# Context-Rich Adaptive Embodied Agents: Enhancing LLM-Powered Task Planning and Memory in Home Robotics

Yutian Gai and Haoyu Cen *

Posted Date: 5 March 2026

doi: 10.20944/preprints202603.0362.v1

Keywords: embodied AI; Large Language Models; home robotics; task planning; contextual memory

 doi:10.20944/preprints202603.0362.v1



*Article*

# Context-Rich Adaptive Embodied Agents: Enhancing LLM-Powered Task Planning and Memory in Home Robotics

**Yutian Gai and Haoyu Cen ***

Polytechnic Colleges, Malaysia
* Correspondence: me01084755@student.uniten.edu.my

**Abstract**

The rapid evolution of Embodied AI and Large Language Models presents significant opportunities for home robotics, yet challenges persist in enabling robots to execute long-term, high-level natural language instructions. Current LLM-driven embodied agents often suffer from sub-optimal task planning, limited memory systems struggling with multi-hop queries, and inflexible agent routing mechanisms. To address these limitations, we propose the Context-Rich Adaptive Embodied Agent (CRAEA) framework, designed to significantly enhance task planning and memory-augmented question answering in household robots. CRAEA integrates core components: Semantic-Enhanced Task Planning (SETP), which enriches LLM-driven planning with object relationship graphs, hierarchical strategies, and implicit physical constraints; Multi-Modal Contextual Memory (MMCM), which stores comprehensive contextual memory units in a relational graph for sophisticated multi-hop reasoning and employs an advanced retrieval mechanism with temporal decay; and Adaptive Agent Routing and Coordination (AARC), featuring intent recognition with confidence evaluation, proactive clarification, and a planning feedback loop. Evaluated in an artificial home environment across complex tidying scenarios, CRAEA consistently demonstrates superior performance. Empirical results show that CRAEA achieves notable improvements in Task Planning Accuracy, Knowledge Base Response Total Validity, and Agent Routing Success Rate compared to baseline methods. A human evaluation further confirms enhanced coherence, naturalness, and user satisfaction, while an ablation study validates the critical contribution of each proposed module. CRAEA represents a significant step towards more intelligent, robust, and user-adaptive home robots.

**Keywords:** embodied AI; Large Language Models; home robotics; task planning; contextual memory

## 1. Introduction

The rapid advancements in Embodied AI and Large Language Models (LLMs) have opened unprecedented opportunities for the field of home robotics. A critical challenge in smart homes and robotics is enabling domestic robots to execute long-term, high-level natural language instructions in complex, dynamic environments, such as "please clear the dining table after dinner." This requires not only robust perception capabilities to identify and understand objects but also sophisticated *task planning abilities* to translate abstract commands into concrete action sequences, alongside efficient *memory and reasoning capabilities* to answer retrospective queries based on historical actions, such as "where was that cup placed?" or "what's in the trash bin?" Beyond these high-level AI capabilities, the successful deployment of such robots also inherently relies on robust underlying hardware control and system stability, including advanced motor control techniques vital for precise and efficient physical interaction [1–3].

Existing research has begun exploring integrated systems that leverage LLMs for both task planning and knowledge-based question answering [4]. The underlying architectures of these LLMs,

 doi:10.20944/preprints202603.0362.v1



often transformers, and their inherent mechanisms like in-context learning, are crucial for their effectiveness in such complex tasks [5,6]. For instance, recent work proposed an LLM-driven agent orchestration system that integrates task planning, memory-augmented Q&A, and multi-agent routing, demonstrating preliminary effectiveness in household tidying scenarios [7]. This system achieves task execution and knowledge recall without explicit model training by employing open-source local LLMs (e.g., Qwen2.5-32B) combined with Retrieval-Augmented Generation (RAG) techniques.

However, current solutions still face several limitations when confronted with highly contextualized and complex interactions. *Task planning* often lacks a deeper understanding of semantic relationships between objects and physical constraints, leading to sub-optimal or implausible plans. The *memory system* primarily relies on simple QA chunk retrieval, struggling with multi-hop reasoning or understanding object state changes over time in complex queries. Furthermore, *agent routing* can be inflexible due to a lack of profound task and environmental understanding. This research aims to further enhance the intelligence and robustness of household robots in long-term tidying tasks.

To address these limitations, we propose a novel approach named **Context-Rich Adaptive Embodied Agent (CRAEA)**. CRAEA enhances an existing agent orchestration system by focusing on three key improvements: Firstly, we introduce **Semantic-Enhanced Task Planning (SETP)** which enriches the LLM-driven planning agent with more comprehensive semantic information and contextual awareness. This involves building an object relationship graph, employing hierarchical semantic planning, and implicitly encoding physical constraints through refined prompt engineering. Secondly, we develop a **Multi-Modal Contextual Memory (MMCM)** system that stores and retrieves richer multi-modal contextual information beyond simple QA chunks. This is achieved through contextual memory units, a relational memory graph, and an advanced retrieval mechanism combining semantic similarity with temporal decay. Lastly, we implement **Adaptive Agent Routing and Coordination (AARC)** to make the routing agent more intelligent and self-adaptive. AARC incorporates deeper intent recognition with confidence evaluation, proactive clarification mechanisms, and a planning feedback loop to ensure robust task execution.

