# LLM-enabled Cyber-Physical Systems: Survey, Research Opportunities, and Challenges

Weizhe Xu
*University of Notre Dame*
South Bend, IN, USA
wxu3@nd.edu

Mengyu Liu
*University of Notre Dame*
South Bend, IN, USA
mliu9@nd.edu

Oleg Sokolsky
*University of Pennsylvania*
Philadelphia, PA, USA
sokolsky@seas.upenn.edu

Insup Lee
*University of Pennsylvania*
Philadelphia, PA, USA
lee@seas.upenn.edu

Fanxin Kong
*University of Notre Dame*
South Bend, IN, USA
fkong@nd.edu

***Abstract*—Cyber-Physical Systems (CPS) integrate computational elements with physical processes via sensors and actuators. While CPS is expected to have human-level intelligence, traditional machine learning which is trained on specific and isolated datasets seems insufficient to meet such expectation. In recent years, Large Language Models (LLMs), like GPT-4, have experienced explosive growth and show significant improvement in reasoning and language comprehension capabilities which promotes LLM-enabled CPS. In this paper, we present a comprehensive review of these studies about LLM-enabled CPS. First, we overview LLM-enabled CPS and the roles that LLM plays in CPS. Second, we categorize existing works in terms of the application domain and discuss their key contributions. Third, we present commonly-used metrics and benchmarks for LLM-enabled CPS evaluation. Finally, we discuss future research opportunities and corresponding challenges of LLM-enabled CPS.**

***Index Terms*—large language model, cyber-physical systems, machine learning**

## I. Introduction

Cyber-Physical Systems (CPS) integrate computational elements with physical processes via sensors and actuators. CPS has a wide range of applications including robots [1], self-driving vehicles [2]–[4] and so on. Researchers keep advancing CPS to be more intelligent, interactive, and working like human beings. Progress in the field of machine learning has empowered CPS with a certain level of intelligence, such as better image processing and natural language processing. However, these machine learning models are usually trained in specific and isolated datasets, which still leaves a significant gap towards human-level sensing and decision-making.

In recent years, Large Language Models (LLMs) have experienced explosive growth with the introduction of the transformer model [5] and the improvement of computing power. These LLMs, such as LLaMA [6], GPT-3 [7], and GPT-4 [8], are trained on massive web-scale datasets and possess billions of parameters. For example, the large language model GPT-3 contains 175 billion parameters while Yolo-v3 [9], a famous deep learning model used for object detection tasks, only has around 61.9 million parameters. Unlike traditional models that learn only from specific domain datasets, the learning process of these LLMs is more similar to that of humans, i.e., learning from news, books, scientific articles, code repositories, etc., which promises significant potential in human-like intelligence.

Inspired by these large language models, researchers in the CPS field have started to embedded LLMs into CPS and create LLM-enabled CPS [1], [2], [10]–[16]. Researchers leverage LLMs to enhance CPS, such as autonomous vehicles [2] and smart homes [17]. Some studies [10] have utilized the powerful natural language processing capabilities of LLMs, allowing users to interact with CPS directly through natural language. Some researchers [11], [14] attempt to take advantage of LLMs' capabilities in logic and reasoning, deploying LLMs as high-level controllers for CPS to provide reasonable planning and decision-making. Given that these LLM-enabled CPS works are spanned over various application domains that utilize different characteristics of LLMs, it is important to provide an overview of these LLM-based applications from a CPS perspective. There is a need to summarize these existing works and identify shortcomings and challenges, to provide directions and suggestions for future research.

Towards this end, we aim to summarize the applications of LLM-enabled CPS, delving into their contributions, impact, and shortcomings. To be specific, our contribution includes: (1) We overview LLM-enabled CPS, present the roles and functionalities of LLM that play in CPS. (2) We categorize existing LLM-enabled CPS works according to their application domains. (3) We have compiled commonly used metrics and benchmarks for evaluating LLM-enabled CPS. (4) We explore potential research opportunities and corresponding challenges of LLM-enabled CPS.

The remainder of the paper is organized as follows. Section II gives an overview of LLM-enabled CPS. Section III reviews existing applications of LLM-enabled CPS. Section IV presents commonly used metrics and benchmarks for evaluation. Section V shows the potential research opportunities and corresponding challenges. Section VI discusses the related survey papers referenced in this paper. Section VII concludes the survey paper.

