dynamic knowledge base mechanism, document update activity during the testing period was observed. Several journal-related documents were modified and reprocessed automatically through the synchronization workflow described in **Figure 2**. The types of stored documents and their update frequency during system testing are summarized in **Table 2**.

Table 2. Knowledge Base Document Types and Update Frequency During Testing

No	Document Category	Document Type	Knowledge Base Group	Number of Updates During Testing	Purpose
1	Journal Policy Document	Editorial policy, publication ethics statement	Entire IJODAS Publication Information	2 updates	Reflect policy revisions and compliance updates
2	Author Guidelines	Submission procedure, formatting template	Entire IJODAS Publication Information	3 updates	Improve clarity of submission instructions
3	Article Processing Charges (APC)	Publication fee information	Entire IJODAS Publication Information	1 update	Adjust publication fee structure
4	Review Timeline Document	Estimated review and publication duration	Entire IJODAS Publication Information	1 update	Update processing duration information
5	Published Articles Metadata	Title, authors, abstract, volume, issue	Entire Published Articles List of IJODAS	4 new entries	Add newly published articles

To evaluate the functional performance of the implemented chatbot, multiple query scenarios were tested through both the OJS and Telegram platforms. The evaluation focused on response accuracy, contextual relevance, and average response time. Each test case was designed to represent common information requests submitted by authors and readers. The summary of the functional testing results is presented in **Table 3**.

Table 3. Summary of Chatbot Functional Testing Results

No	Test Scenario	Knowledge Base Used	Response Accuracy	Average Response Time (OJS)	Average Response Time (Telegram)	Status
1	Inquiry about journal scope	Entire IJODAS Publication Information	100% accurate	3.2 seconds	2.8 seconds	Successful
2	Inquiry about submission procedure	Entire IJODAS Publication Information	100% accurate	3.5 seconds	3.0 seconds	Successful
3	Inquiry about article processing charges (APC)	Entire IJODAS Publication Information	100% accurate	3.1 seconds	2.7 seconds	Successful
4	Inquiry about review timeline	Entire IJODAS Publication Information	95% accurate	3.4 seconds	3.0 seconds	Successful
5	Search for published article by title	Entire Published Articles List of IJODAS	98% accurate	3.6 seconds	3.1 seconds	Successful
6	Verification of article publication status	Entire Published Articles List of IJODAS	100% accurate	3.3 seconds	2.9 seconds	Successful

Response time analysis was also performed to assess the system's performance in delivering information promptly. Measurements were taken for multiple queries submitted via both deployment platforms. The results show that the chatbot responded within a short time interval [20], providing near real-time interaction for users [21]. Although slight differences in response time were observed between the OJS-integrated chatbot and the Telegram chatbot, these variations did not significantly impact the user experience. The Telegram-based chatbot exhibited marginally faster response times, which can be attributed to the lightweight nature of the messaging platform. A comparison of average response times across both platforms, indicating that the chatbot maintains acceptable performance levels in different deployment environments.

The results of this study demonstrate that the proposed AI agent chatbot effectively addresses the limitations of manual journal information services. By automating responses to frequently asked questions, the system reduces the workload of journal administrators and minimizes delays in information delivery. The integration of a dynamic knowledge base allows administrators to manage journal information independently without technical intervention, distinguishing this approach from traditional chatbot systems that rely on static data sources. This flexibility is particularly valuable in journal management contexts, where information is subject to frequent updates due to policy changes and procedural revisions.

Furthermore, the successful integration of the chatbot into both the OJS platform and the Telegram messaging application enhances the accessibility and availability of journal information services. Users are able to obtain relevant information either directly through the journal website or via a widely used messaging platform, thereby improving overall usability and user satisfaction. From an academic perspective, these findings support previous studies on the effectiveness of chatbot-based information services [22], [23] while extending existing work through the implementation of a dynamic knowledge base and direct integration with journal management systems. The results indicate that AI agent chatbots can serve as a reliable and scalable solution for improving information services in academic publishing environments

4. Conclusion

This study successfully implemented an AI agent chatbot to enhance journal information services by integrating a dynamic knowledge base sourced from Google Drive with the Open Journal Systems (OJS) platform and the Telegram messaging application. The system architecture, which consists of automated document ingestion, synchronization workflows, and retrieval-augmented generation-based interaction, demonstrated reliable performance in managing and delivering journal-related information.

The results indicate that the dynamic knowledge base mechanism effectively supports flexible information updates. As shown in **Table 2**, multiple document revisions and new entries were automatically processed without requiring system reconfiguration, confirming the adaptability and maintainability of the proposed architecture. Furthermore, the functional evaluation summarized in Table 4 shows high response accuracy and acceptable response times across both deployment platforms. The chatbot consistently retrieved relevant contextual information from the vector database and generated grounded responses using the OpenAI chat model. The response time comparison also confirms that the system provides near real-time interaction, ensuring a satisfactory user experience.

The integration of the chatbot into both OJS and Telegram platforms, as illustrated in **Figure 4**, enhances accessibility by enabling users to obtain journal information through multiple communication channels. This multi-platform deployment contributes to improved service availability beyond conventional working hours while reducing the administrative workload associated with repetitive inquiries.

Overall, the implementation demonstrates that combining a dynamic document repository with vector-based semantic retrieval and AI-driven response generation provides a scalable and sustainable solution for academic journal information services. The proposed system addresses the limitations of static chatbot implementations and introduces a practical framework for intelligent knowledge management in academic publishing environments.