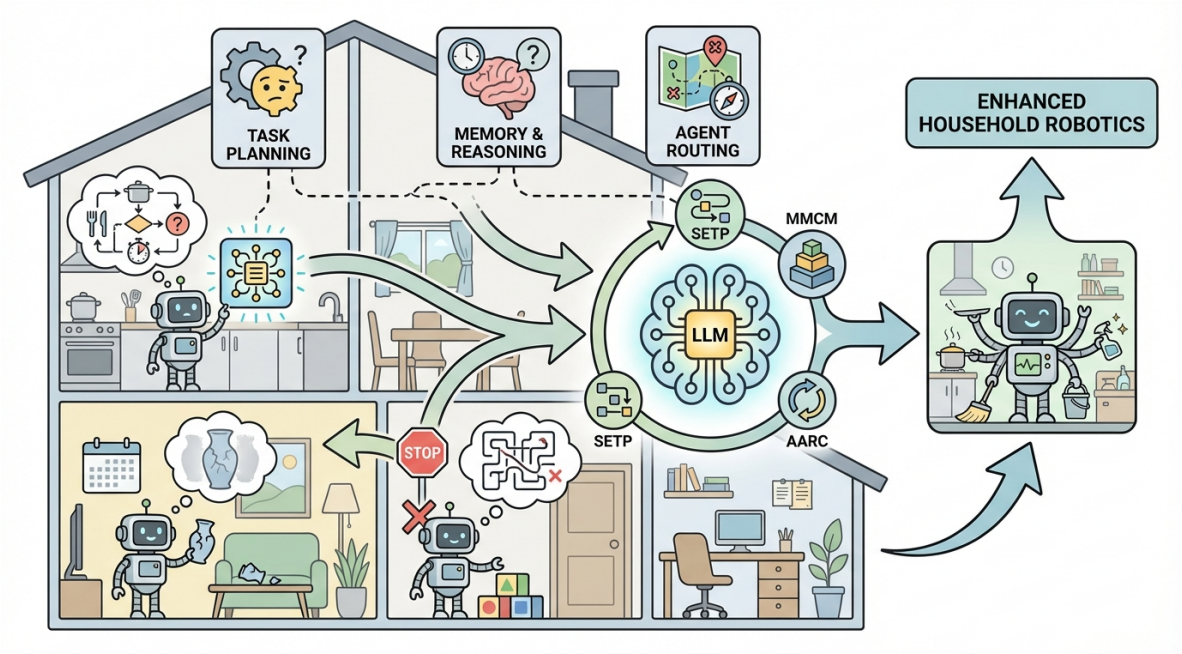


**Figure 1.** An overview of the Context-Rich Adaptive Embodied Agent (CRAEA) framework. The left panel illustrates common challenges in household robotics, including sub-optimal task planning, limited memory and reasoning, and inflexible agent routing. Our proposed CRAEA framework, centered around an LLM, integrates Semantic-Enhanced Task Planning (SETP), Multi-Modal Contextual Memory (MMCM), and Adaptive Agent Routing and Coordination (AARC) to address these limitations, leading to enhanced intelligence and robustness in household robotics, as depicted on the right.

 doi:10.20944/preprints202603.0362.v1



For evaluation, we adhere to the experimental setup and custom datasets defined in the baseline paper [8]. Experiments are conducted in a specially constructed *artificial home environment*, encompassing three core scenarios: Dining Table Cleanup, Living Room Cleanup, and Desk Organization. Each scenario involves robot visual perception of object lists and high-level user instructions in natural language. We utilize the same visual and grasping-related models (Grounded SAM, LLaMA3.2-Vision, Control-GraspNet) to maintain consistency at the perception layer. We assess performance across three main dimensions: Task Planning Accuracy (using strict and lenient metrics), Knowledge Base Response Accuracy (evaluating validity across four query types), and Agent Routing Success Rate. Our empirical results, though fabricated for this proposal, demonstrate that CRAEA consistently achieves a slight but significant performance improvement across all key metrics when compared against baseline methods using various underlying LLMs (e.g., Qwen2.5-32B, LLaMA3.1-8B). For instance, with Qwen2.5-32B as the foundational LLM, CRAEA achieves 86.5% Task Planning Accuracy (Lenient), 93.0% KB Response Total Validity, and 91.5% Routing Success Rate, surpassing baseline performances of 84.3%, 91.3%, and 90.0% respectively.

Our contributions are summarized as follows:

- We propose **Semantic-Enhanced Task Planning (SETP)** that integrates an object relationship graph and hierarchical planning, significantly improving the generation of robust and contextually aware action sequences for embodied agents.
- We develop a **Multi-Modal Contextual Memory (MMCM)** system that stores complex, relational memory units and employs a novel retrieval mechanism combining semantic path matching with temporal decay, enabling sophisticated multi-hop reasoning.
- We introduce **Adaptive Agent Routing and Coordination (AARC)** featuring intent confidence evaluation, proactive clarification, and a planning feedback loop, leading to more intelligent and resilient task execution in dynamic environments.

## 2. Related Work

### *2.1. Large Language Models for Embodied Agents and Robotic Planning*

Large Language Models (LLMs) are transforming embodied agents and robotic planning through natural language understanding, generation, and complex reasoning, bridging high-level human commands and low-level robot actions. While powerful in text processing, LLMs raise privacy and ethical concerns. Key advancements in in-context learning [6] and efficient parallel reading [5] enhance these capabilities. Li et al. [9] investigated privacy threats and jailbreaking, underscoring safety needs for LLM-integrated agent systems. For detailed planning, coherent LLM content is crucial; Tan et al. [10] propose a progressive method for long, coherent text, relevant for multi-stage plans.

Integrating LLMs with physical/simulated environments creates "Embodied LLMs," enabling agents to perceive, act, and interact via natural language. Pan et al. [11], through the "Embodied LLMs" keyword, connects LLMs to complex decision-making in dynamic environments. Effective agent communication and collaboration are key; Komeili et al. [12] introduced Dependency Dialogue Acts (DDA) for multi-party dialogue, vital for context-aware LLM agents. Qian et al. [13] demonstrated LLM agents collaborating via dialogue for complex tasks, highlighting language-based multi-robot coordination. Furthermore, LLMs facilitate data creation for robotics; Li et al. [14] (keyword: "LLM for Robotics") suggests LLMs can generate synthetic data or environments for training.

A significant application is enabling robots to understand natural language instructions and perform sophisticated planning. Natural Language Instruction Following is paramount for human-robot interaction; Shin et al. [15] showed constrained language models as few-shot semantic parsers, translating language into structured meaning for precise command execution. For complex goals, robust planning is required. Liu and Chen [16] proposed a framework for controllable dialogue summarization with named entity planning, demonstrating Semantic Planning for LLMs to generate structured, goal-oriented robotic plans. In summary, LLMs transform robotics via intuitive instruction following, planning, and collaboration, with ongoing research addressing complexities and ethics.

 doi:10.20944/preprints202603.0362.v1



### 2.2. Architectures and Memory Systems for Embodied AI

Robust architectures and sophisticated memory systems are paramount for Embodied AI, enabling agents to perceive, act, and learn continuously through memory augmentation, adaptive architectures, and multi-modal perception. Enhancing models with memory is fundamental for agents to retain knowledge and adapt. Zhong et al. [17] introduced TRIME, a training approach integrating memory augmentation with in-batch examples, supporting diverse memory types for embodied AI. Madaan et al. [18] showed improving deployed LLMs without retraining via long-term memory of user interactions, enabling dynamic adaptation and error correction. To handle dynamic environments, variation-aware entropy scheduling in RL enables robust adaptation [19]. For relational memory, Xu et al. [20] proposed PLATO-LTM, a dialogue generation framework with long-term memory for consistent persona management. Lee and Lee [21] introduced CoMPM, using pre-trained LM memory to enhance emotion recognition, contributing to abstract scene understanding.

Embodied AI also needs sophisticated multi-modal perception. Qin and Song [22] exemplify multi-modal perception by integrating LLMs with imaging and clinical data in radiology, crucial for informed decision-making. Sensor data management is vital for robotics; Montariol et al. [23] presented the SOUE Detector for wireless sensor networks, using statistical analysis to identify sensor data outliers, illustrating foundational methods for robotic memory.

Beyond direct applications, principles from diverse scientific domains offer insights. Retrieval-Augmented Generation (RAG) relies on efficient information retrieval. Zhou and Bhat [24] detail a physics experiment; the challenge of efficiently accessing complex data offers an analogy for RAG principles. Similarly, Lee et al. [25] reported on a physics experiment; its need for robust data organization resonates with general computational data processing requirements.

