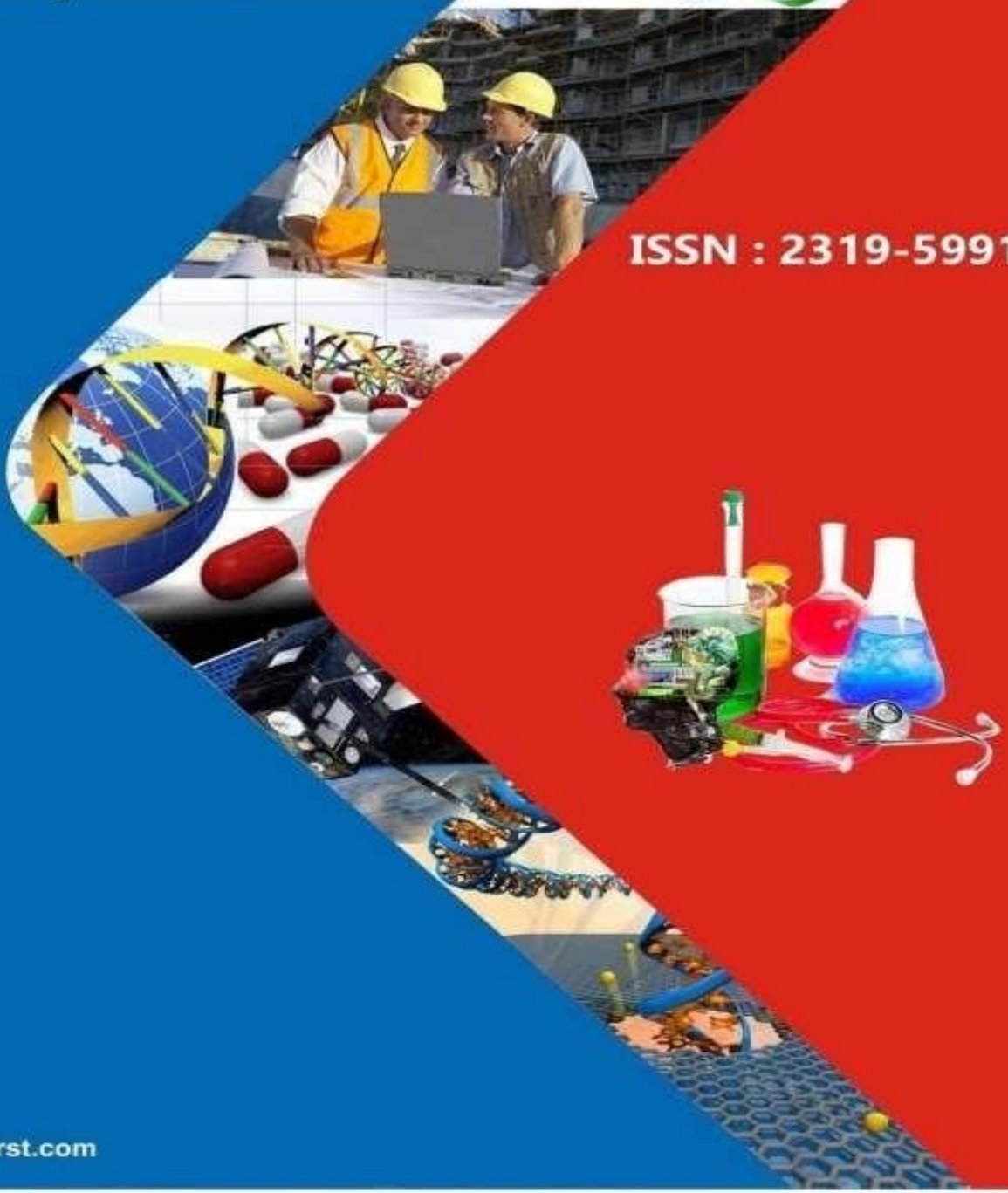
International Journal of Engineering Research and Science & Technology

IJERST

ISSN : 2319-5991



www.ijerst.com

Email: editor@ijerst.com or editor.ijerst@gmail.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 21, Issue 2, 2025

Real-time News Intelligence Integrating Dynamic Content with Adaptive APIs

N.Nagoor Meeravali[1], Gutha Siri vennela[2], Linga.Bhargavi[3], Yadhapalli Raghavendra[4], Padala Dineshkumar[5]

