International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 3 (2026)

ijerst.editor@gmail.com

editor@ijerst.com

ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

SELF-EVOLVING AI AGENT WITH SKILL MEMORY

[1] R Uma, [2] B Shravani, [3] M Shivani, [4] C Komal , [5] N Dinesh Kumar
[1]Assistant Professor, [2345]Students
Department of CSE(AIML)
Siddhartha Institute of Technology & Sciences, Narapally
umakola_cse@siddhartha.co.in, 23TQ1A6616@siddhartha.co.in, 23TQ1A6613@siddhartha.co.in, 23TQ1A6663@siddhartha.co.in, 23TQ1A6629@siddhartha.co.in

ABSTRACT

The Self-Evolving AI Agent with Skill Memory is an intelligent software system designed to improve its ability to perform tasks by learning from previous interactions, successful task executions, user feedback, and stored skills. Traditional AI assistants generally respond to individual requests without maintaining a structured memory of how tasks were solved previously. The proposed system introduces a skill-memory mechanism that allows the agent to reuse useful knowledge and improve its future performance.

The system receives a user's task through a conversational interface and analyzes the objective, required actions, available tools, and relevant context. An agentic reasoning module creates a plan and selects appropriate tools or previously learned skills. After completing the task, the system evaluates the result and stores successful procedures, reusable workflows, and important task information in a structured skill memory.

The skill-memory layer contains reusable skills, task procedures, tool configurations, previous outcomes, and performance information. When a similar task is received in the future, the agent can retrieve relevant skills and adapt them to the current situation. This reduces repetitive reasoning and allows the agent to improve its efficiency over multiple interactions.

The proposed system also includes feedback and evaluation mechanisms that -3Edetermine whether a task was completed successfully. Failed or low-quality attempts can be analyzed to identify improvements, while successful workflows can be converted into reusable skills after appropriate validation. Human approval can be included before newly learned skills are permanently added or used for important operations.

Overall, the Self-Evolving AI Agent with Skill Memory provides a framework for creating AI agents that become more capable through controlled learning and experience. It can be applied to software development, IT support, research, productivity, business automation, and other task-oriented environments. Future enhancements can include multi-agent skill sharing, automatic skill generation, reinforcement learning, multimodal memory, long-term planning, and advanced skill verification.

I. INTRODUCTION

Artificial Intelligence systems have become increasingly capable of understanding natural language, reasoning over information, and performing tasks using external tools. Modern AI agents can interact with APIs, databases, applications, files, and other digital systems. However, many agents still treat each task as a new problem and may repeat similar reasoning or actions across different sessions.

Human learning provides an important inspiration for improving intelligent agents. When a person successfully solves a problem, they can remember the procedure and apply it to similar problems in the future. A similar capability can be introduced into AI agents through structured memory that stores reusable knowledge, workflows, and skills.

A **skill memory** is different from simple conversation history. Instead of only remembering what was discussed, the system stores reusable procedures such as how to complete a task, which tools were effective, what parameters were required, and what results were obtained. This allows the agent to retrieve and adapt previously learned skills when encountering similar tasks.

The **Self-Evolving AI Agent with Skill Memory** combines an AI reasoning engine with short-term context, long-term memory, skill retrieval, tool execution, evaluation, and controlled skill updating. The agent first understands a task, retrieves relevant skills, plans its actions, executes the task, evaluates the result, and updates its memory when appropriate.

The objective of the proposed system is to create an AI agent that can improve its task-solving capabilities over time while maintaining appropriate safety and human oversight. The system should not blindly learn every action it performs. Instead, newly created skills should be evaluated, validated, versioned, and subject to permission controls before being reused in future tasks.

II. LITERATURE SURVEY

Traditional AI assistants generally operate using predefined rules, fixed workflows, or static models. Such systems can perform specific tasks effectively but may have limited ability to learn reusable procedures from previous interactions. Updating their capabilities often requires developers to modify the underlying system or retrain models.

Machine learning systems introduced the ability to learn patterns from historical data. Supervised and reinforcement learning approaches allow systems to improve performance based on examples or feedback. However, continuously adapting an AI agent in an open-ended environment introduces challenges related to data quality, safety, evaluation, and unintended behavior.

Memory-augmented AI systems have explored methods for maintaining information across interactions. Short-term memory can maintain the current conversation context, while long-term memory can store information that remains useful across sessions. Retrieval mechanisms can then identify relevant memories when a new task is received.

Retrieval-Augmented Generation has further improved AI systems by allowing language models to retrieve external information before generating responses. Similar retrieval techniques can be applied to skill memory, where the system retrieves previously successful workflows, procedures, and tool-use patterns rather than only retrieving factual documents.