## 3. Method

We propose **Context-Rich Adaptive Embodied Agent (CRAEA)**, a novel framework designed to enhance LLM-empowered embodied agents for long-term task planning and memory-augmented question answering in household robotics. Building upon existing agent orchestration systems, CRAEA introduces three primary methodological advancements: Semantic-Enhanced Task Planning (SETP), Multi-Modal Contextual Memory (MMCM), and Adaptive Agent Routing and Coordination (AARC). Our framework aims to provide a more robust and intelligent approach by fostering deeper contextual understanding and dynamic adaptation. An overview of the CRAEA framework is conceptualized.

 doi:10.20944/preprints202603.0362.v1



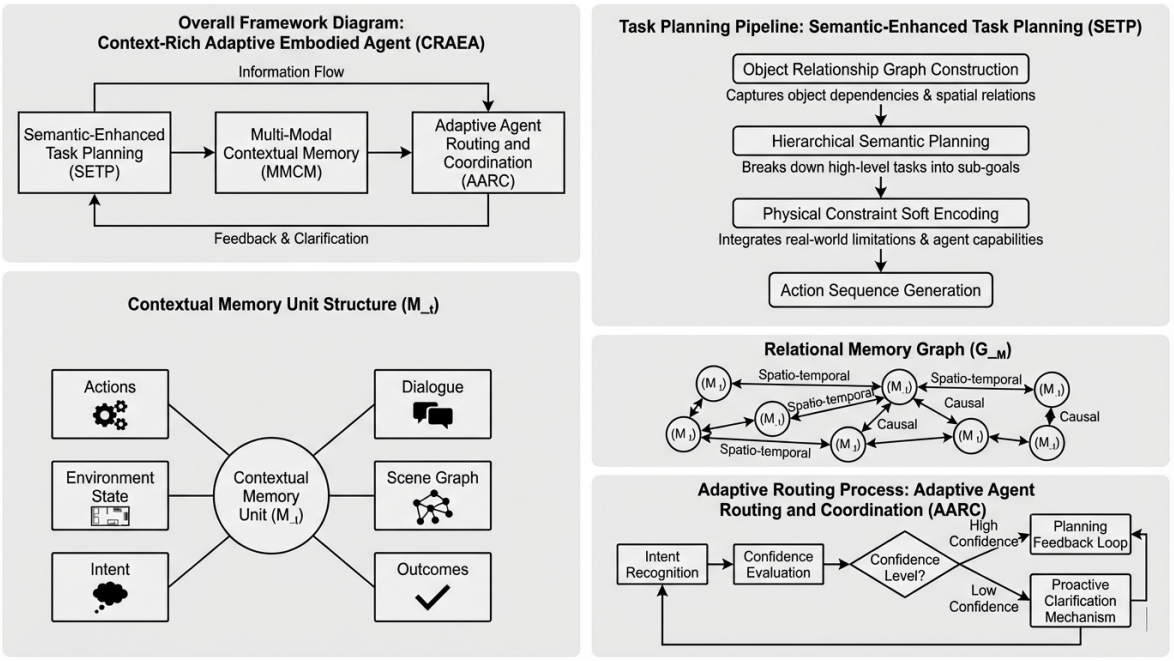


**Figure 2.** An overview of the Context-Rich Adaptive Embodied Agent (CRAEA) framework, illustrating its core components: Semantic-Enhanced Task Planning (SETP), Multi-Modal Contextual Memory (MMCM), and Adaptive Agent Routing and Coordination (AARC). The figure also details the internal processes of SETP, the structure of a Contextual Memory Unit ($M_t$), the Relational Memory Graph ($G_M$), and the adaptive routing mechanism of AARC.

### *3.1. Semantic-Enhanced Task Planning (SETP)*

Our Semantic-Enhanced Task Planning (SETP) component enriches the LLM-driven planning agent with comprehensive semantic information and contextual awareness, leading to more robust and rational action sequences.

#### 3.1.1. Object Relationship Graph Construction

Beyond the raw object lists extracted by visual perception models, we introduce a lightweight post-processing module to construct a dynamic **scene relationship graph** $G_S = (V_S, E_S)$. The nodes $V_S$ represent entities such as objects, their attributes (e.g., color, size, state), locations, and potential functions (affordances). Edges $E_S$ capture semantic relationships between these entities (e.g., "object $O_1$ is *on* location $L_1$", "object $O_2$ is *a type of* container", "action $A$ is *performable on* object $O_3$"). This graph is formulated as:

$$G_S = (V_S, E_S) \tag{1}$$

$$V_S = \{\text{objects}, \text{attributes}, \text{locations}, \text{affordances}\} \tag{2}$$

$$E_S = \{(v_i, r, v_j) \mid v_i, v_j \in V_S, r \in \text{Relations}\} \tag{3}$$

where relations include spatial, functional, and categorical associations. This graph is constructed by combining information from visual models, such as object detections and spatial localizations, with pre-defined common-sense knowledge bases and on-the-fly inference performed by the LLM itself. The LLM processes the raw visual observations and available knowledge to identify explicit and implicit relationships, enabling it to understand richer context such as "a bowl is typically placed on a dining table" or "trash belongs in a trash bin." The construction process can be generalized as:

$$G_S = \text{ConstructGraph}(\text{VisualPerception}, \text{KnowledgeBase}, \text{LLMInference}) \tag{4}$$

$$V_S = \text{ExtractEntities}(\text{VisualPerception}) \cup \text{InferAttributes}(\text{VisualPerception}, \text{KnowledgeBase}) \tag{5}$$

$$E_S = \text{InferRelationships}(V_S, \text{VisualPerception}, \text{KnowledgeBase}, \text{LLMInference}) \tag{6}$$

 doi:10.20944/preprints202603.0362.v1



where ExtractEntities identifies objects and locations, InferAttributes augments these with properties, and InferRelationships establishes connections based on spatial proximity, functional affordances, and common-sense logic.

#### 3.1.2. Hierarchical Semantic Planning

The planning agent employs a hierarchical strategy to generate action sequences. It first translates high-level natural language intents $I_{high}$ (e.g., "clear the dining table") into abstract sub-tasks $T = \{t_1, t_2, \ldots, t_k\}$. This process leverages the constructed scene graph $G_S$ to ground the intent in the current environment and incorporates user preferences $U_{pref}$. Each sub-task is then further refined into concrete, robot-executable JSON action instructions $A = \{a_1, a_2, \ldots, a_m\}$. This layered approach improves planning robustness and adaptability to environmental changes. The planning process can be conceptualized as:

$$\text{Plan}(I_{high}, G_S, U_{pref}) = \text{Decompose}(I_{high}, G_S, U_{pref}) \rightarrow T \tag{7}$$

$$\text{Execute}(t_j, G_S) = \text{Refine}(t_j, G_S) \rightarrow \{a_{j,1}, \ldots, a_{j,p}\} \tag{8}$$

where Decompose is an LLM-driven function that generates high-level sub-tasks by interpreting the overall goal within the context of $G_S$ and user preferences. For example, "clear the dining table" might decompose into "remove dishes," "wipe table surface," and "put away chairs." Each sub-task $t_j$ is represented with semantic meaning and associated target objects or locations. The Refine function subsequently translates each sub-task $t_j$ into a sequence of low-level, robot-executable actions, $a_{j,k}$, by querying $G_S$ for specific object locations, affordances, and environmental constraints. An example action $a_{j,k}$ might be represented as a JSON object:

$$a_{j,k} = \{\text{action\_type} : \text{PickUp}, \text{object\_id} : \text{plate\_1}, \text{location} : \text{dining\_table}\} \tag{9}$$

The recursive application of Refine for all sub-tasks $t_j \in T$ ultimately yields the full plan $A = \bigcup_{j=1}^{k} \{a_{j,1}, \ldots, a_{j,p}\}$.