[1]Assistant Professor, Department of Information Technology, Chalapathi Institute of Engineering and Technology, Chalapathi Rd, Nagar, Lam, Guntur, Andhra Pradesh- 522034

[2,3,4,5] Students, Department of Information Technology, Chalapathi Institute of Engineering and Technology, Chalapathi Rd, Nagar, Lam, Guntur, Andhra Pradesh- 522034

Email id: meeravali587@gmail.com[1], sirivennelagutha10@gmail.com[2], lingabhargavi9@gmail.com[3], raghavendrayadlapalli300@gmail.com[4], padaladineshkumar@gmail.com[5]

Abstract:

Real-Time News Intelligence platform is developed using Django to leverage dynamic content and adaptive APIs, delivering timely and actionable insights from a diverse range of news sources. This system efficiently processes, categorizes, and prioritizes news in real time, empowering users with data-driven decision-making capabilities. Built on Django's robust framework, the platform integrates real-time API ingestion, adaptive filtering, and sentiment analysis. It offers an intuitive, user-friendly interface with customizable dashboards and interactive data visualizations. Security remains a top priority, ensuring data integrity, user authentication, and compliance with privacy regulations. Extensive testing across multiple news categories validates the system's reliability and adaptability. By merging Django's scalable architecture with cutting-edge data processing technologies, this platform enhances the way users engage with and interpret real-time news data.

Keywords: Django, News analytics, Real-time processing, API integration, Sentiment analysis, Data visualization, Adaptive filtering, User authentication, Information security

1.Introduction

In the fast-paced digital era, real-time access to relevant and accurate news is critical for businesses, analysts, and consumers. Traditional news aggregation methods often struggle to keep up with the ever-evolving information landscape, leading to delays, content silos, and lack of personalization. Real-Time News Intelligence is a cutting-edge approach that integrates dynamic content with adaptive APIs, enabling seamless access to continuously updated news feeds, breaking developments, and contextual insights. By leveraging AI-powered analytics, machine learning, and Natural Language Processing (NLP), this system filters, categorizes, and prioritizes news based on user preferences, industry trends, and sentiment analysis. Adaptive APIs play a crucial role in this ecosystem, allowing platforms to ingest, analyze, and distribute real-time news from multiple sources, including mainstream media, social platforms, government updates, and niche publications. This ensures businesses, journalists, and individuals receive personalized, verified, and high-impact news tailored to their needs. This paper explores the architecture, functionality, and impact of Real-Time News Intelligence,

demonstrating how it transforms the way information is consumed, analyzed, and acted upon in an increasingly data-driven world. Web search APIs empower developers to seamlessly integrate powerful search capabilities into their applications, providing access to vast troves of internet data with just a few lines of code. These APIs act as gateways to sophisticated search engines, allowing applications to programmatically query the web and retrieve relevant results including webpages, images, news articles, and more. By using web search APIs, developers can enhance their applications with up-to-date information from across the internet, enabling features like content discovery, trend analysis, and intelligent recommendations. With customizable parameters for refining searches and structured response formats for parsing, web search APIs offer a flexible and efficient solution for harnessing the wealth of information available on the web.Amazon Bedrock Agents offers a powerful solution for enhancing chatbot capabilities, and when combined with web search APIs, they address a critical customer pain point. In this post, we demonstrate how to use Amazon Bedrock Agents with a web search API to integrate dynamic web content in your generative AI application.