Agentic AI systems combine language models with planning, reasoning, tool usage, and iterative execution. An agent can break a complex task into smaller actions, select tools, observe results, and adjust its plan. However, without persistent skill memory, agents may repeatedly solve similar tasks from scratch.

Recent research directions include self-improving agents, autonomous task learning, reflection, experience replay, procedural memory, and multi-agent learning. These approaches demonstrate the potential for agents to improve from experience but also highlight the need for controlled learning, evaluation, skill versioning, permissions, and human oversight. The proposed **Self-Evolving AI Agent with Skill Memory** combines these concepts into a structured architecture for controlled and reusable agent learning.

III. SYSTEM ANALYSIS

The **Self-Evolving AI Agent with Skill Memory** is designed to perform tasks and improve its capabilities through controlled learning from previous experiences. The system accepts a user's task through a conversational or application interface and analyzes the objective, context, constraints, and required resources. A task-understanding module converts the request into a structured task representation. The skill-retrieval module searches the skill memory for previously stored procedures that are relevant to the current task. An agentic planning engine selects useful skills, tools, and actions to create an execution plan. The tool-execution layer performs approved actions through APIs, applications, databases, files, or other connected services. The agent observes execution results and can revise its plan when an action fails or produces unexpected results. An evaluation module measures task completion, output quality, efficiency, and user feedback. Successful and validated workflows can be converted into reusable skills and stored in the skill-memory repository. A skill-management module handles skill versioning, metadata, confidence, permissions, and lifecycle status. A safety layer prevents unauthorized or unsafe skills from being automatically reused. Finally, monitoring and feedback mechanisms continuously evaluate agent performance and support controlled improvement.

Existing System

Existing AI assistants and task automation systems commonly rely on fixed workflows, predefined tools, static prompts, or a general-purpose language model. These systems can perform many tasks but may not retain detailed knowledge about how previous tasks were completed. When users submit similar requests repeatedly, the agent may perform the same reasoning process instead of reusing an optimized workflow. Traditional conversation memory primarily stores dialogue context rather than reusable task procedures. Rule-based automation systems require developers to manually create and update workflows. Some AI agents can use tools but may not have persistent procedural memory. Failed task executions may not be systematically analyzed for future improvement. Successful workflows may also remain isolated within individual sessions. Existing systems can therefore have limited ability to accumulate practical task-solving experience. They may also lack structured skill versioning and performance tracking. Automatically allowing an AI system to modify its own capabilities can create security and reliability risks. These limitations create a need for a controlled self-evolving agent architecture with persistent skill memory.

Disadvantages of Existing System

- Limited learning from previous task executions.
- Repeated reasoning for similar tasks.
- Limited procedural or skill-based memory.
- Heavy dependence on predefined workflows.
- Manual workflow updates may be required.
- Failed attempts may not be systematically analyzed.
- Successful procedures may not be reused effectively.
- Limited skill version management.
- Limited performance-based skill selection.
- Risk of unsafe autonomous self-modification.
- Limited feedback-driven improvement.
- Difficult to maintain long-term task-solving experience.

Proposed System

The proposed **Self-Evolving AI Agent with Skill Memory** provides an intelligent agent that can learn and reuse validated task-solving skills over time. The system receives a task and first analyzes its objective, context, constraints, and required tools. A skill-retrieval engine searches the long-term skill memory for relevant previously validated procedures. The agentic planning module combines retrieved skills with current task requirements and creates an execution plan. The tool layer performs approved actions through connected applications, APIs, databases, files, or services. During execution, the agent observes results and adapts its plan when necessary. An evaluation module checks whether the task was completed successfully and measures output quality and efficiency. User feedback can also be used to evaluate the usefulness of the generated result. When a workflow demonstrates reliable performance, the system can create or update a reusable skill after validation. The skill repository stores procedures, tool requirements, examples, performance metrics, versions, and permission information. A safety and governance layer controls which skills can be created, modified, retrieved, or executed. Human approval can be required for high-impact or newly generated skills. This architecture enables controlled improvement without allowing unrestricted autonomous modification of the system.

Advantages of Proposed System

- Learns reusable skills from previous experiences.
- Reduces repetitive reasoning.
- Improves efficiency for recurring tasks.
- Provides persistent procedural memory.
- Supports adaptive task planning.
- Reuses validated workflows.
- Learns from task outcomes and feedback.
- Supports skill versioning.
- Tracks skill performance and confidence.