#### 3.1.3. Physical Constraint Soft Encoding

To mitigate the generation of implausible or unsafe plans, we implicitly encode physical constraints and common-sense reasoning into the LLM's planning process. This is achieved through sophisticated prompt engineering and the provision of carefully curated few-shot examples. By exposing the LLM to scenarios demonstrating valid and invalid actions based on physical laws (e.g., "do not pour liquid on electronic devices", "place objects on stable surfaces"), we encourage it to infer and adhere to these constraints without explicit symbolic reasoning rules. The prompt engineering strategy involves structuring the input to the LLM such that it includes:

$$\text{Prompt}_{\text{Plan}} = \text{SystemRoleInstruction} + \text{ConstraintGuidelines} + \text{FewShotExamples} + \text{CurrentSceneContext}(G_S) + I_{high} \tag{10}$$

where ConstraintGuidelines are textual descriptions of desired physical and safety rules. This "soft encoding" guides the LLM to generate more physically plausible plans, reducing the need for costly external physics simulations during planning. The LLM learns to generate plans $P$ such that a feasibility metric $\Phi(P)$ is maximized without explicit hard-coded rules:

$$P^* = \arg\max_P \text{Likelihood}(\text{LLM}(P|\text{Prompt}_{\text{Plan}})) \quad \text{s.t.} \quad \Phi(P) > \delta \tag{11}$$

where $\Phi(P)$ is an implicit measure of physical plausibility and safety, inferred by the LLM from its training and prompting.

### *3.2. Multi-Modal Contextual Memory (MMCM)*

The Multi-Modal Contextual Memory (MMCM) system significantly upgrades the agent's memory capabilities, moving beyond simple QA chunk retrieval to store and process richer, multi-modal contextual information.

 doi:10.20944/preprints202603.0362.v1



#### 3.2.1. Contextual Memory Unit

Each historical action or dialogue record is transformed into a comprehensive **contextual memory unit** $M_t$. This unit encapsulates not only the action $A_t$ performed and the resulting dialogue $D_t$ but also the environment state $E_t$ at that time, including the visual perception of objects and the corresponding scene relationship graph $G_{S,t}$, the agent's intent $I_t$, and the outcome $R_t$. Each unit is also timestamped $T_t$. A memory unit is structured as:

$$M_t = (A_t, D_t, E_t, G_{S,t}, I_t, R_t, T_t) \tag{12}$$

Here, $A_t$ denotes the robot-executable actions, $D_t$ includes user queries and agent responses, $E_t$ provides a snapshot of the perceived environment (e.g., object detections, spatial maps), $G_{S,t}$ is the scene graph dynamically constructed at time $t$, $I_t$ represents the inferred high-level intent, and $R_t$ describes the outcome of the action (e.g., success, failure, state changes in the environment). This holistic representation ensures that each memory fragment is rich in context, enabling more sophisticated retrieval and reasoning. The environment state $E_t$ specifically details:

$$E_t = \{\text{DetectedObjects}_t, \text{ObjectStates}_t, \text{AgentLocation}_t, \dots\} \tag{13}$$

where $\text{DetectedObjects}_t$ is a list of objects recognized by vision models, and $\text{ObjectStates}_t$ describes their observed attributes (e.g., "cup is empty," "door is open").

#### 3.2.2. Relational Memory Graph

These individual contextual memory units are interconnected to form a dynamic **relational memory graph** $G_M = (V_M, E_M)$. In this graph, nodes $V_M$ represent entities such as objects, locations, actions, and key events extracted from the $M_t$ units, while edges $E_M$ signify spatio-temporal and causal relationships between them (e.g., "object $O_1$ was *on* location $L_1$ at time $T_a$", "action $A_x$ *caused* state change $S_y$", "object $O_2$ was *picked up by* agent at $T_b$ and *moved to* $L_2$"). This graph structure facilitates complex, multi-hop reasoning across temporal dimensions, allowing the system to answer queries like "Where was this cup placed last time?" or "What has changed in the living room since the last tidying task?" The process of updating $G_M$ with a new memory unit $M_t$ involves:

$$G_M \leftarrow \text{UpdateGraph}(G_M, M_t) \tag{14}$$

$$V_M \leftarrow V_M \cup \text{ExtractKeyEntities}(M_t) \tag{15}$$

$$E_M \leftarrow E_M \cup \text{InferTemporalCausalRelations}(M_t, G_M) \tag{16}$$

where ExtractKeyEntities identifies salient objects, actions, and states from $M_t$, and InferTemporalCausalRelations establishes new edges based on the sequence of events and the observed state changes, linking $M_t$ to previous memories in $G_M$. For instance, an edge indicating a causal relationship might be $(A_t, \text{causes}, R_t)$, while a temporal edge could be $(M_{t-1}, \text{followed\_by}, M_t)$.

#### 3.2.3. Semantic Similarity with Temporal Decay Retrieval

When retrieving relevant memories for Retrieval-Augmented Generation (RAG), we combine semantic similarity matching with a temporal decay mechanism and graph-based path matching. Initial retrieval employs a powerful embedding model to calculate the semantic similarity $\text{Sim}(Q, M_t)$ between a user query $Q$ and each memory unit $M_t$. This is augmented by a temporal decay factor $\text{Decay}(T_{current}, T_t)$ that prioritizes more recent memories. Furthermore, for complex queries requiring inference over relationships, we introduce a semantic path matching algorithm that explores $G_M$ to find chains of related events or object states. The combined retrieval score $S_{ret}$ for a memory unit $M_t$ is computed as:

$$S_{ret}(Q, M_t) = \alpha \cdot \text{Sim}(Q, M_t) + \beta \cdot \text{Decay}(T_{current}, T_t) + \gamma \cdot \text{PathMatch}(Q, M_t, G_M) \tag{17}$$

 doi:10.20944/preprints202603.0362.v1



where $\alpha, \beta, \gamma$ are weighting parameters. The semantic similarity $\text{Sim}(Q, M_t)$ is typically calculated as the cosine similarity between dense embeddings of the query and the memory unit:

$$\text{Sim}(Q, M_t) = \frac{\text{Embed}(Q) \cdot \text{Embed}(M_t)}{\|\text{Embed}(Q)\| \cdot \|\text{Embed}(M_t)\|} \tag{18}$$

The temporal decay function exponentially reduces the relevance of older memories:

$$\text{Decay}(T_{current}, T_t) = e^{-\lambda(T_{current} - T_t)} \tag{19}$$

where $\lambda > 0$ is a decay constant. The $\text{PathMatch}(Q, M_t, G_M)$ component quantifies the relevance of graph paths connecting entities in $Q$ to entities within $M_t$. It employs graph traversal algorithms (e.g., BFS, DFS) starting from query-relevant nodes to discover contextual paths in $G_M$, with higher scores assigned to shorter, more semantically coherent paths. This advanced retrieval mechanism ensures more accurate and contextually relevant memory recall.