2.Literature review

Benefits of integrating a web search API with Amazon Bedrock Agents

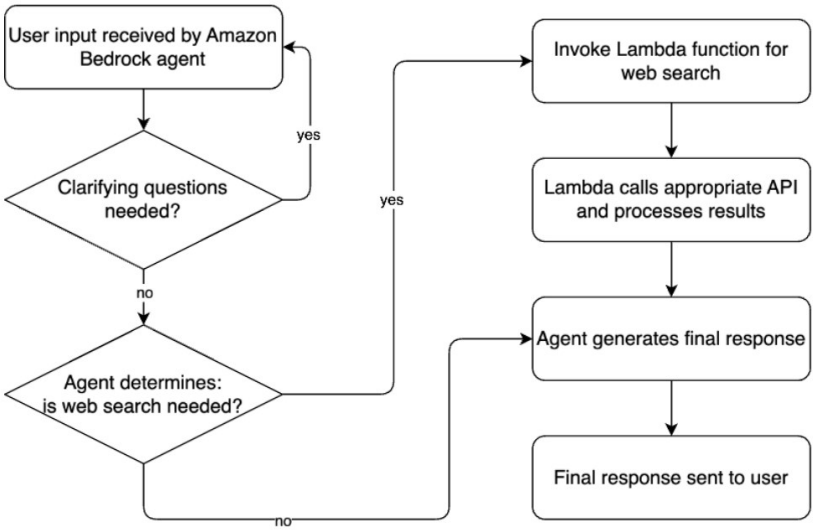
Let's explore how this integration can revolutionize your chatbot experience:

- Seamless in-chat web search – By incorporating web search APIs into your Amazon Bedrock agents, you can empower your chatbot to perform real-time web searches without forcing users to leave the chat interface. This keeps users engaged within your application, improving overall user experience and retention.
- Dynamic information retrieval – Amazon Bedrock agents can use web search APIs to fetch up-to-date information on a wide range of topics. This makes sure that your chatbot provides the most current and relevant responses, enhancing its utility and user trust.
- Contextual responses – Amazon Bedrock agent uses CoT prompting, enabling FMs to plan and run actions dynamically. Through this approach, agents can analyze user queries and determine when a web search is necessary or—if enabled—gather more information from the user to complete the task. This allows your chatbot to blend information from APIs, knowledge bases, and up-to-date web-sourced content, creating a more natural and informative conversation flow. With these capabilities, agents can provide responses that are better tailored to the user's needs and the current context of the interaction.
- Enhanced problem solving – By integrating web search APIs, your Amazon Bedrock agent can tackle a broader range of user inquiries. Whether it's troubleshooting a technical issue or providing industry insights, your chatbot becomes a more versatile and valuable resource for users.

3.Methodology

This solution uses Amazon Bedrock Agents with a web search capability that integrates external search APIs (SerpAPI and Tavily AI) with the agent. The architecture consists of the following key components:

International Journal of Engineering Research and Science & Technology
ISSN 2319-5991 www.ijerst.com
Vol. 21, Issue 2, 2025



The solution flow is as follows:

- User input is received by the Amazon Bedrock agent, powered by Anthropic Claude 3 Sonnet on Amazon Bedrock.
- The agent determines if a web search is necessary, or comes back to the user with clarifying questions.
- If required, the agent invokes one of two Lambda functions to perform a web search: SerpAPI for up-to-date events or Tavily AI for web research-heavy questions.
- The Lambda function retrieves the API secrets securely from Secrets Manager, calls the appropriate search API, and processes the results.
- The agent generates the final response based on the search results.
- The response is returned to the user after final output guardrails are applied.

We securely store the obtained API keys in Secrets Manager. The following examples create secrets for the API keys:

```
aws secretsmanager create-secret \
--name SERPER_API_KEY \
--description "The API secret key for Serper." \
--secret-string "$SERPER_API_KEY"

aws secretsmanager create-secret \
--name TAVILY_API_KEY \
--description "The API secret key for Tavily AI." \
--secret-string "$TAVILY_API_KEY"
```

4. Implementation

The implementation of the AI-driven news analysis chatbot, as visualized in fig involved a strategic orchestration of four key technology components. The architecture diagram illustrates the integrated system of components enabling the chatbot to perform complex news analysis tasks by leveraging user queries, processing information, and retrieving relevant data. For a

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 21, Issue 2, 2025

Robotic Process Automation (RPA) practitioner, this diagram provides a clear blueprint for automating the flow of information and tasks between different technologies, such as Microsoft Copilot Studio, Microsoft Power Automate (MPA), Microsoft Dataverse (MD), and Google Gemini API (GGAPI). By understanding this architecture, RPA practitioners can effectively implement and optimize the chatbot for dynamic news analysis, ensuring seamless communication, efficient data processing, and accurate information retrieval. Microsoft Copilot Studio served as the front end for user interaction, enabling seamless communication and query processing. It captures user intents, whether related to obtaining news summaries, calculating correlations, or posing generic queries. This information is then passed to MPA, which acts as the central orchestrator, facilitating automated interactions and data flow between the different components.

