A Use Case of Natural Language Programming for Industrial Robots

Alessio Saladino
Sapienza University of Rome, Italy
saladino@diag.uniroma1.it

Luca Iocchi
Sapienza University of Rome, Italy
iocchi@diag.uniroma1.it

Vincenzo Agostiniano
R&D Home Care
Procter&Gamble, Belgium
agostiniano.v@pg.com

Pieter Saveyn
R&D Home Care
Procter&Gamble, Belgium
saveyn.p.1@pg.com

***Index Terms*—Cognitive Architecture, Natural Language Programming, Collaborative Robotics**

I. Introduction

Recent advances in Large Language Models (LLMs) have enabled their use in natural language tasks across various domains [6], including robotics. Recent studies have extensively explored the integration of LLMs with social robots, highlighting how their ability to generate human-like language can lead to more natural, coherent, and effective interactions with people. In industrial settings, combining LLMs and other AI technologies with collaborative or industrial robots can support the development of more intelligent robotic behaviors, ultimately improving task execution efficiency. Previous research has demonstrated that it is possible to integrate LLMs and vision-based LLMs to perform robotic tasks described using natural language [1] [5]. Moreover, LLMs have demonstrated the ability to convert natural language into low-level actions that enable the execution of industrial tasks [3].

LLMs have limitations, such as biased outputs, hallucinations, lack of context, and no persistent memory, making adaptation to specific environments challenging. To address this, the authors previously developed a cognitive architecture [4] using LLMs and common sense knowledge to process semi-structured data via modular components (Supervisors), utilizing prompt engineering. Fig. 1 shows a high-level view of the architecture components.

Although this architecture was originally designed for general interaction scenarios, in this work, we aim to explore its integration within an industrial context. Unlike social environments, which require the robot to be capable of handling social behaviors and intents, here the robot is required to understand and execute complex tasks. To achieve its goals, the robot must be capable of understanding the human expert's requirements and feedback expressed in natural language. It must also be capable of reason and understand how to comply with specified constraints and avoid diverging from the human-imposed goal.

Funded by the European Union - Next Generation EU, CUP B53C24003450004.

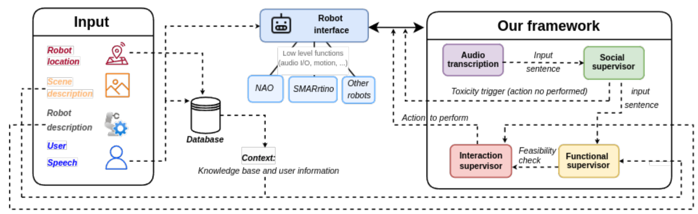

Fig. 1. Cognitive architecture for LLM-driven robots.

II. Use case

The integration of natural language processing capabilities in collaborative robots (cobots) presents a significant opportunity for the fast-moving consumer goods (FMCG) industry, particularly in the development of innovative home care dishwashing products. In this context, cobots can execute a variety of tasks aimed at predicting and enhancing the cleaning and sudsing behaviors of hand dishwashing products. By leveraging natural language commands, domain-expert users—such as product developers—can interact with the cobots more intuitively, enabling them to specify complex experimental protocols and parameters without requiring extensive programming knowledge. For instance, a cobot equipped with a LLM can be tasked with conducting experiments that evaluate the efficacy of different formulations on various types of dishware. Using natural language instructions, the cobot can autonomously adjust variables such as water temperature, detergent concentration, and scrubbing techniques, while simultaneously gathering real-time feedback on cleaning performance. This capability allows for a more nuanced understanding of how different product formulations behave under varying conditions and different consumer habits, ultimately leading to more informed product development decisions. Moreover, the data collected during these experiments can be correlated with consumer feedback to refine product offerings further. By analyzing how well different formulations perform in controlled tests versus actual consumer experiences, the cobot can help identify gaps in product performance and user satisfaction. This dual approach—merging technical testing with consumer insights—can accelerate the development cycle, allowing FMCG companies to bring innovative and effective dishwashing products to market more swiftly.

III. METHOD

This section briefly describes the customization of the above-mentioned cognitive architecture for industrial collaborative robots. In particular, we focus on scenarios in which domain-expert users without robot programming expertise are willing to define robot behaviors by using natural language commands. The architecture is thus used to finally produce executable robot controls according to the user specifications given in natural language.

To enhance accuracy and reduce hallucinations, Retrieval-Augmented Generation (RAG) can be integrated into the system to provide the robot with long-term memory. This approach allows the LLM to generate responses that are more precise and aligned with the specific application domain. In industrial contexts [2], RAG can also be used to store production parameters collected by the robot during operation.