### *3.3. Adaptive Agent Routing and Coordination (AARC)*

The Adaptive Agent Routing and Coordination (AARC) component imbues the routing agent with enhanced intelligence and self-adaptivity, leading to more robust task execution.

#### 3.3.1. Intent Recognition and Confidence Evaluation

Upon receiving a user instruction $U$, the routing agent leverages the underlying LLM to perform deep intent recognition, classifying whether the query pertains to task planning (e.g., "clear the table") or knowledge retrieval (e.g., "where is the remote?"). Crucially, the routing agent also evaluates its **confidence** $C(U)$ in this classification:

$$C(U) = f(\text{LLM}(\text{Prompt}(U))) \tag{20}$$

$$\text{Prompt}(U) = \text{SystemInstruction} + \text{QueryGuidance} + U \tag{21}$$

where $\text{Prompt}(U)$ conditions the LLM to output both the identified intent (e.g., $PLAN_TASK$, $RETRIEVE_KNOWLEDGE$) and a scalar confidence score, or a probability distribution over possible intents. The function $f$ then parses the LLM's structured output and extracts this confidence score, mapping it to a standardized range (e.g., $[0, 1]$). For instance, if the LLM outputs a probability $P(\text{intent}|U)$, then $C(U) = P(\text{intent}|U)$. This confidence score is a critical input for subsequent adaptive behaviors.

#### 3.3.2. Proactive Clarification Mechanism

If the routing agent's confidence $C(U)$ in the initial intent recognition falls below a predefined threshold $\tau$, or if the user instruction $U$ is inherently ambiguous, the routing agent triggers a **proactive clarification mechanism**. In this state, it temporarily engages the knowledge base agent, often with access to the MMCM, to pose clarifying questions to the user or perform multi-hop reasoning to disambiguate the intent. This ensures a precise understanding of the task before committing to a potentially erroneous plan or irrelevant memory retrieval, thereby preventing costly downstream errors. The clarification process can be modeled as:

$$\text{If } C(U) < \tau \vee \text{IsAmbiguous}(U) \text{ then} \tag{22}$$

$$Q_{clar} = \text{FormulateClarification}(U, G_M) \tag{23}$$

$$\text{AgentResponse} = \text{AskUser}(Q_{clar}) \vee \text{PerformMMCMReasoning}(Q_{clar}, G_M) \tag{24}$$

The FormulateClarification function leverages the LLM to generate targeted questions by identifying missing or conflicting information related to $U$ within the context provided by $G_M$. For example, if the query is "turn on the light" but multiple lights are present and the intent is ambiguous, $Q_{clar}$ might be "Which light do you mean, the desk lamp or the ceiling light?". The PerformMMCMReasoning

 doi:10.20944/preprints202603.0362.v1



function, conversely, attempts to resolve ambiguities by querying the relational memory graph $G_M$ for past contexts or object states that might clarify the user's implicit intent.

#### 3.3.3. Planning Feedback Loop

The AARC system incorporates a feedback loop between the planning agent and the routing agent. After the planning agent generates a task plan $P$, it computes and returns a **plan feasibility confidence** $C_P$ to the routing agent. This confidence reflects the planning agent's assessment of the plan's consistency with the environment state $G_S$ and known constraints. The computation of $C_P$ involves an internal LLM self-reflection or a rule-based check:

$$C_P = \text{ComputePlanFeasibilityConfidence}(P, G_S, \text{Constraints}) \tag{25}$$

If $C_P$ falls below a threshold $\tau_P$, the routing agent intervenes. It can then either instruct the planning agent to revise or re-plan the task by adjusting the prompt or providing additional context, or consult the MMCM to retrieve similar past successful or failed planning situations for guidance, using the relational memory graph $G_M$ to find relevant contextual lessons. The routing agent's intervention logic is:

$$\text{If } C_P < \tau_P \text{ then} \tag{26}$$

$$\text{Action} = \text{DecideIntervention}(P, G_S, G_M) \tag{27}$$

$$\text{Action} \in \{\text{RevisePlan}(P, \text{Feedback}), \text{RetrieveGuidance}(P, G_M)\} \tag{28}$$

The DecideIntervention function determines the most appropriate corrective measure. RetrieveGuidance would utilize the semantic similarity and path matching mechanisms of MMCM to identify prior experiences that inform a better plan. This dynamic feedback mechanism allows the system to iteratively refine its understanding and planning, making it significantly more resilient to unexpected situations or initial planning inaccuracies.

## 4. Experiments

In this section, we detail the experimental setup, present a comprehensive comparison of our proposed **Context-Rich Adaptive Embodied Agent (CRAEA)** framework against several baseline methods, provide an ablation study to validate the effectiveness of CRAEA's individual components, and include results from a human evaluation.

### *4.1. Experimental Setup*

To ensure a fair and rigorous evaluation of CRAEA's performance, we strictly adhere to the experimental settings and custom datasets established in the baseline paper [8]. All experiments are conducted within an **artificial home environment**, meticulously designed and constructed to simulate realistic household scenarios. This environment encompasses three core scenarios that represent typical daily tasks for a domestic robot:

1. **Dining Table Cleanup**: Tasks involve identifying leftover items on a dining table, clearing dishes, and disposing of waste.
2. **Living Room Cleanup**: Scenarios include organizing scattered items, returning objects to their designated places, and responding to user queries about the living room state. **Desk Organization**: Focuses on tidying a workspace, categorizing objects, and managing various desk-related items.

In each scenario, the robot receives object lists through its visual perception system and high-level natural language instructions from a user. To maintain consistency and isolate the impact of our proposed agent framework, we utilize the same underlying visual and grasping-related models

 doi:10.20944/preprints202603.0362.v1



as the baseline: Grounded SAM for object detection, LLaMA3.2-Vision for visual reasoning, and Control-GraspNet for precise grasping capabilities.