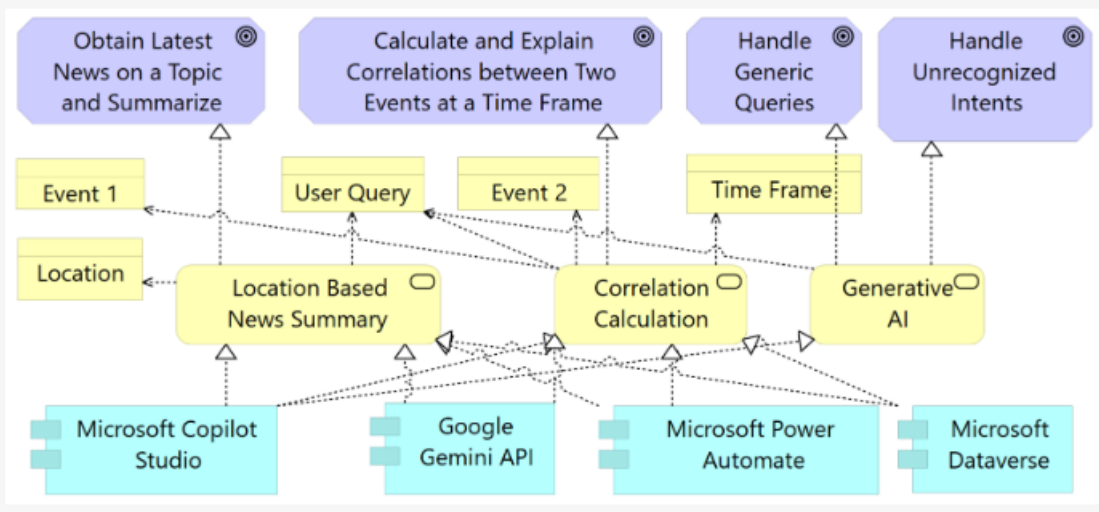


Figure: Architectural diagram for implementing the chatbot agent-based news analytics system

MPA routes requests to appropriate modules based on the user's intent. For instance, requests for news summaries are directed to both GGAPI and MD. The GGAPI leverages its advanced functionalities to accurately match news categories to user queries and generate Fetch XML for efficient retrieval of relevant news reports from the Dataverse repository, which houses a live news database comprising 991,325 news reports categorized into 202 distinct news categories. Additionally, Google Gemini's summarization capabilities facilitate the concise and informative presentation of news data to the user. Similarly, requests for correlation calculations are processed by a dedicated module, utilizing both the GGAPI and MD to fetch and analyze relevant news data. Generic queries are handled by leveraging the generative AI capabilities of the system.

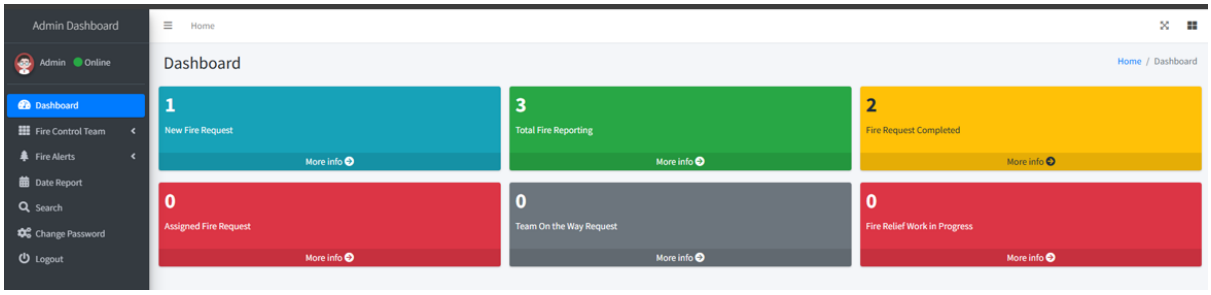
Table: Technology components used for implementing the automated news analytics solution.

