PromptX: A Cognitive Agent Platform with Long-term Memory

Binhao Wang
City University of Hong Kong
Hong Kong SAR., China
binhao.wang@my.cityu.edu.hk

Jianglin Huang
Deepractice AI Limited
Hong Kong SAR., China
danny@deepractice.ai

Xiao Hu
Deepractice AI Limited
Hong Kong SAR., China
dason@deepractice.ai

Shan Jiang*
Deepractice AI Limited
Hong Kong SAR., China
sean@deepractice.ai

Maolin Wang†
City University of Hong Kong & Deepractice AI Limited
Hong Kong SAR., China
morin.wang@my.cityu.edu.hk

Ching-ho Yang
Deepractice AI Limited
Hong Kong SAR., China
yangqinghe@deepractice.ai

Jian Jiang
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
jiangjian@deepractice.ai

Junhao Ye
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
yejunhao@deepractice.ai

Yaozu Cen
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
cenyaozu@deepractice.ai

Rui Zeng
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
zengrui@deepractice.ai

Yingtong Zhou
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
zhouyingtong@deepractice.ai

Yingjie Luo
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
zhouyingtong@deepractice.ai

Guanjie Wu
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
wuguanjie@deepractice.ai

Wangzhong Xu
Deepractice Artificial Intelligence Technology Co., Ltd.
Changsha, Hunan, China
xuwangzhong@deepractice.ai

Feiyu Zhou
New York University
New York, NY, USA
fz2176@nyu.edu

Xiangyu Zhao
City University of Hong Kong
Hong Kong SAR., China
xy.zhao@cityu.edu.hk

Abstract

While large language models (LLMs) demonstrate impressive contextual understanding, their limitations in long-term memory and personalized reasoning constrain their practical impact in industrial settings. To address these gaps, we introduce **PromptX**, a cognitive platform that enables AI agents to construct structured memory and develop their reasoning over time. PromptX integrates three core technologies: (1) A new prompt markup language to define agent personas and memory organization; (2) Engram-based activation-diffusion memory networks that unify raw experiences with conceptual sequences, enabling associative retrieval through graph network propagation; (3) a protocol-driven orchestration layer enabling dynamic tool discovery and coordination, inspired by **HATEOAS** principles from web-engineering. During five months of real-world deployment across a range of 5 enterprises in 6 industries„ PromptX has been validated in multiple industry domains (e.g., software engineering, education, healthcare), accumulating **50K+** downloads and **3K+** GitHub stars and evidencing practical feasibility and commercial value in production workflows. Our demo and initial product are available at **https://promptx.deepractice.ai/**. The source code and documentation are available online at *https://github.com/Deepractice/PromptX*. The **supplementary materials** are also available online [1].

*Core contributor
†Corresponding author

WWW Companion'26, Dubai, UAE

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

CCS Concepts

• **Human-centered computing → Interaction design**; • **Information systems → Web applications**.

Keywords

AI Agents, Memory Networks, Agent Context Protocol

ACM Reference Format:
Binhao Wang, Jianglin Huang, Xiao Hu, Shan Jiang, Maolin Wang, Ching-ho Yang, Jian Jiang, Junhao Ye, Yaozu Cen, Rui Zeng, Yingtong Zhou, Yingjie Luo, Guanjie Wu, Wangzhong Xu, Xiangyu Zhao. 2026. PromptX: A Cognitive Agent Platform with Long-term Memory. In *Proceedings of The ACM Web Conference 2026 (WWW Companion'26).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

[1]https://drive.google.com/drive/folders/1wPhEQCKeaZIsQAcFnC7XKmXTtwY1gJL1?usp=sharing

1 Introduction

Despite the power of Large Language Models (LLMs), their inherent limitations—including high computational costs, catastrophic forgetting, and a deficiency in logical reasoning [10]—fundamentally constrain their potential in real-world applications requiring persistent knowledge accumulation. Consequently, while Retrieval-Augmented Generation (RAG) has become the dominant paradigm, recent work has made it clear that a shift from simple "retrieval augmentation" to true "memory systems" is necessary for continual learning [5, 6]. This aligns with industry analysis suggesting that successful AI agents depend more on structured data organization than on purely algorithmic breakthroughs [2].