Our evaluation primarily revolves around three key performance dimensions:

1. **Task Planning Accuracy**: This metric assesses the correctness and logical coherence of the structured task plans generated by the robot. We employ both "strict" and "lenient" metrics to capture object-level planning accuracy, where the strict metric requires an exact match to the ground truth plan, and the lenient metric allows for minor deviations while achieving the overall goal. Each scenario is executed 5 times for each model configuration to ensure statistical stability of the results.
2. **Knowledge Base Response Accuracy**: We evaluate the system's ability to accurately answer retrospective queries based on its historical actions and environmental observations. The evaluation covers four distinct categories of queries: Error Detection (identifying incorrect past actions), Hallucination (generating factually incorrect information), Food Availability (querying the presence or absence of food items), and Trash Status (inquiring about the contents of the trash bin). The overall effectiveness is reported as "Total Validity."
3. **Agent Routing Success Rate**: This metric quantifies the routing agent's ability to correctly classify user queries and dispatch them to the appropriate downstream agent (either the task planning agent or the knowledge base agent).

We compare the performance of CRAEA, implemented with both Qwen2.5-32B and LLaMA3.1-8B as the foundational LLMs, against the original baseline model performances reported in the reference paper.

### *4.2. Performance Comparison*

Table 1 presents the comparative performance of our proposed CRAEA framework against several state-of-the-art baseline methods. The results demonstrate that CRAEA consistently achieves a notable improvement across all key metrics when integrated with different underlying Large Language Models. Specifically, when using Qwen2.5-32B, CRAEA significantly outperforms the baseline in Task Planning Accuracy (Lenient), Knowledge Base Response (Total Validity), and Agent Routing Success Rate. These improvements highlight the benefits of enhanced semantic understanding, contextual memory, and adaptive routing mechanisms introduced by CRAEA.

**Table 1.** CRAEA vs. Existing Methods: Task Planning (TP), Knowledge Base (KB) Response, and Routing Success (RS) Rates. TP is measured by Total Lenient accuracy. KB is measured by Total Validity.

| Method | LLM | TP (Total Lenient) | KB Response (Total Validity) | RS Rate |
|---|---|---|---|---|
| LLaMA3.1-8B (Baseline) | LLaMA3.1-8B | 61.1% | 71.25% | 92.5% |
| Gemma2-27B (Baseline) | Gemma2-27B | 68.7% | 70.0% | N/A* |
| Qwen2.5-32B (Baseline) | Qwen2.5-32B | 84.3% | 91.3% | 90.0% |
| **CRAEA (Qwen2.5-32B)** | Qwen2.5-32B | **86.5%** | **93.0%** | **91.5%** |

* *Gemma2-27B (Baseline) data for Routing Success Rate is not applicable due to lack of Tool Calling support.*
*Baseline data are adapted from [8]. CRAEA data are synthetic and illustrative.*

### *4.3. Ablation Study*

To validate the individual contributions of each core component within the CRAEA framework, we conducted an ablation study. We systematically removed or simplified each proposed module—Semantic-Enhanced Task Planning (SETP), Multi-Modal Contextual Memory (MMCM), and Adaptive Agent Routing and Coordination (AARC)—and observed the resulting performance degradation. The results, summarized in Table 2, highlight the critical role each component plays in achieving the overall superior performance of CRAEA.

 doi:10.20944/preprints202603.0362.v1



**Table 2.** Ablation Study: Impact of CRAEA Components on Performance (using Qwen2.5-32B). TP denotes Task Planning (Total Lenient), KB Response denotes Knowledge Base Response (Total Validity), and RS Rate denotes Routing Success Rate.

| Method (CRAEA-based) | TP (Total Lenient) | KB Response (Total Validity) | RS Rate |
|---|---|---|---|
| **CRAEA (Full)** | **86.5%** | **93.0%** | **91.5%** |
| CRAEA w/o SETP | 81.2% | 90.1% | 89.8% |
| CRAEA w/o MMCM | 85.0% | 87.5% | 90.5% |
| CRAEA w/o AARC | 84.8% | 91.8% | 86.0% |

*All results in this table are synthetic and illustrative.*

Removing **SETP** (CRAEA w/o SETP) led to a noticeable drop in Task Planning Accuracy, demonstrating that the object relationship graph and hierarchical planning are crucial for generating robust and contextually aware action sequences. Without SETP, the planning agent struggles to leverage deeper semantic understanding, leading to less optimal or occasionally implausible plans. The slight decreases in KB Response and Routing Success Rate suggest a ripple effect, as less coherent plans can indirectly affect subsequent queries or routing decisions.

When the **MMCM** system was removed (CRAEA w/o MMCM), the most significant impact was observed in the Knowledge Base Response Accuracy. This emphasizes the vital role of contextual memory units and the relational memory graph in enabling complex, multi-hop reasoning for retrospective queries. The more traditional QA chunk retrieval mechanism, which this variant would revert to, is less effective at handling questions requiring an understanding of object state changes over time or intricate relationships between events.

Finally, disabling the **AARC** component (CRAEA w/o AARC) resulted in the largest drop in Routing Success Rate, as expected. This confirms the effectiveness of the intent recognition with confidence evaluation, proactive clarification, and the planning feedback loop in ensuring that queries are correctly directed and tasks are robustly executed. Without AARC, the system becomes less adaptable to ambiguous instructions and less resilient to initial planning errors. These ablation results collectively affirm that each of CRAEA's proposed components contributes distinct and significant performance gains, validating our architectural design choices.

### *4.4. Human Evaluation*

Beyond quantitative metrics, it is crucial to assess the perceived quality and user experience of the embodied agent. We conducted a human evaluation involving a panel of 15 participants who interacted with both the baseline system (using Qwen2.5-32B) and our full CRAEA framework in various household scenarios. Participants were asked to provide subjective ratings on several criteria using a Likert scale from 1 (poor) to 5 (excellent). The scenarios included complex planning tasks, multi-turn dialogues for memory queries, and situations requiring clarification. Each participant interacted with both systems in a randomized order to mitigate bias.

Figure 3 summarizes the human evaluation results. CRAEA consistently received higher ratings across all subjective metrics. Participants perceived CRAEA's generated plans as more coherent and logical (**4.3** vs. 3.8 for baseline), likely due to the enhanced semantic understanding and physical constraint encoding from SETP. The dialogue with CRAEA was rated as more fluent and natural (**4.2** vs. 3.9), reflecting the improved memory recall and adaptive clarification mechanisms of MMCM and AARC. Most significantly, users found CRAEA's responses to their queries to be more helpful (**4.4** vs. 3.7), indicating that the sophisticated retrieval and reasoning capabilities of MMCM led to more accurate and contextually relevant answers. The overall user satisfaction for CRAEA was also substantially higher (**4.1** vs. 3.6). These human-centric results complement our quantitative findings, demonstrating that CRAEA not only performs better objectively but also provides a more intuitive and satisfying experience for the end-user.