The main strength of our cognitive architecture is scalability: it can be adapted to any context and robot without the need for code rewriting. The architecture only requires specifying the application context, its constraints, and the physical description of the robot, accompanied by the list of actions that it can perform. The architecture reasons at a high level and provides the robot behavior as output, formatted in a semi-structured way. This semi-structured data can be used to map the high-level reasoning with the low-level controller of the robot. This flexibility easily allows the use of our architecture in several industrial contexts featuring different requirements.

IV. PRELIMINARY RESULTS

As a preliminary experiment, we set up a UR5e robot to be used in a dish-washing context (Fig. 2). We have defined some elementary dish-washing actions that the robot can perform, associated with some parameters. Each action is composed of three parameters: movement type (circular, sinusoidal, square, or zig-zag), applied force and end-effector inclination. During the first interaction, the robot has no prior knowledge, and it doesn't know the correct way to perform each action. We asked the robot to perform a dish-cleaning movement, giving rough indications about the type of rubbing, force, and inclination to apply. The robot initially provided arbitrary force and inclination values while executing the correct movement. For the purpose of this experiment, we corrected the robot's behavior by using natural language, asking it to adjust the applied force and orientation for that specific movement. The robot adjusted the execution parameters based on the indicated vocal commands and then saved the correct execution of each action in its long-term memory. We repeated this experiment, teaching the robot the correct values of force and orientation associated with each movement. In subsequent interactions, we asked the robot to perform the same movements that it had previously executed, without specifying the desired force of inclination. The robot exploited its long-term memory and the reasoning performed by the supervisors of the architecture to determine the correct parameter values associated with each of the requested values, without the need for human correction.

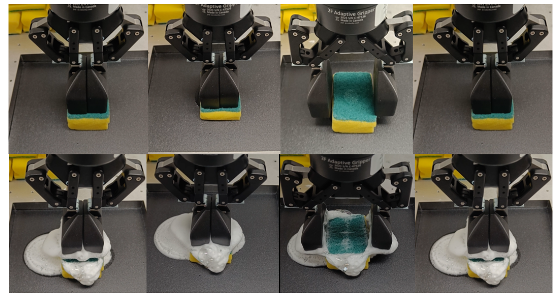

Fig. 2. Preliminary experiments made with an UR5e robot in a dish-washing context.

V. CONCLUSIONS

Our preliminary results show the robot's ability to learn and remember the correct execution of simple actions. Based on this aspect, we aim to expand this work by increasing the complexity of the scenarios. Currently, we use human feedback to adjust the robot's behavior to a desired outcome. As future improvements, we aim to expand this work by allowing the robot to adapt its behavior without the need for human feedback. The robot can use production variables acquired by sensors (i.e. water temperature or detergent concentration) to autonomously adjust its behavior to comply with constraints defined on such variables. We also aim to define an "Autonomous Exploration Mode", in which the robot is given a goal, for which it may or may not know the solution already. In this mode, the robot attempts different strategies to complete the task or learn it from scratch. Feedback is derived from sensor data collected during the task and stored in long-term memory, allowing the robot to identify and learn from failures, while keeping track of the parameters that led to those failures. If the robot already knows how to complete the task, its goal will be to explore alternative approaches to solve it. This enables a comparison of different solutions, allowing the robot to select the most efficient one.

REFERENCES

[1] Ahn, M., et al.: Do as i can, not as i say: Grounding language in robotic affordances (2022), https://arxiv.org/abs/2204.01691

[2] Choi, H., Jeong, J.: A conceptual framework for a latest information-maintaining method using retrieval-augmented generation and a large language model in smart manufacturing: Theoretical approach and performance analysis. Machines **13**(2), 94 (2025), https://www.proquest.com/scholarly-journals/conceptual-frame

[3] Fan, H., Liu, X., Fuh, J.Y.H., Lu, W.F., Li, B.: Embodied intelligence in manufacturing: leveraging large language models for autonomous industrial robotics. Journal of Intelligent Manufacturing (Jan 2024). https://doi.org/10.1007/s10845-023-02294-y, https://doi.org/10.1007/s10845-023-02294-y

[4] Saladino, A., Brienza, M., Suriani, V., Bloisi, D.D., Iocchi, L.: A cognitive architecture for embodied ai based on llm common-sense knowledge (2025)

[5] Wang, J., et al.: Large language models for robotics: Opportunities, challenges, and perspectives (2024), https://arxiv.org/abs/2401.04334

[6] Zhang, W., et al.: Sentiment analysis in the era of large language models: A reality check. findings of the association for computational linguistics: Naacl 2024 (2024)