However, current RAG and its variants still face significant challenges. Basic RAG methods rely on "flat" text chunk retrieval, which prevents them from capturing complex inter-dependencies and leads to fragmented answers [4]. This issue is part of a broader challenge of cognitive fragmentation, where the prompts used to guide agents are brittle and difficult to manage, hindering the construction of complex behaviors [8]. Even advanced Knowledge Graph-Augmented RAG (KG-RAG) faces a difficult trade-off: a choice between efficient but rigid static graphs, and flexible but slow and expensive dynamic graph traversal [7].

To break through these limitations, we advocate a paradigm shift from "retrieval augmentation" to a "cognitive architecture" inspired by human memory mechanisms [10]. While existing solutions like LangChain and MemGPT offer fragmented approaches, PromptX provides a unified platform built on three core innovations: 1) Prompt Markup Language (PML) for defining a parsable, structured cognitive model; 2) an associative memory network that operationalizes the cognitive process of "ecphory" [3, 7] for dynamic and long-term multi-hop retrieval; and 3) autonomous tool discovery via the Agent Context Protocol (ACP), leveraging hypermedia principles for navigation and action [9].

This work makes three primary contributions:

- We propose the first open-source implementation integrating PML-based cognitive architecture, Engram activation-diffusion memory, and the ACP protocol [1], providing production-ready support for AI applications like Claude and Cursor.
- We demonstrate reproducible end-to-end scenarios including conversational role evolution and autonomous tool orchestration, validated by **system internals and algorithms**.
- We provide comprehensive real-world deployment evidence spanning 5 months, 50K+ downloads, 15+ enterprises, and 6 industry verticals, proving engineering feasibility and commercial value.

2 Design and Framework

PromptX unifies cognitive structure, associative memory, and autonomous tool orchestration into a three-layer architecture. The *identity layer* defines machine-parsable personas through PML syntax; the *memory layer* implements graph-based activation for conceptual retrieval beyond text similarity, and the *capability layer* enables hypermedia-driven tool discovery.

2.1 Core Capabilities

Conversational Role Creation (Nuwa). We propose a human-AI collaboration paradigm **ISSUE** designed to structure intelligent teamwork, which consists of five steps: **I**nitiate (humans define the problem and set priorities), **s**tructure (select an appropriate methodology or framework), **S**ocratic (AI engages in guided questioning to refine understanding), **U**nify (integrate insights into a coherent plan), and **E**xecute (transform the plan into actionable tasks). Guided by the ISSUE, the role creation engine creates PML roles in 2-3 minutes via 3-5 questions, automatically generating structured cognitive architectures with modular components for reuse and version control.

Rapid Tool Integration (Luban). Integrate any API into AI-callable tools within 3 minutes, generating capability specifications with security constraints, validated in sandbox before registration for dynamic discovery.

Engram Memory Networks. Memory units containing four fields: *content* (raw experience), *schema* (conceptual sequence), *strength* (importance), *type* (ATOMIC/LINK/PATTERN). These units are organized in a graph database, where schemas define how concepts connect to one another. A graph neural network operates on the structure, allowing information to spread across related nodes - a process we called activation-diffusion - so that the system can retrieve associated memories through conceptual relationships, rather than relying on embedding similarity.

2.2 System Architecture

PromptX adopts a three-layer architecture (Figure 1). The *Clients* in Service Layer provides Desktop (Electron), CLI (Node.js) and API clients. The *Server* implements protocol parsing and request routing. The *Cognitive Engine* in Context Engineering Layer contains Nuwa and Luban. The *Memory* provides PML-based Repository, Engram Database, Graph Networks and Sandbox. The *Context Assembler* integrates memory retrieval, persona instructions, tool feedback and session episodes. The *Foundation LLM* supports multiple LLMs.

2.3 PML Parsing Mechanism

PML (Prompt Markup Language) declares cognitive architecture as machine-parsable XML documents repo. The PML ***ContentParser*** parses role files to extract reference arrays, ***ResourceManager*** recursively loads thought/execution/knowledge files, and ***SemanticRenderer*** composes unified prompts. The reference mechanism implements single source of truth: each concept is defined once, modifications propagate globally, avoiding duplication. For example, a query optimizer role referencing an “explain-plan-analysis” thought file can be reused by multiple related roles; updating that file immediately affects all referrers.