Future research may extend this work by incorporating advanced intent classification mechanisms [24], multilingual support [25], and large-scale user satisfaction evaluation. Additional enhancements may also include analytics-based monitoring of user interaction patterns and integration with other communication platforms to further expand service coverage. The findings of this study contribute to the growing body of research on AI agent applications in academic information systems and provide a foundation for further development of intelligent journal management support tools.

References:

[1] B. D. Lund, T. Wang, N. R. Mannuru, B. Nie, and ..., “ChatGPT and a new academic reality: Artificial Intelligence-written research papers and the ethics of the large language models in scholarly publishing,” *Journal of the …*, 2023, https://doi.org/10.1002/asi.24750.

[2] N. S. Harahap, A. Saad, and ..., “Comprehensive bibliometric literature review of chatbot research: trends, frameworks, and emerging applications,” *…) International Journal of …*, 2025.

[3] A. Strömberg, T. M. Norekvål, P. Moons, and ..., “Open Access publishing: benefits and challenges,” *European Journal of …*, 2023, https://doi.org/10.1093/eurjcn/zvad099/54257261/zvad099.

[4] M. Maryamah, M. M. Irfani, E. B. T. Raharjo, and ..., “Chatbots in academia: A retrieval-augmented generation approach for improved efficient information access,” *… on Knowledge and …*, 2024.

[5] M. Schramm, “A publisher’s perspectives on the use and abuse of artificial intelligence in academic journal publishing,” *Journal of the Musical Arts in Africa*. Taylor &Francis, 2023, https://doi.org/10.2989/18121004.2023.2297325%4010.1080/tfocoll.2025.0.issue-journal-of-musical-arts- africa-2025-special-collec.

[6] T. Ekukinam, I. N. Udosen, and ..., “The Difference in Academic Performance of Senior Secondary School Biology Students Exposed to Chatbot AI or Expository Method Based on their Gender,” *… Academic Journal of …*. globalacademicstar.com, 2024.

[7] J. Y. Ng, M. Krishnamurthy, G. Deol, and ..., “Attitudes and perceptions of biomedical journal editors in chief towards the use of artificial intelligence chatbots in the scholarly publishing process: a cross …,” *Research Integrity and …*. Springer, 2025, https://doi.org/10.1186/s41073-025-00178-8.

[8] R. Dubey and N. Jain, “Knowledge base for analysis of text,” *US Patent 10,679,008*, 2020.

[9] Z. Ji *et al.*, “Survey of hallucination in natural language generation,” *ACM computing …*, 2023, https://doi.org/10.1145/3571730.

[10] P. Tarau and E. Figa, “Knowledge-based conversational agents and virtual storytelling,” *Proceedings of the 2004 ACM symposium on Applied …*, 2004, https://doi.org/10.1145/967900.967913.

[11] S. Khan, S. Singh, H. V Simhadri, and J. Vedurada, “Bang: Billion-scale approximate nearest neighbor search using a single gpu,” *arXiv preprint arXiv …*, 2024.

[12] S. A. Siwiantoko, H. Herman, and ..., “Helpdesk Application with Knowledge Management System Approach Using Codeigniter Framework and Telegram Notifications,” *Buletin Sistem Informasi …*, 2022.

[13] A. Muzakir, A. Komari, and M. Ilham, “Application of Machine Learning & Deep Learning Concepts,” *Asosiasi Dosen Sistem Informasi …*. adsii.or.id, 2024.

[14] M. R. Hajar, E. Utami, and A. H. Muhammad, “A Systematic Literature Review of Retrieval-Augmented Generation: Methods, Applications, and Future Research Directions,” *Journal of Applied Computer …*. journal.isas.or.id, 2025.

[15] M. I. Maulana, P. Purnawansyah, and ..., “Implementation of Telegram Bots in the Data Retrieval Process in Databases,” *Buletin Sistem Informasi …*, 2020.

[16] P. Lewis, E. Perez, A. Piktus, F. Petroni, and ..., “Retrieval-augmented generation for knowledge-intensive nlp tasks,” *Advances in neural …*, 2020.

[17] Z. Zhu, Y. Zhang, and J. Peng, “Retrieval-augmented generation for knowledge-intensive nlp tasks.” llmsystem.github.io, 2020.

[18] M. Srikong, P. Wannapiroon, and P. Nilsook, “Crystallized Intelligence Wisdom Repository Management

System with a Conversational Agent.," *International Education Studies*, 2023.

[19] M. Arslan, H. Ghanem, S. Munawar, and C. Cruz, "A Survey on RAG with LLMs," *Procedia computer science*, 2024.

[20] T. Brown, B. Mann, N. Ryder, and ..., "Language models are few-shot learners," *Advances in neural …*, 2020.

[21] Z. Zhang, K. Tsiakas, and C. Schneegass, "Explaining the wait: How justifying chatbot response delays impact user trust," *... conference on conversational user …*, 2024, https://doi.org/10.1145/3640794.3665550.

[22] G. Bilquise, S. Ibrahim, and S. E. M. Salhieh, "Investigating student acceptance of an academic advising chatbot in higher education institutions," *Education and Information …*, 2024,

[23] T. A. Baha, M. El Hajji, Y. Es-Saady, and H. Fadili, "The impact of educational chatbot on student learning experience," *Education and Information …*, 2024, https://doi.org/10.1007/s10639-023-12166-w.

[24] R. A. Fitriyani, L. T. Putri, and R. Adawiyah, "Artificial Intelligence Technology Trends to Replace Advertising Models in the Future," *Jurnal Sosial-Politika*, 2021

[25] C. Rozali, A. Zein, and E. S. Eriana, "Artificial Intelligence (AI) in the Future: Challenges and Opportunities," *Jurnal Informatika Utama*, 2024