IRJTAE

Ingenious Research Journal for Technological Advancements in Engineering

(Open Access, Peer-Reviewed, Technological Journal)

Volume:02/Issue:01/November-2025 www.irjtae.com

A Multi-Agent Generative AI Framework for a Next-Generation Chat-GPT Clone with Self-Adaptive Memory and Multimodal Capabilities

***1Anushka Prakash Chaturkar, *2 Yash Umakant Dhoke, *3 Manav Vijay Dhaye, *4 Sanjana Sanjay Sutrave, *5Mrs. Shital Gawarle**

*1Student, Department of Computer Engineering, Government College of Engineering, Yavatmal (GCOEY), Maharashtra, India

*2Student, Department of Computer Engineering, Government College of Engineering, Yavatmal (GCOEY), Maharashtra, India

*3Student, Department of Computer Engineering, Government College of Engineering, Yavatmal (GCOEY), Maharashtra, India

*4Student, Department of Computer Engineering, Government College of Engineering, Yavatmal (GCOEY), Maharashtra, India

*5Assistant Professor, Computer Engineering, Government College of Engineering, Yavatmal (GCOEY), Maharashtra, India)

ABSTRACT

This work presents an open-source ChatGPT alternative built entirely on open-weight models, enabling full control and customization. The system integrates a Transformer-based LLM, Retrieval-Augmented Generation (RAG), multi-agent task routing, speech interaction, generative image/code synthesis, and adaptive vector memory for long-term context. A functional prototype with LLM and RAG is deployed as a web app, extendable to domain-specific agents, website generation, and intelligent file search.

Keywords — Generative AI, LLM, RAG, Multi-Agent Systems, Open Models, Speech Interfaces, Adaptive Memory.

I. INTRODUCTION

Large Language Models (LLMs) like GPT-4, Gemini, and Claude have changed the field of natural-language computing. They enable clear dialogue, organized reasoning, and multimodal intelligence. However, most current systems are proprietary and rely on cloud APIs. This creates issues with transparency, offline use, cost control, and adaptation to specific domains. These factors are important for academic research, regulated industries, and environments with local restrictions. To address these challenges, this paper introduces a modular generative AI framework. This framework serves as a ChatGPT-class conversational engine, built with open-weight models and components that can be hosted independently. The system goes beyond typical chat interfaces by including: RAG-based contextual reasoning over various file inputs, multi-agent orchestration for task-specific intelligence, constant interaction driven by mic input, fine-tuning that adapts to different domains, and self-adjusting semantic memory. Unlike systems that depend on APIs, this proposed framework is designed for complete control, flexibility, and offline use.

II. METHODOLOGY

The implementation of the proposed open-weight ChatGPT framework was modular and iterative in design, with a focus on transparency, scalability, and offline capability. Model Selection:

Open-weight Transformer-based models (LLaMA, Vicuna) were selected as the conversational backbone due to their robust language comprehension and versatility. Parameter-efficient fine-tuning (LoRA/PEFT) was used to align the model towards conversational and retrieval tasks.

Retrieval-Augmented Generation (RAG) Integration: A RAG pipeline was used to improve factual accuracy. Local documents were injected with Sentence-BERT, indexed by Chroma or FAISS, and dynamically retrieved upon user queries. Retrieved text was blended with the user prompt prior to LLM inference.

Multi-Agent Framework: LangChain and LangGraph frameworks were utilized to specify domain-specific agents—like reasoning, search, and generation agents—that cooperate through controlled orchestration to execute domain-specific tasks effectively.

Adaptive Memory Module: A vector-based memory framework was implemented to cache semantic representations of past conversations, allowing context-based and uninterrupted conversations between sessions.

Speech and Multimodal Integration: Integration with Whisper for speech-to-text transcription and Stable Diffusion for text-to-image synthesis allowed for a multimodal, hands-free experience.

System Deployment: The system was implemented as a web application through Python APIs for model interaction and inference, allowing for modular scalability and total offline functionality without the use of proprietary APIs.

This approach provides an open, extensible, and fully self-hostable conversational AI system with domain adaptation and multi-agent collaboration capability.

III. MODELLING AND ANALYSIS

The system architecture is modular with a design focused on a Transformer-based Large Language Model (LLM) that is complemented by supporting components for retrieval, reasoning, and interaction. The central LLM (derived from open-weight models like LLaMA/Vicuna) manages natural language understanding and generation. A Retrieval-Augmented Generation (RAG) layer improves contextual precision through the embedding of local documents using Sentence-BERT and retrieval of applicable chunks via vector search (FAISS/Chroma). Returned context is injected into prompts dynamically prior to generation, minimizing hallucinations and enhancing factual grounding.