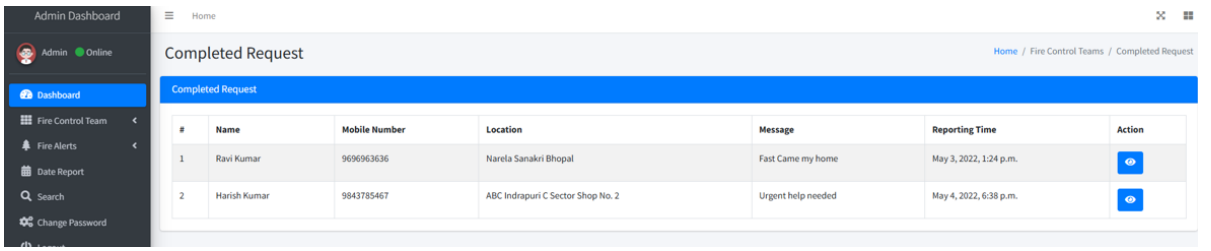
International Journal of Engineering Research and Science & Technology

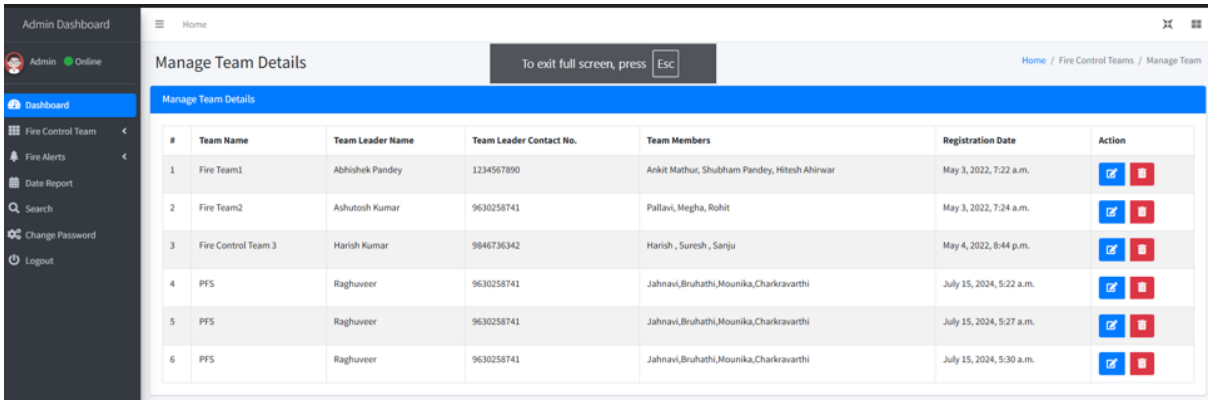
ISSN 2319-5991 www.ijerst.com

Vol. 21, Issue 2, 2025

Requirements for the Proposed System	Microsoft Copilot Studio	Microsoft Power Automate	Microsoft Dataverse	Google Gemini
ObtainLatestLocationNewsSummary()	✓	✓	✓	✓
- Obtain Event and Time Frame	✓			
- Match relevant News Categories		✓		✓
- Generate Fetch XML				✓
- Obtain all Relevant News			✓	
- Summarization	✓	✓		✓
CalculateCorrelation()	✓	✓	✓	✓
- Obtain Event 1, Event 2, Time Frame	✓			
- Match relevant News Categories		✓		✓
- Generate Fetch XML				✓
- Obtain all Relevant News			✓	
- Calculate Daily Frequencies	✓			✓
- Calculate correlations with explanation	✓			✓
HandleGenericQueries()	✓			