2.4 ACP and Tool Definitions

PromptX introduces the **A**gent **C**ontext **P**rotocol (**ACP**), a service-oriented extension of the MCP paradigm that transforms AI models from passive tool operators into active, professional agents. Unlike traditional methods that hardcode tool lists, ACP is inspired by HATEOAS principles, enabling dynamic tool discovery. It provides context-aware links that guide the agent to appropriate tools based on its current state. Responses contain an `available_actions` array, where each action specifies its relationship (*rel*), endpoint (*href*), and parameters. This allows the AI to autonomously match

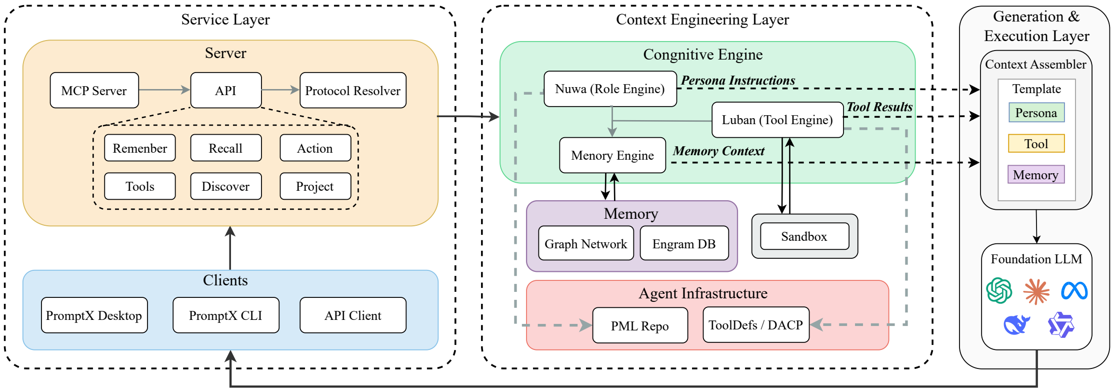

Figure 1: The System Architecture of PromptX. The diagram illustrates the three primary layers of our framework: the Service Layer for handling client interactions, the Context Engineering Layer for cognitive processing and memory management, and the Generation & Execution Layer for producing the final output.

Algorithm 1 Remember Agent: Experience Encoding

Require: Raw experience E, Role R
Ensure: Engram $G = (c, s, w, t)$

$c \leftarrow E$ ▷ Preserve original content
$keywords \leftarrow \textsc{ExtractKeywords}(E)$
$s \leftarrow \textsc{Join}(keywords)$ ▷ Schema as keyword sequence
$w \leftarrow \textsc{ComputeStrength}(E, R)$ ▷ Recency, frequency, importance
$t \leftarrow \textsc{ClassifyType}(E)$ ▷ ATOMIC/LINK/PATTERN
$nodes \leftarrow \textsc{ExtractNodes}(s)$
for all pairs (n_i, n_j) co-occurring in s **do**
    $\textsc{AddEdge}(n_i, n_j, \text{weight})$
end for
$\textsc{UpdateCentrality}(Graph)$
return $G = (c, s, w, t)$

---

needs to tool capabilities based on dialogue context, shifting the interaction from merely "using tools" to "delegating tasks."

This design boasts key advantages: dynamic tool discovery for runtime extension (no code changes), context/capability-based invocation (not static signatures), and auditable, traceable actions. Fundamentally, ACP adds the missing procedural accountability to existing frameworks, making LLM agents process-bound service entities rather than stateless APIs.

2.5 Core Flow: Memory System

PromptX's memory system is managed by two core agents that are integrated into Server API: **Remember** and **Recall**. As illustrated in **Algorithm** 1, the **Remember agent** transforms raw experiences into structured Engrams, our fundamental memory units. Each Engram encapsulates the original content, its conceptual schema (extracted keywords), and its calculated importance. These schemas are interconnected within a graph network, forming the structural basis for associative memory.