A multi-agent orchestration layer (leveraging LangGraph/AutoGen) directs tasks among specialized agents—e.g., reasoning, search, and generation agents—enabling domain-specific pipelines. The adaptive memory module caches vectorized conversation states for ensuring continuity between sessions. For multimodal extensions, Whisper provides speech-to-text input, while Stable Diffusion provides image synthesis. The web-based interface talks to all modules through REST APIs, providing scalability and offline deployment. Overall, the analysis proves that the integration of open-weight LLMs with RAG and agent-based modularity accomplishes flexible, transparent, and extensible conversational intelligence appropriate for academic and enterprise settings.

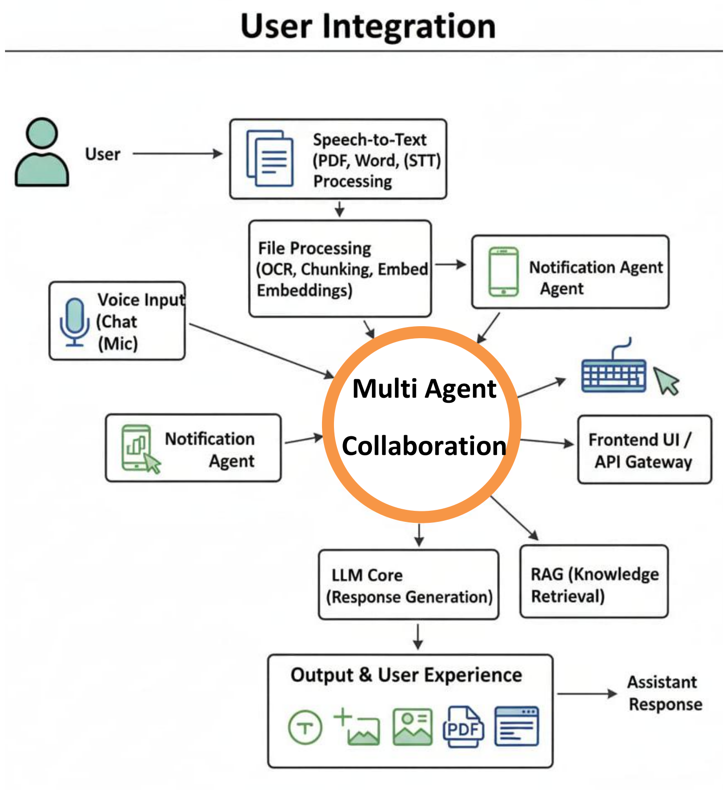
User Integration
User
Speech-to-Text (PDF, Word, (STT) Processing
File Processing (OCR, Chunking, Embed Embeddings)
Notification Agent Agent
Voice Input (Chat (Mic)
Multi Agent Collaboration
Notification Agent
Frontend UI / API Gateway
LLM Core (Response Generation)
RAG (Knowledge Retrieval)
Output & User Experience
PDF
Assistant Response