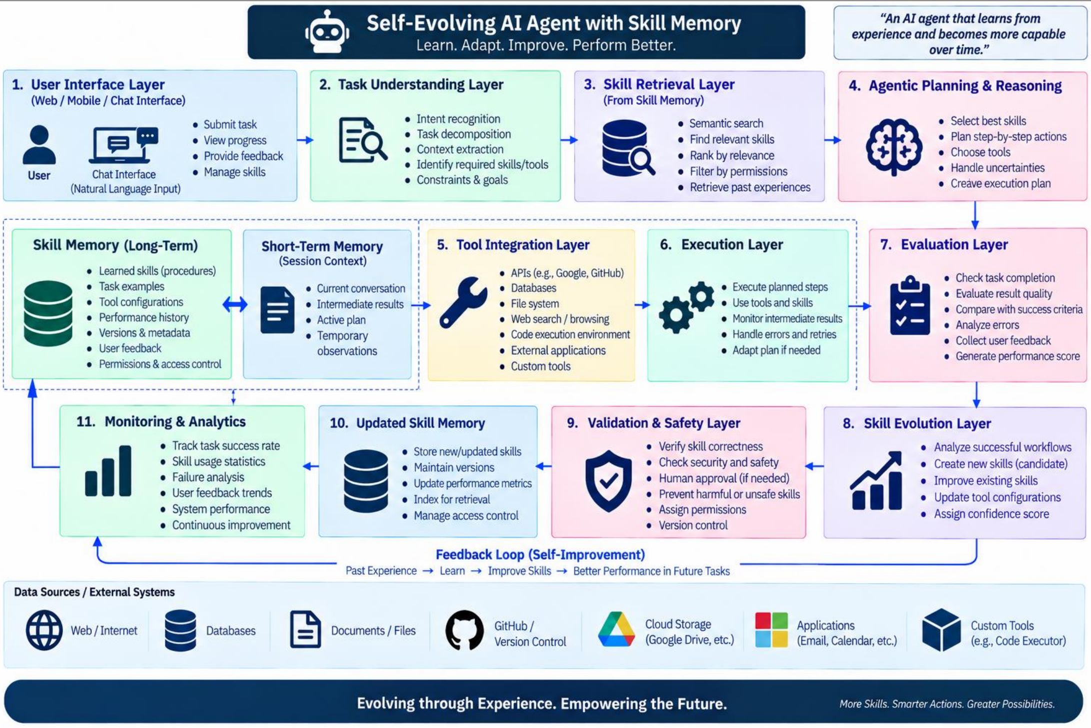
- Provides controlled self-improvement.
- Supports human approval and governance.
- Can improve over extended usage.

IV. METHODOLOGY

The methodology begins when a user submits a task through the AI agent interface. The task-understanding module analyzes the request and extracts the objective, context, constraints, expected output, and required capabilities. The skill-retrieval module searches the skill-memory repository using semantic similarity, metadata, task type, and previous performance information. Relevant skills are selected according to their relevance, reliability, permissions, and compatibility with the current task. The agentic planning engine combines the retrieved skills with available tools and generates an execution plan. The tool-execution layer performs approved actions through APIs, databases, applications, files, or other connected services. The agent observes intermediate results and uses reasoning to continue, modify, or retry appropriate steps. After execution, an evaluation module checks the result using task-specific success criteria, quality measures, and user feedback. Successful workflows can be converted into candidate reusable skills with descriptions, steps, tool requirements, and performance information. A skill-validation module reviews the candidate skill for correctness, safety, permissions, and reliability. Approved skills are stored with version numbers and confidence scores in the long-term skill memory. The monitoring layer tracks skill usage, task success, failures, latency, and user feedback. Finally, the system uses these evaluated experiences to improve future skill retrieval, planning, and task execution while maintaining appropriate human oversight.

SYSTEM ARCHITECTURE

The Self-Evolving AI Agent with Skill Memory consists of interconnected layers that allow the agent to understand tasks, retrieve previous skills, execute actions, evaluate results, and improve its reusable capabilities. The User Interface Layer provides a conversational interface through which users submit tasks and review results. The Task Understanding Layer analyzes natural-language requests and extracts objectives, constraints, context, and required capabilities. The Skill Retrieval Layer searches the long-term skill-memory repository for relevant validated procedures and previous experiences. The Agentic Planning Layer combines retrieved skills with the current task and available tools to generate an execution plan. The Tool Integration Layer provides controlled access to APIs, databases, files, applications, and external services. The Execution and Observation Layer performs actions and collects intermediate results from connected tools. The Evaluation Layer measures task success, output quality, efficiency, and user feedback. The Skill Evolution Layer identifies successful workflows and creates candidate reusable skills after appropriate validation. The Skill Repository stores skill definitions, versions, metadata, permissions, confidence scores, examples, and performance history. The Safety and Governance Layer controls skill creation, modification, retrieval, and execution. The Monitoring Layer tracks agent behavior, failures, performance, and resource usage. Finally, the feedback loop allows validated experience to improve future skill retrieval and task execution.