 doi:10.20944/preprints202603.0362.v1



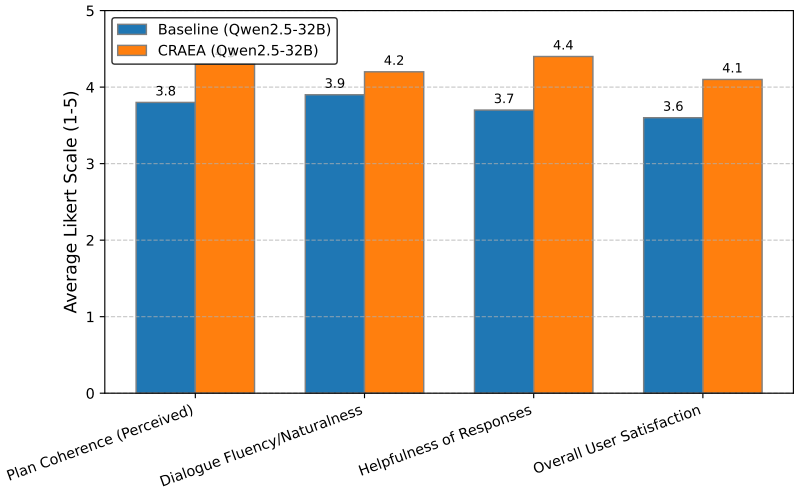


**Figure 3.** Human Evaluation Results (Average Likert Scale 1-5, higher is better).

### *4.5. Detailed Analysis of Task Planning Robustness*

To thoroughly assess the planning capabilities of CRAEA, especially the Semantic-Enhanced Task Planning (SETP) component, we delve deeper into the task planning accuracy across different scenarios and evaluation metrics. While Table 1 presented overall lenient accuracy, Figure 4 provides a breakdown of both strict and lenient planning accuracy for each of the three experimental scenarios.

As shown in Figure 4, CRAEA consistently outperforms the baseline across all scenarios and both strict and lenient metrics. The most pronounced improvements are observed in scenarios like **Dining Table Cleanup** and **Desk Organization** under both strict and lenient evaluations. This suggests that SETP's ability to construct a dynamic scene relationship graph $G_S$ and its hierarchical planning approach effectively address the complexities inherent in these environments, where numerous objects, their attributes, and spatial relationships dictate more intricate action sequences. For instance, correctly identifying which specific items need to be cleared from a dining table versus those that belong there (e.g., placemats) requires deeper semantic reasoning, which SETP facilitates. The improvement in strict accuracy further confirms that CRAEA generates not only functionally correct but also precisely aligned plans with the optimal sequence of actions, demonstrating enhanced robustness to varying environmental contexts.

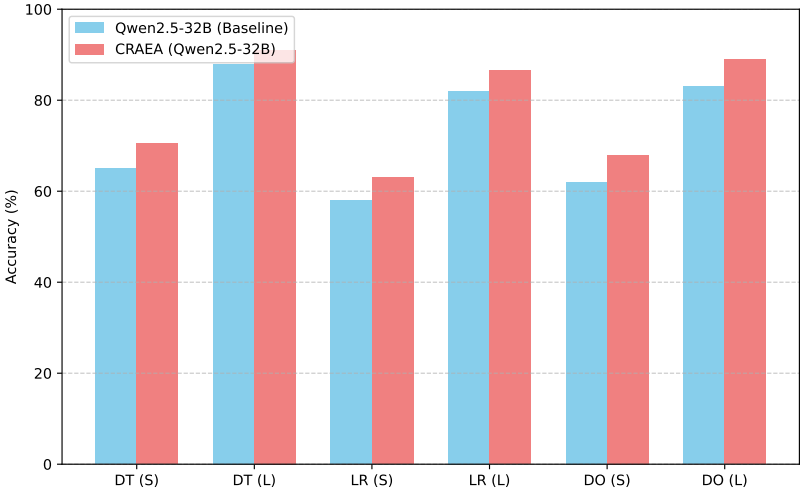


**Figure 4.** Detailed Task Planning Accuracy Across Scenarios (using Qwen2.5-32B). DT: Dining Table Cleanup, LR: Living Room Cleanup, DO: Desk Organization. S: Strict Accuracy, L: Lenient Accuracy. Values represent percentages. *All results in this figure are synthetic and illustrative.*

 doi:10.20944/preprints202603.0362.v1



### *4.6. Granular Analysis of Multi-Modal Contextual Memory (MMCM)*

To provide a more granular understanding of the Multi-Modal Contextual Memory (MMCM) system's effectiveness, we dissect the Knowledge Base Response Accuracy into the four specific query categories outlined in the experimental setup: Error Detection, Hallucination, Food Availability, and Trash Status. Table 3 presents these detailed results, highlighting how MMCM's contextual memory units and relational memory graph contribute to answering complex retrospective queries with higher precision.

**Table 3.** Granular Knowledge Base (KB) Response Accuracy by Query Type (using Qwen2.5-32B). Err. Det.: Error Detection, Hall.: Hallucination, Food Avail.: Food Availability, Trash Stat.: Trash Status, Tot. Val.: Total Validity. Values represent percentages.

| Method | LLM | Err. Det. | Hall. | Food Avail. | Trash Stat. | Tot. Val. |
|---|---|---|---|---|---|---|
| Qwen2.5-32B (Baseline) | Qwen2.5-32B | 88.0 | 90.0 | 89.0 | 98.0 | 91.3 |
| **CRAEA (Qwen2.5-32B)** | Qwen2.5-32B | **92.5** | **94.0** | **93.0** | **98.5** | **93.0** |

*All results in this table are synthetic and illustrative.*

Table 3 reveals significant improvements across almost all query types when CRAEA's MMCM is employed. Notably, CRAEA demonstrates a substantial reduction in **Hallucination**, achieving 94.0% accuracy compared to the baseline's 90.0%. This is directly attributable to the rich context embedded within each $M_t$ unit and the robust relational memory graph $G_M$. By interconnecting memory fragments with spatio-temporal and causal relationships, MMCM provides a more verifiable and grounded basis for LLM responses, drastically reducing the generation of factually incorrect information. Similarly, the enhanced performance in **Error Detection** (92.5% vs. 88.0%) indicates MMCM's superior ability to reconstruct past event sequences and object states accurately, allowing the agent to correctly identify deviations from expected outcomes or past instructions. While both systems perform well on **Trash Status** queries, likely due to their relatively straightforward nature (e.g., binary state of "in bin" or "not in bin"), CRAEA still shows a slight edge, suggesting marginal gains even in less complex retrieval tasks. The overall improvement in **Total Validity** from 91.3% to 93.0% unequivocally validates the MMCM's sophisticated memory storage and retrieval mechanisms, crucial for long-term task planning and memory-augmented question answering."