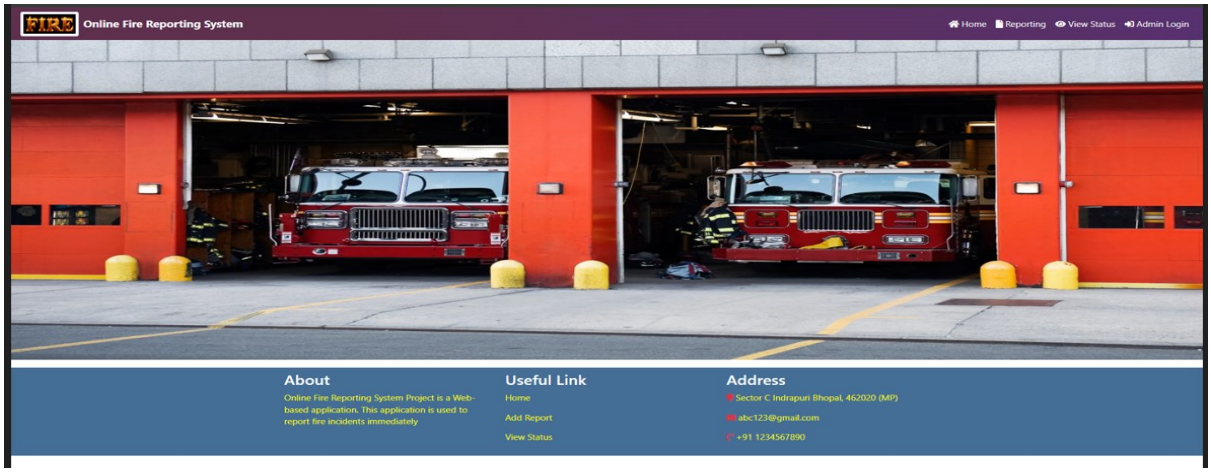


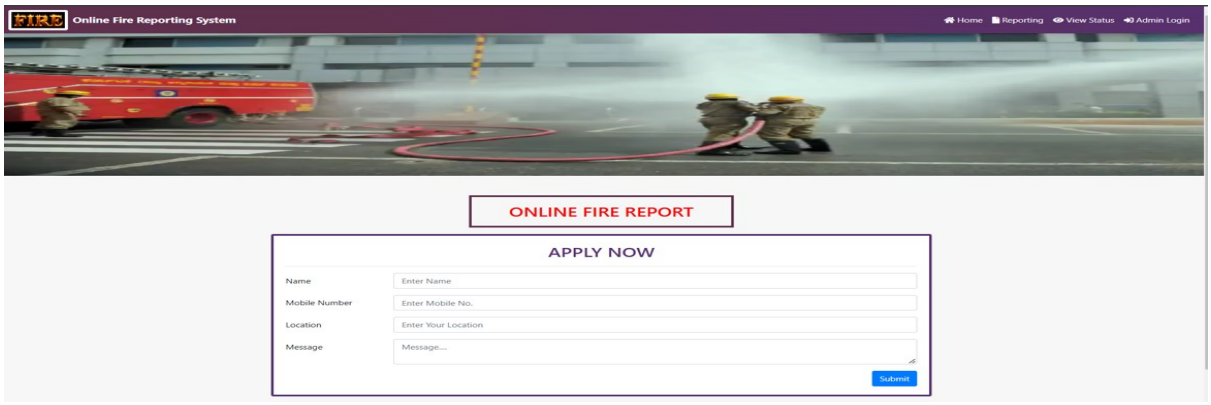


International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 21, Issue 2, 2025





the manuscript meticulously outlines the architectural configuration of the AI-driven chatbot, employing ArchiMate as the notation system. This diagram is central to illustrating the structural and functional relationships within the system, prominently featuring 'access' (i.e., dashed arrow with regular arrowhead) and 'realization' (i.e., dashed arrow with white triangle at the arrowhead) relations, which are pivotal in ArchiMate for depicting interaction dynamics and implementation details, respectively. It is paramount to note that the diagram employs ArchiMate, a standardized notational language specifically designed for architectural representation within complex information systems ArchiMate is endorsed by The Open Group and adheres to rigorous international standards, which dictate the use of specific symbols and color schemes to represent different architectural domains and relationships.

Conclusions

This study has introduced an AI-driven chatbot for real-time news automation, demonstrating its effectiveness in processing large-scale global news data through event summarization and correlation analysis. By leveraging advanced AI methodologies, including Generative Pre-trained Transformers, knowledge graphs, and real-time retrieval systems, the chatbot successfully categorizes and structures information to provide users with precise and context-aware insights. The experimental evaluation validated its efficacy, showing high accuracy in news classification, summarization, and event correlation detection. These capabilities position the chatbot as a valuable tool for applications in critical domains such as disaster response, policy analysis, cybersecurity, and media intelligence, where timely and accurate information is essential for decision-making.