V. RESULT AND OUTPUT

VI. CONCLUSION

The Self-Evolving AI Agent with Skill Memory provides an intelligent approach for building AI agents that can learn from previous task executions and reuse validated skills in future tasks. Unlike conventional assistants that may solve every request independently, the proposed system maintains a structured memory of useful procedures, tool configurations, previous outcomes, and learned skills.

The system combines Generative AI, agentic reasoning, semantic skill retrieval, tool integration, task execution, evaluation, and long-term memory to improve task-solving efficiency. When a new task is received, the agent retrieves relevant skills, adapts them to the current requirements, executes the planned actions, and evaluates the resulting output.

A major advantage of the system is its controlled skill-evolution mechanism. Successful workflows can be converted into reusable skills, while skill validation, versioning, permissions, and performance tracking help maintain reliability. User feedback and task outcomes can further help the system identify which skills are useful and which require improvement.

Overall, the Self-Evolving AI Agent with Skill Memory demonstrates how AI agents can move beyond one-time task execution toward continuous, controlled improvement through experience. The system can be applied to software development, IT support, research, data analysis, business automation, and personal productivity.

Future enhancements can include multi-agent skill sharing, multimodal memory, reinforcement learning, automatic workflow optimization, advanced reflection mechanisms, real-time skill evaluation, knowledge-graph integration, and improved autonomous planning. Strong safety controls, human approval, monitoring, and permission management should remain essential when the agent is allowed to modify or execute new skills.

REFERENCES

[1] Kumar, R. D., Prudhviraj, G., Vijay, K., Kumar, P. S., & Plugmann, P. (2024). Exploring COVID-19 through intensive investigation with supervised machine learning algorithm. In Handbook of Artificial Intelligence and Wearables (pp. 145-158). CRC Press.

[2] Swathi, B., Vijay, K., Sushanth Babu, M., & Dinesh Kumar, R. (2024, November). Machine Learning Techniques in Cloud Based Intrusion Detection. In The International Conference on Artificial Intelligence and Smart Environment (pp. 557-564). Cham: Springer Nature Switzerland.

[3] Sv satyakrishna, shirisha rangu ,bhargavi nalacheruve.(2024) Prospective investigation on colorectal cancer with SMOTE on machine learning Algorithm

[4] Dr.G.Vishnu Murthy, BhargaviNalacheruve 1Professor, Department of computer Science & engineering, Anurag University, TS, India. 2Student, Department of computer Science & engineering, Anurag University, TS, India.

[5] V. N. S. Manaswini, K. K, C. Nigam, S. S. Ali, R. Niranjana, and Suman, "Real-Time Object Detection in Drone Surveillance Using YOLOv5," in Proc. 2025 3rd Int. Conf. IoT, Communication and Automation Technology (ICICAT), Gorakhpur, India, 2025, pp. 1–6, doi: 10.1109/ICICAT68430.2025.11414670.

[6] B. Soundarya, V. N. S. Manaswini, M. Ayyakrishnan, R. D. Kumar, "Contextual Analysis of Big Data Analytics in Intelligent Transportation Frameworks," in Intersection of Artificial Intelligence, Data Science, and Cutting-Edge Technologies:

From Concepts to Applications in Smart Environment, Lecture Notes in Networks and Systems, vol. 1353, Cham: Springer, 2025, doi: 10.1007/978-3-031-88304-0_79.
[7] R. D. Kumar, V. N. S.Manaswini, “Applications of blockchain in smart cities: detecting fake documents from land records using blockchain technology,” in Blockchain for Smart Cities, Elsevier, 2021, pp. 105–117, doi: 10.1016/B978-0-12-824446-3.00017-X.
[8] Tejavath Veeramma, Badarla Anil, Guguloth Ravinder, “An advanced movie recommender using collaborative filtering and sentiment analysis,” International Research Journal of Modernization in Engineering Technology and Science, vol. 7, no. 7, July 2025, doi: 10.56726/IRJMETS81618.
[9] Ravi Kumar Banoth, Ramana Murthy B V, “Automatic crop recommendation system using LightGBM and decision tree machine learning models,” Journal of Machine and Computing, vol. 5, no. 1, pp. 343, Jan. 2025, doi: 10.53759/7669/jmc202505026.
[10] Ravi Kumar Banoth, Dr. B.V. Ramana Murthy, “Smart agriculture through IoT and machine learning for analyzing carbon footprints,” in Proc. Int. Conf. Computer Science and Communication Engineering (ICCSCE), Apr. 2025.
[11] Ravi Kumar Banoth, B. V. Ramana Murthy, “Soil image classification using transfer learning approach: MobileNetV2 with CNN,” SN Computer Science, vol. 5, art. no. 199, 2024, doi: 10.1007/s42979-023-02500-x.
[12] V. N. S. Manaswini, K. K, C. Nigam, S. S. Ali, R. Niranjana, and Suman, “Real-Time Object Detection in Drone Surveillance Using YOLOv5,” in Proc. 2025 3rd Int. Conf. IoT, Communication and Automation Technology (ICICAT), Gorakhpur, India, 2025, pp. 1–6, doi: 10.1109/ICICAT68430.2025.11414670.
[13] B. Soundarya, V. N. S. Manaswini, M. Ayyakrishnan, R. D. Kumar, “Contextual Analysis of Big Data Analytics in Intelligent Transportation Frameworks,” in Intersection of Artificial Intelligence, Data Science, and Cutting-Edge Technologies: From Concepts to Applications in Smart Environment, Lecture Notes in Networks and Systems, vol. 1353, Cham: Springer, 2025, doi: 10.1007/978-3-031-88304-0_79.
[14] R. D. Kumar, V. N. S. Manaswini, “Applications of blockchain in smart cities: detecting fake documents from land records using blockchain technology,” in Blockchain for Smart Cities, Elsevier, 2021, pp. 105–117, doi: 10.1016/B978-0-12-824446-3.00017-X.
[15] M. M. Rao, M. B. Mohiuddin, P. Srinu, A. Satyanarayana, J. Madhavan and R. D. Kumar, "Carbon-Neutral-Agriculture: Renewable-Energy Powered IoT Systems and Resource Optimization," *2026 3rd International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE)*, Chennai, India, 2026, pp. 1-6, doi: 10.1109/RMKMATE69073.2026.11518903.
[16] R. Uma, M. Maggidi, A. R. Ekkati, S. Bandlamudi, C. Madugundi and S. K. Taduri, "Zero-Trust Architecture for Hybrid Cloud Security in Critical Infrastructures," 2026 IEEE International Conference for Convergence in Computing Technology (I3CTCON), Lonavala, India, 2026, pp. 1-6, doi: 10.1109/I3CTCON68242.2026.11507458.
[17] VNS Manaswini, Chithanoor Arjun, Yamini Chouhan, & V. Sumalatha. (2026). MB-EFF: A Multi-Branch Environmental Feature Fusion Deep Learning Model for Crop Classification and Recommendation. *International Journal of Computer Information Systems and Industrial Management Applications*, *18*(22s), 460–468. https://doi.org/10.70917/ijcisim-2026-5453

[18] Yamini Chouhan, SIDDOJU DIVYA, VNS Manaswini, & J. Priyanka. (2026). Vision-Threatening Diabetic Retinopathy Detection Using EfficientNetB4: A Two-Stage Transfer Learning Approach. *International Journal of Computer Information Systems and Industrial Management Applications*, *18*(22s), 469–476. https://doi.org/10.70917/ijcisim-2026-5454

[19] Real Time Vehicle Counting and Classification system, **Dr. A.Satyanarayana,et.al** International Research Journal of Moderrnization in Engineering Technology and Science, Vol-06, Issue-05, 2024 **[ISSN: 2582-5208] doi:** 10.56726/IRJMETS57804

[20] Facial Depth Maps: Detecting Sleep Apnea with Deep Learning, **Dr. A.Satyanarayana,et.al** , International Journal of Scientific Development and Research, Vol 9, Issue-5, 2024 **[ISSN: 2455-2631]**

[21] Moodchef: Mood Based Smart Recipe Recommender with cooking Assistance, **Dr. A.Satyanarayana,et.al ,** International Journal of Progressive Research in Engineering Management and Sciences, Vol-05,Issue-7, July 2025 **[ISSN: 2583-5162]** doi: **10.58257/IJPREMS42696**

[22] News Article Text Summarization, **Dr. A.Satyanarayana,et.al,** Journal of Emerging Technologies and Innovative Research, Vol-11, Issue -5, 2024, **[ISSN: 2349-5162]**

[23] Voice Assistant and gesture controlled virtual mouse using Deep learning Techniques, **Dr. A.Satyanarayana,et.al ,** International Research Journal of Moderrnization in Engineering Technology and Science Vol-6, Issue-5, 2024 **[ISSN: 2582-5208] 10.56726/IRJMETS57214**