## II. Overview

CPS powered by LLMs are anticipated to efficiently execute a variety of tasks, utilizing the human-like abilities of LLMs. When embedded into CPS, the roles of LLM in these systems can be broadly divided into two main categories: **Assistant**. LLM serves as an assistant for various characteristics such as data processing and context grounding. They do not involve specific decision-making within CPS but provide assistance to CPS. They can assist the system in interacting with the external world by handling input and output of natural language, images, and other information. In these works, LLMs bring the capability of interaction and perception to CPS. **Brain**. LLM serves as the brain of CPS to decide the motion of the controllable agent. LLMs analyze and organize information and make reasonable decisions based on knowledge from pre-trained data. In these studies, CPS leverage the advantages of LLMs in planning and reasoning. These relationships are illustrated in Figure 1. The central part of the figure represents LLM-enabled CPS. The left part illustrates the role and function of LLMs with the systems, while the right part depicts the application areas of these systems.

Thus, the functionality of LLMs for CPS can be encapsulated in the following key aspects:

**Perception.** Perception here means the capability of sensing the environment through inputs. The capabilities of LLMs in natural language processing enable them to perceive their surrounding environment through descriptions provided by users in natural language. In addition to NLP, some LLMs also possess powerful image and

979-8-3503-6345-6/24/$31.00 ©2024 IEEE
DOI 10.1109/FMSys62467.2024.00013

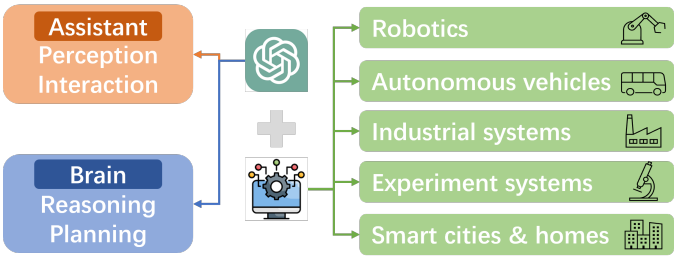


Fig. 1. Overview of LLM-enabled CPS.

video analysis capabilities, which can assist CPS in object recognition, target detection, and scene understanding. Since CPS's input spans from natural language and images to continuous data like the velocity of the car and so on, this multimodal perception capability enables CPS to better perceive their surroundings, thereby completing users' objectives more accurately. For instance, RT-2 [18] builds a multimodal LLM that directly takes images and user instructions as input, generating plans for tasks.

**Interaction.** LLMs endow CPS with improved interactive capabilities [19]. Conventional CPS's interaction functions mostly remain at the level where the user gives commands, and the CPS returns relevant execution data. This mode of interaction places a high demand on the user's expertise, as they need to analyze the data returned by the CPS themselves to determine if it meets the objective. At the same time, how to predict in advance whether the input commands will achieve the desired effect is also a problem. In contrast, LLM-enabled CPS perform much better in interaction. LLMs excel in speech recognition, semantic understanding, and natural language generation. CPS can utilize LLMs to comprehend commands issued by users in natural language, execute the requested tasks, and then provide feedback in both numerical data and natural language to the users. Meanwhile, LLM-enabled CPS can engage in multi-turn dialogues with users to help them understand the system's comprehension of the instructions and potential execution scenarios, providing timely feedback to enhance the likelihood of task completion. This mode of interaction aligns more closely with human behavior and requirements, thereby enhancing the user-friendliness and acceptance of CPS.

**Reasoning.** By utilizing existing knowledge to summarize and infer about the current issue, the reasoning capability plays a crucial role in problem-solving and decision-making. In conventional CPS, the reasoning ability is usually undertaken by algorithms deployed by the designers before. The reasoning capability of such systems is limited to the deployed algorithms, requiring designers to consider all possible scenarios, which makes it difficult to handle complex and variable situations. Although machine learning has been applied to CPS systems in recent years, the reasoning capabilities of these traditional machine learning models are also limited to the specific training dataset. In contrast, LLMs exhibit powerful reasoning capability by making inferences based on general knowledge about the world. Benefiting from the web-scale training datasets, this enables them to provide explanations or make decisions that require an understanding of expert concepts and activities. In addition, LLMs are capable of drawing analogies between different concepts, which is useful for tasks that are out of LLM's pre-trained knowledge. LLMs apply reasoning to do enhanced decision-making in these tasks after analyzing the background information provided by users.