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 21, Issue 2, 2025

Despite its promising contributions, this research has certain limitations. The chatbot's performance is dependent on the quality and diversity of available news sources, making it vulnerable to biases and misinformation inherent in the dataset. Furthermore, the computational overhead required for real-time processing, particularly in correlation analysis, poses challenges in high-frequency data environments. Another limitation is the chatbot's reliance on predefined taxonomies for event categorization, which may not fully capture emerging or evolving news narratives. The system also faces challenges in interpreting ambiguous queries, necessitating further advancements in natural language understanding to improve user interactions.

Future research directions aim to enhance the chatbot's adaptability and scalability by integrating reinforcement learning for continuous model improvement based on user feedback. Expanding multilingual support will enable broader accessibility, allowing the chatbot to process non-English news sources effectively. Further refinements in misinformation detection using fact-checking AI models and sentiment analysis can enhance the chatbot's reliability. Additionally, optimizing computational efficiency through distributed AI processing and cloud-based architectures will enable faster response times, making the system more practical for large-scale deployments

References:

1. Balkus, S.V.; Yan, D. Improving short text classification with augmented data using GPT-3. *Nat. Lang. Eng.* **2022**, *30*, 943–972.
2. Kausar, N.; AliKhan, A.; Sattar, M. Towards better representation learning using hybrid deep learning model for fake news detection. *Soc. Netw. Anal. Min.* **2022**, *12*, 165.
3. Barua, A.; Sharif, O.; Hoque, M.M. Multi-class Sports News Categorization using Machine Learning Techniques: Resource Creation and Evaluation. *Procedia Comput. Sci.* **2021**, *193*, 112–121.
4. Levshun, D.; Kotenko, I. A survey on artificial intelligence techniques for security event correlation: Models, challenges, and opportunities. *Artif. Intell. Rev.* **2023**, *56*, 8547–8590.
5. Pawar, C.S.; Makwana, A. Comparison of bert-base and gpt-3 for marathi text classification. In *Proceedings of the Futuristic Trends in Networks and Computing Technologies: Select Proceedings of Fourth International Conference on FTNCT 2021*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 563–574.
6. Nguyen, T.P.; Carvalho, B.; Sukhdeo, H.; Joudi, K.; Guo, N.; Chen, M.; Wolpaw, J.T.; Kiefer, J.J.; Byrne, M.; Jamroz, T.; et al. Comparison of artificial intelligence large language model chatbots in answering frequently asked questions in anaesthesia. *BJA Open* **2024**, *10*, 100280.
7. Babu, A.; Boddu, S.B. BERT-Based Medical Chatbot: Enhancing Healthcare Communication through Natural Language Understanding. *Explor. Res. Clin. Soc. Pharm.* **2024**, *13*, 100419.

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 21, Issue 2, 2025

8. Hasib, K.M.; Towhid, N.A.; Faruk, K.O.; Mahmud, J.A.; Mridha, M.F. Strategies for enhancing the performance of news article classification in Bangla: Handling imbalance and interpretation. *Eng. Appl. Artif. Intell.* **2023**, *125*, 106688.

9. Maham, S.; Tariq, A.; Khan, M.U.G.; Alamri, F.S.; Rehman, A.; Saba, T. ANN: Adversarial news net for robust fake news classification. *Sci. Rep.* **2024**, *14*, 7897.

10. Sufi, F.K. AI-GlobalEvents: A Software for analyzing, identifying and explaining global events with Artificial Intelligence. *Softw. Impacts* **2022**, *11*, 100218.

11. Sufi, F.K. Identifying the drivers of negative news with sentiment, entity and regression analysis. *Int. J. Inf. Manag. Data Insights* **2022**, *2*, 100074.

12. Gruenhagen, J.H.; Sinclair, P.M.; Carroll, J.A.; Baker, P.R.; Wilson, A.; Demant, D. The rapid rise of generative AI and its implications for academic integrity: Students' perceptions and use of chatbots for assistance with assessments. *Comput. Educ. Artif. Intell.* **2024**, *7*, 100273.

13. Fatemi, B.; Rabbi, F.; Opdahl, A.L. Evaluating the effectiveness of gpt large language model for news classification in the iptc news ontology. *IEEE Access* **2023**, *11*, 145386–145394.

14. Nicholson, D.N.; Greene, C.S. Constructing knowledge graphs and their biomedical applications. *Comput. Struct. Biotechnol. J.* **2020**, *18*, 1414–1428.