The **Recall** agent retrieves relevant memories not through simple keyword matching, but via a graph-based activation-diffusion process. When a query is received, activation spreads from the query's core concepts through the graph network across multiple hops. This allows the system to uncover causally or conceptually related Engrams, even if they are not textually similar. For example, given the query "orders slow," the system can traverse the graph to recall a three-day-old Engram suggesting an "index optimization," demonstrating associative reasoning that goes far beyond direct keyword search. Detailed algorithms and implementation specifics are available in our online documentation and project repository.

3 Demonstration Scenario

An end-to-end scenario demonstrates PromptX's capabilities through a user, Bob, who creates a stock analysis agent with long-term memory from scratch[2].

Step 1: Rapid Tool Integration (t=0–2.5 min). Bob first activates **Luban**, the tool creation expert, requesting: "I need a tool that can query real-time stock data from Alpha Vantage." **Luban** autonomously researches the API documentation, completes the tool's creation, and *completes validation via a dry-run test* within 3 minutes, all without manual coding.

Step 2: Conversational Role Creation (t=2.5–4 min). Next, Bob activates **Nuwa**, the role creation expert, instructing: "Create a stock trading character and use the tool you just created." **Nuwa** understands the composite request, automatically plans and executes a series of tasks, and ultimately generates a new "Stock Trading Analyst" AI role bound to the new tool.

Step 3: Task Execution & Memory Formation (t=4–7 min, Session 1). Bob activates the new role and informs it of his holdings: "I currently hold Tesla and Amazon stocks, I hope to make a profit." The agent analyzes the stocks and saves this core fact ("I hold TSLA and AMZN") into memory as a structured *Engram*.

Step 4: Cross-Session Memory Recall (t=2 hours later, Session 2). To verify its long-term memory, Bob starts a *new session* and issues a compound command: "Activate the stock analyst, first recall the stocks I hold, then analyze today's market." The agent successfully *precisely recalls* Bob's portfolio (Tesla and Amazon)

[2]A video walkthrough of this scenario is available at: **https://youtu.be/R6ENaj9i0oE**

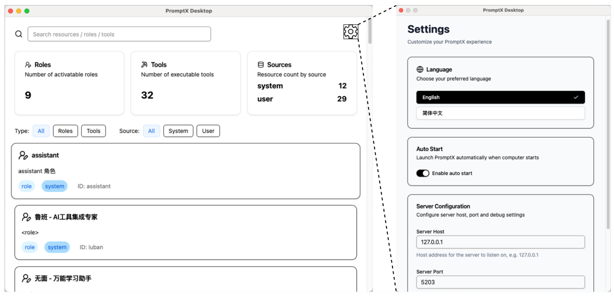

Figure 2: PromptX Desktop Interface. The main dashboard (left) provides dynamic resource discovery with 9 activatable roles, 32 executable tools, and hierarchical source organization (system/user). The settings panel (right) demonstrates multi-language support and server configuration, showcasing the system's production-ready deployment capabilities.

from its memory network and then provides a fresh market analysis focused only on those two stocks, perfectly continuing the context.

Comparison. This scenario highlights key differences from existing approaches. *Stateless LLMs* would lose all context in a new session. *Traditional RAG systems* struggle to accurately and efficiently extract core facts from unstructured history. In contrast, **PromptX**, through its structured memory system, achieves precise cross-session knowledge recall and application, demonstrating a more robust and human-like long-term memory.

4 Implementation and Deployment

The PromptX demo is implemented in Node.js/TypeScript with an MCP server supporting streaming HTTP and PML parsing via xml2js. Its memory system supports both filesystem (JSON) and database (SQLite/PostgreSQL) storage, utilizing graph networks through in-memory adjacency lists and optimized activation-diffusion algorithms. The tool sandbox operates on Docker with 0.5 CPU cores, 512MB memory, and 30s timeout under whitelisted network access. Both Desktop (Electron) and CLI (Commander.js) clients are available. Released under MIT license, PromptX is available as a Docker image (`deepracticexs/promptx:latest`). The interface design is illustrated in Figure 2.

PromptX underwent a 5-month deployment across 15+ enterprises in 6 industries, **attracting over 3K+** GitHub stars and **50K+** downloads. Applications span programming, law, education, tourism, fiction writing, and medicine. Table 1 summarizes key industrial deployments, integrating quantitative outcomes with qualitative feedback from our partners. Specific application certification documents are included in the **supplementary materials**[3].