**Planning.** Based on reasoning, planning capability refers to breaking down complex problems into smaller sub-problems and providing a step-by-step plan to gradually solve the entire issue. The performance of traditional CPS systems in planning capabilities is similar to that in reasoning. Both have significant limitations. In contrast, LLMs like the GPT series have demonstrated a noteworthy ability in planning across various contexts, benefiting from their powerful reasoning abilities. In addition, the plans generated by LLM-enabled CPS span several levels, from high-level instructions such as 'go to the market', to low-level actions like 'turn left'. This goal, which once required the deployment of multiple specific planners to achieve, can now all be solved with LLMs by giving them the appropriate information. The integration of LLMs with planning capabilities into CPS represents a significant advancement in making these systems more intelligent, autonomous, and efficient [2], [19].

Note: Unlike other domains, nearly every LLM-enabled CPS leverages reasoning and planning capabilities. Systems like question-answering that only use reasoning and not planning are not seen as CPS, since they do not interact with the physical world.

TABLE I
APPLICATIONS OF LLM-ENABLED CPS

| Domain | Works |
|---|---|
| **Robotics** | **Jasen [10], Ahn et al [11], GD [12],RT-2 [18], PaLM-E [13], Text2Motion [14], SayPlan [1]** |
| **Autonomous vehicles** | **Dilu [16], DriveGPT4 [15], Talk2BEV [20], Cui et al. [21], Cui et al. [19], Talk2Drive [2]** |
| **Industrial systems** | **Xia et al [22]** |
| **Experiment systems** | **Inagaki et al [23], CLAIRIFY [24], Boiko et al [25], Coscientist [26]** |
| **Smart cities & homes** | **GPT-in-the-Loop [27], PromptGAT [28], LLM-Mind [17]** |

## III. APPLICATION

In this section, we offer a concise overview of existing works and we organize them based on their applications across five distinct areas: robotics, autonomous vehicles, industrial systems, experiment systems, smart cities and homes. To provide a more intuitive perspective, we list these representative works in Table I.

**Robotics.** LLM-enabled robotics demonstrate powerful versatility. They can break down natural language commands into executable actions or sequences of skills through a combination of perception, interaction, reasoning, and planning to control the robot. In contrast, traditional methods that assist in controlling robots are only suitable for specific tasks according to the pre-designed algorithms or pre-trained data, struggling to understand natural language instructions. As a result, they are difficult to accurately achieve objectives for normal users without expert knowledge.

LLM-enabled CPS in the robotics domain evolves with the amount of information input to LLM. Jansen [10] shows the ability of LLM to produce high-level instructions solely from natural language instructions. In addition, it also demonstrates that providing even a small amount of visual information, such as the robot's location at the start of a task, can significantly improve the success rate of LLM-enabled planners. This inspires subsequent researchers to integrate visual information into LLM-enabled CPS. Some researchers employ additional models to perceive environmental information, generate usable instructions, and input them to LLMs in the form of natural language. In these cases, LLMs don't perceive the environment directly by themselves. In the work [11], robots perceive the environment and then apply reinforcement learning (RL) to give actionable

instructions for LLMs to select. Then they demonstrate that LLMs are capable of producing precise high-level instructions using verbal instructions from the user and other perception models. Subsequent researches show that LLMs with multimodality can directly perceive environmental information like images. RT-2 [18] builds a multimodal LLM that takes images and user's instructions as input, generating plans in an end-to-end manner. PaLM-E [13] introduces embodied language models to directly integrate real-world continuous sensor data into LLM, and thereby perceive the environment even further. This method interleaves visual, continuous state estimation, and textual input to formulate plans for robotics.

As for complex skill sequences, LLM-enabled CPS also show significant advancement. Different from the previous problem, "skills" refer to instructions with more information and constraints, such as environmental conditions and execution sequence, which are important in long-horizon planning problems. For example, Text2Motion [14] deals with sequence manipulation tasks that require long-horizon reasoning. Unlike previous methods that only consider the feasibility of individual instruction, Text2Motion considers the geometric dependency between sequences of skills during the reasoning process. Moreover, it demonstrates improved results in various types of complex tasks, such as long-horizon, multiple object instances, and tasks where skills' dependency cannot be obtained from the initial state. Current LLMs still fall short in dealing with large-scale environments and long-horizon problems, for example, they cannot adequately consider the sequence dependency of skills in long sequences. SayPlan [1] tackles these shortcomings by incorporating a classical path planner, such as Dijkstra, to shorten the LLM's planning horizon. This integration allows a mobile manipulator robot to successfully execute these large-scale, long-horizon tasks that are derived from abstract and natural language-based instructions.