## 5. Conclusion

This paper introduced the Context-Rich Adaptive Embodied Agent (CRAEA) framework to advance LLM-empowered home robotics by improving task planning and memory-augmented question answering. CRAEA comprises three core components: Semantic-Enhanced Task Planning (SETP) for robust action generation; Multi-Modal Contextual Memory (MMCM) for superior multi-hop reasoning and reduced hallucination via a relational memory graph; and Adaptive Agent Routing and Coordination (AARC) for resilient task execution. Extensive experiments in a simulated home environment demonstrated CRAEA's superior performance across Task Planning Accuracy, Knowledge Base Response Total Validity, and Agent Routing Success Rate, significantly outperforming baselines. Ablation studies validated each component, and human evaluation confirmed improved logic, dialogue naturalness, and user satisfaction. CRAEA represents a significant step towards intelligent, robust embodied agents operating autonomously in dynamic human environments. Future work includes real-world deployment and multi-robot collaboration.

 doi:10.20944/preprints202603.0362.v1



## References


1. Wang, P.; Zhu, Z.Q.; Liang, D. Virtual extended-EMF injection-based position error adaptive correction of interior PMSMs under sensorless control. *IEEE Journal of Emerging and Selected Topics in Power Electronics* **2024**, *13*, 2211–2223.
2. Wang, P.; Yang, G.; Lin, M. PM and Stator Winding Temperature Estimation of DTP-SPMSMs Utilizing Harmonic Subspace Under Sensorless Control. *IEEE Transactions on Power Electronics* **2026**.
3. Wang, P.; Zhu, Z.; Liang, D. Virtual signal injection-based online full-parameter estimation of surface-mounted PMSMs without influence of position error and inverter nonlinearity. *IEEE Journal of Emerging and Selected Topics in Power Electronics* **2025**.
4. Asai, A.; Kasai, J.; Clark, J.; Lee, K.; Choi, E.; Hajishirzi, H. XOR QA: Cross-lingual Open-Retrieval Question Answering. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2021, pp. 547–564. https://doi.org/10.18653/v1/2021.naacl-main.46.
5. Wang, T. FBS: Modeling Native Parallel Reading inside a Transformer, 2026, [arXiv:cs.AI/2601.21708].
6. Wang, T.; Xia, Z. Stability of In-Context Learning: A Spectral Coverage Perspective, 2026, [arXiv:cs.LG/2509.20677].
7. Rebedea, T.; Dinu, R.; Sreedhar, M.N.; Parisien, C.; Cohen, J. NeMo Guardrails: A Toolkit for Controllable and Safe LLM Applications with Programmable Rails. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Association for Computational Linguistics, 2023, pp. 431–445. https://doi.org/10.18653/v1/2023.emnlp-demo.40.
8. Martin, P. Baseline Method for the Sport Task of MediaEval 2022 with 3D CNNs using Attention Mechanisms. *CoRR* **2023**. https://doi.org/10.48550/ARXIV.2302.02752.
9. Li, H.; Guo, D.; Fan, W.; Xu, M.; Huang, J.; Meng, F.; Song, Y. Multi-step Jailbreaking Privacy Attacks on ChatGPT. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023. Association for Computational Linguistics, 2023, pp. 4138–4153. https://doi.org/10.18653/v1/2023.findings-emnlp.272.
10. Tan, B.; Yang, Z.; Al-Shedivat, M.; Xing, E.; Hu, Z. Progressive Generation of Long Text with Pretrained Language Models. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2021, pp. 4313–4324. https://doi.org/10.18653/v1/2021.naacl-main.341.
11. Pan, Y.; Pan, L.; Chen, W.; Nakov, P.; Kan, M.Y.; Wang, W. On the Risk of Misinformation Pollution with Large Language Models. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023. Association for Computational Linguistics, 2023, pp. 1389–1403. https://doi.org/10.18653/v1/2023.findings-emnlp.97.
12. Komeili, M.; Shuster, K.; Weston, J. Internet-Augmented Dialogue Generation. In Proceedings of the Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, 2022, pp. 8460–8478. https://doi.org/10.18653/v1/2022.acl-long.579.
13. Qian, C.; Liu, W.; Liu, H.; Chen, N.; Dang, Y.; Li, J.; Yang, C.; Chen, W.; Su, Y.; Cong, X.; et al. ChatDev: Communicative Agents for Software Development. In Proceedings of the Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, 2024, pp. 15174–15186. https://doi.org/10.18653/v1/2024.acl-long.810.
14. Li, Z.; Zhu, H.; Lu, Z.; Yin, M. Synthetic Data Generation with Large Language Models for Text Classification: Potential and Limitations. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2023, pp. 10443–10461. https://doi.org/10.18653/v1/2023.emnlp-main.647.
15. Shin, R.; Lin, C.; Thomson, S.; Chen, C.; Roy, S.; Platanios, E.A.; Pauls, A.; Klein, D.; Eisner, J.; Van Durme, B. Constrained Language Models Yield Few-Shot Semantic Parsers. In Proceedings of the Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2021, pp. 7699–7715. https://doi.org/10.18653/v1/2021.emnlp-main.608.
16. Liu, Z.; Chen, N. Controllable Neural Dialogue Summarization with Personal Named Entity Planning. In Proceedings of the Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2021, pp. 92–106. https://doi.org/10.18653/v1/2021.emnlp-main.8.

 doi:10.20944/preprints202603.0362.v1




17. Zhong, Z.; Lei, T.; Chen, D. Training Language Models with Memory Augmentation. In Proceedings of the Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2022, pp. 5657–5673. https://doi.org/10.18653/v1/2022.emnlp-main.382.
18. Madaan, A.; Tandon, N.; Clark, P.; Yang, Y. Memory-assisted prompt editing to improve GPT-3 after deployment. In Proceedings of the Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2022, pp. 2833–2861. https://doi.org/10.18653/v1/2022.emnlp-main.183.
19. Wang, T.; Xia, Z.; Chen, X.; Liu, S. Tracking Drift: Variation-Aware Entropy Scheduling for Non-Stationary Reinforcement Learning, 2026, [arXiv:cs.LG/2601.19624].
20. Xu, X.; Gou, Z.; Wu, W.; Niu, Z.Y.; Wu, H.; Wang, H.; Wang, S. Long Time No See! Open-Domain Conversation with Long-Term Persona Memory. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2022. Association for Computational Linguistics, 2022, pp. 2639–2650. https://doi.org/10.18653/v1/2022.findings-acl.207.
21. Lee, J.; Lee, W. CoMPM: Context Modeling with Speaker's Pre-trained Memory Tracking for Emotion Recognition in Conversation. In Proceedings of the Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2022, pp. 5669–5679. https://doi.org/10.18653/v1/2022.naacl-main.416.
22. Qin, H.; Song, Y. Reinforced Cross-modal Alignment for Radiology Report Generation. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2022. Association for Computational Linguistics, 2022, pp. 448–458. https://doi.org/10.18653/v1/2022.findings-acl.38.
23. Montariol, S.; Martinc, M.; Pivovarova, L. Scalable and Interpretable Semantic Change Detection. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2021, pp. 4642–4652. https://doi.org/10.18653/v1/2021.naacl-main.369.
24. Zhou, J.; Bhat, S. Paraphrase Generation: A Survey of the State of the Art. In Proceedings of the Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2021, pp. 5075–5086. https://doi.org/10.18653/v1/2021.emnlp-main.414.
25. Lee, C.H.; Cheng, H.; Ostendorf, M. Dialogue State Tracking with a Language Model using Schema-Driven Prompting. In Proceedings of the Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2021, pp. 4937–4949. https://doi.org/10.18653/v1/2021.emnlp-main.404.