The system's technical design was further validated by the open-source community, with one developer noting: *"The activation-diffusion retrieval finds conceptual connections I didn't explicitly program—it reasons about relationships."*

5 Conclusion

PromptX demonstrates a paradigm shift from retrieval-augmented generation to cognitive architecture, transforming AI agents from stateless responders into persistent collaborators capable of accumulating knowledge, developing expertise, and maintaining long-term relationships with users. Our open-source implementation integrates three technical innovations: PML for machine-parsable cognitive architecture, Engram activation-diffusion memory unifying raw experiences with conceptual sequences for associative reasoning, and MCP+HATEOAS for hypermedia-driven tool discovery supporting zero-configuration extension. Real-world deployment across 15+ enterprises, 6 industries, 5 months, and 50K+ downloads validates engineering feasibility and commercial value. Future work includes multi-agent collaboration, reinforcement learning for memory refinement, and cloud support for sharing roles and workflows. The community is invited to contribute at **https://github.com/Deepractice/PromptX**, to advance AI agents that can remember, learn and evolve.

[3]https://drive.google.com/drive/folders/1wPhEQCKeaZIsQAcFnC7XKmXTtwY1gJL1?usp=sharing

Table 1: Real-World Deployment Cases and Feedback

Industry	Application	Reported Impact & Feedback
Tourism	End-to-end Content & service agent pipeline	Cost -30%, revenue 2×. *"Created 6 specialized agents... in one afternoon , **code-free**."*
Education	Memory-augmented AI teaching assistant	Personalized tutoring. *"Memory networks enable AI to remember each student's learning trajectory... Students report feeling 'understood'."*
Consulting	Sales knowledge systematization and onboarding automation	Recruitment acceleration. *"Reduced ramp-up time from 6 months to 6 weeks... Impossible with traditional RAG."*

References

[1] Anthropic. 2024. Model Context Protocol (MCP) Specification. https://modelcontextprotocol.io/.
[2] CB Insights. 2025. AI Agent Bible: The Essential Guide to Agentic AI. https://www.cbinsights.com/research/ai-agent-bible/.
[3] Allan M. Collins and Elizabeth F. Loftus. 1975. A spreading-activation theory of semantic processing. *Psychological Review* 82, 6 (1975), 407–428.
[4] Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. 2024. LightRAG: Simple and Fast Retrieval-Augmented Generation. (2024). arXiv:2410.05779 [cs.IR]
[5] Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. HippoRAG: Neurobiologically Inspired Long-Term Memory for Large Language Models. In *Proceedings of the Thirty-eighth Annual Conference on Neural Information Processing Systems (NeurIPS '24)*.
[6] Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From RAG to Memory: Non-Parametric Continual Learning for Large Language Models. *arXiv preprint arXiv:2502.14802* (2025). arXiv:2502.14802 [cs.CL]
[7] Zirui Liao. 2025. EcphoryRAG: Re-Imagining Knowledge-Graph RAG via Human Associative Memory. arXiv:2510.08958 [cs.AI]
[8] Hari Subramonyam, Roy Pea, Christopher Pondoc, Maneesh Agrawala, and Colleen Seifert. 2024. Bridging the Gulf of Envisioning: Cognitive Challenges in Prompt Based Interactions with LLMs. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. ACM, Article 1039, 19 pages.
[9] Danai Vachtsevanou, Jérémy Lemee, Raffael Rot, Simon Mayer, Andrei Ciortea, and Ganesh Ramanathan. 2023. HyperBrain: Human-inspired Hypermedia Guidance using a Large Language Model. In *Proceedings of the 34th ACM Conference on Hypertext and Social Media*. 1–5.
[10] Qinghua Zheng, Huan Liu, Xiaoqing Zhang, Caixia Yan, Xiangyong Cao, Tieliang Gong, Yong-Jin Liu, Bin Shi, Zhen Peng, Xiaocen Fan, Ying Cai, and Jun Liu. 2025. Machine Memory Intelligence: Inspired by Human Memory Mechanisms. *Engineering* (2025). doi:10.1016/j.eng.2025.01.012