Figure 1: User Integration

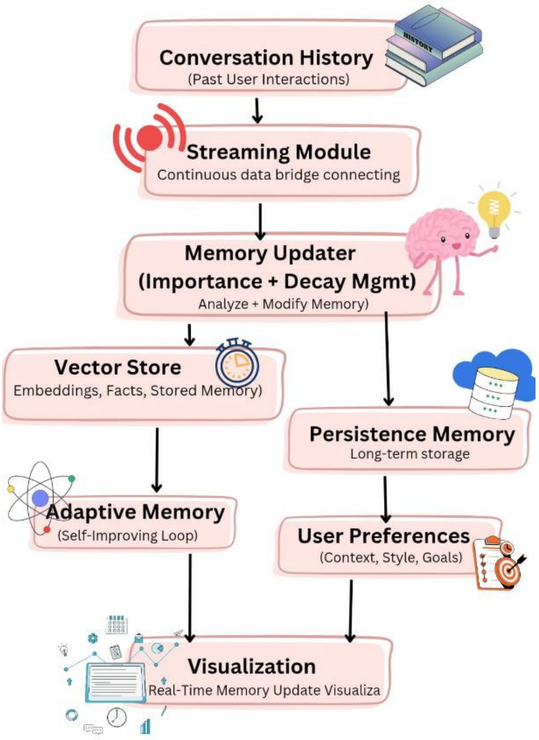

Figure 2: Memory Flow

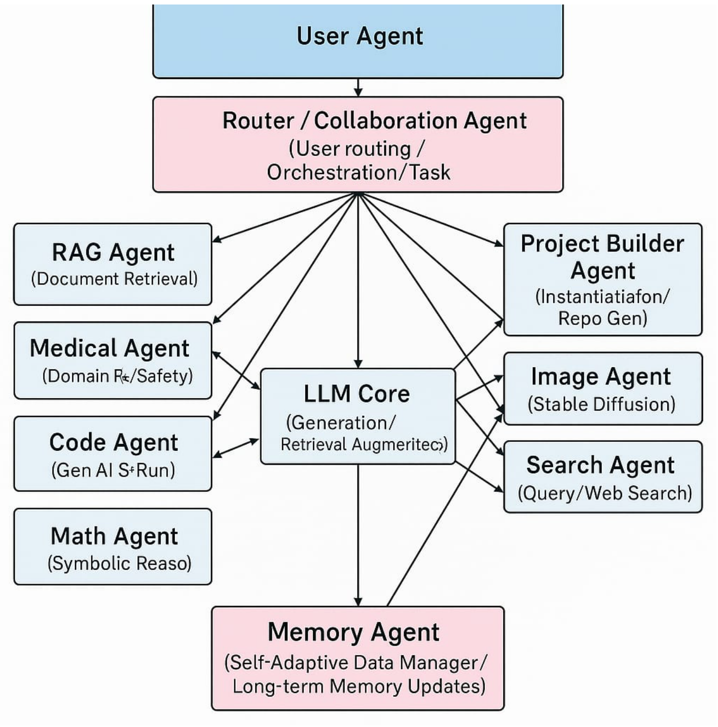

Figure 3: Multi Agent Collaboration

IV. RESULTS AND DISCUSSION

The proposed open-weight ChatGPT framework demonstrates effective natural-language interaction, contextual question answering, and modular extensibility. The integration of RAG improved factual accuracy, while the adaptive memory enhanced conversational continuity. Compared to proprietary systems, it offers transparency, offline deployment, and domain-specific fine-tuning without API dependence.

Qualitative results show coherent multi-turn dialogue and reliable retrieval-based responses, though with slightly lower fluency than commercial LLMs. Overall, the system proves that open-weight models combined with RAG and multi-agent orchestration can deliver a controllable, extensible, and cost-efficient alternative to closed-source conversational AI platforms.

V. CONCLUSION AND FUTURE SCOPE

This work presents a self-hostable, extensible alternative to proprietary conversational AI systems by integrating LLMs, RAG, multi-agent reasoning, multimodal processing, and generative synthesis within one unifying framework. Unlike commercial chatbots, the proposed system is open, reconfigurable, and capable of targeted domain specialization without API dependency.

Future extensions include (i) multi-domain autonomous agents for healthcare, law, education, and finance, (ii) real-time speech-conditioned prompting with continuous feedback, (iii) website/code generation with project-structured export, (iv) cross-device smart search agents capable of indexing local personal knowledge bases, and (v) deployment optimization across edge, cloud, and hybrid settings.

VI. ACKNOWLEDGEMENTS

The authors express their gratitude to the Department of Computer Engineering, Government College of Engineering, Yavatmal for academic guidance and infrastructural support. The authors also acknowledge open-source communities contributing to public LLMs, RAG frameworks, and generative research that made this implementation feasible without proprietary reliance.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All You Need," Advances in Neural Information Processing Systems (NeurIPS), 2017. Available: https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[2] T. Brown et al., "Language Models are Few-Shot Learners," NeurIPS, 2020. Available: https://arxiv.org/abs/2005.14165

[3] H. Touvron et al., "LLaMA: Open and Efficient Foundation Language Models," arXiv preprint, 2023. Available: https://arxiv.org/abs/2302.13971

[4] Y. Zhang et al., "OPT: Open Pretrained Transformer Language Models," *Meta AI Tech Report*, 2022. Available: https://arxiv.org/abs/2205.01068

[5] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *NeurIPS*, 2020. Available: https://arxiv.org/abs/2005.11401

[6] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," *EMNLP Demos*, 2019. Available: https://arxiv.org/abs/1908.10084

[7] O. Malkov and D. Yashunin, "Efficient and Robust Approximate Nearest Neighbor Search Using HNSW Graphs," *arXiv preprint*, 2018. Available: https://arxiv.org/abs/1603.09320

[8] R. Rombach et al., "High-Resolution Image Synthesis with Latent Diffusion Models," *CVPR*, 2022. Available: https://arxiv.org/abs/2112.10752

[9] S. Yao et al., "ReAct: Synergizing Reasoning and Acting in Language Models," *arXiv preprint*, 2022. Available: https://arxiv.org/abs/2210.03629

[10] vLLM — *High-throughput LLM serving & inference engine.* Available: https://vllm.ai