**Autonomous vehicles.** LLMs hold great potential for perception, interaction, planning, and control in autonomous vehicles. Dilu [16] introduces the idea of incorporating LLMs as decision-makers in autonomous vehicles to create sequences of actions.

Enhanced by LLMs, multimodal Large Language Models (MLLMs) have attracted considerable attention for their ability to analyze non-textual data such as images and point clouds alongside text, a skill particularly valuable in the field of autonomous driving. For instance, DriveGPT4 [15] processes video inputs to produce textual responses related to driving, aiding in the analysis of vehicle actions. Talk2BEV [20] utilizes pre-trained image-language models to integrate Bird's Eye View (BEV) maps with linguistic context. This integration facilitates visuo-linguistic reasoning in autonomous vehicles, enhancing their interpretation and navigation.

As a mode of transportation for humans, autonomous vehicles have higher requirements for safety and explainability. [21] and [19] introduce frameworks where LLMs leverage their perception and reasoning capabilities to provide descriptions of how they perceive and react to environmental factors, such as weather and traffic conditions. These researches also demonstrate the capacity of adapting driving behaviors in response to human commands. Beyond simulator-based self-driving experiments, Cui et al. [2] take into account safety, efficiency, and comfort to develop Talk2Drive. This marks the first instance of a LLM-enabled autonomous driving system being applied in a real-world experiment.

**Industrial systems.** In the domain of industrial engineering, LLMs utilized in CPS are used for intelligent planning and control of production processes. Unlike previous fields, 'brains' in industrial engineering require more specialized knowledge, such as how to use these complex production equipment. The interaction and reasoning capabilities of LLMs can effectively overcome this challenge. By inputting relevant materials into the LLM, it can easily learn how to use the equipment. Reference [22] introduces an innovative approach that combines LLMs with digital twin technology to meet the dynamic needs of production. They retrofit the engineering system for a modular production facility and create control inference at different levels. Informed by digital twin data, LLMs are developed to have the capability of adjusting to particular complex tasks. LLMs in the system can manage and execute a range of basic functions and skills, facilitating production tasks across different levels of the automation hierarchy. This study showcases the promising potential of incorporating LLMs into industrial automation frameworks, offering novel strategies for achieving more intelligent, adaptable, and efficient production workflows.

**Experiment systems.** In the fields of biology and chemistry, LLMs can serve as experiment assistants in the laboratory to help design and conduct experiments. Given some instructions, LLMs can design experiments and issue commands to experimental equipment for automatic execution. In this field, the reasoning ability of LLMs is particularly important. Because LLMs need to consider the context of scientific research to propose suitable experimental plans to achieve corresponding research objectives, such as validating a particular inference. For instance, researchers in [23] combine LLMs with OT-2, an automated liquid-handling robot used in biological laboratories. Based on the context of biological experiments, LLM writes and executes operation scripts for the OT-2, easing the workload of biological researchers. As for chemical experiments, CLAIRIFY [24] combines high-level plans generated by LLMs with low-level plans generated by traditional algorithms. LLM first generates a long-term plan from natural language instructions. Then the plan is executed by solving a constrained task and motion planning problem using PDDLStream solvers [29]. Real robots complete two basic chemical experiments, solubility and recrystallization, showcasing notable outcomes. Research [25] [26] goes even further. The LLM-enabled systems collect enough information and propose an experimental plan by blending the context of the experiment with the outcomes of internet searches. Following this, the LLM consults relevant documentation on experimental equipment to generate Python code for executing. Researchers only need to provide the experimental objective as input throughout the entire process.

**Smart cities and homes.** In the fields of smart cities and homes, systems incorporate numerous sensors and actuators. Embedding LLMs into these systems also has broad prospects, capable of bringing numerous advantages including energy saving and efficiency improvement. For instance, GPT-in-the-Loop [27] is proposed for multi-agent systems. They leverage the advanced reasoning capability of GPT models within the loop of decision-making to create a self-adaptive IoT multi-agent system. This method has been applied to smart streetlights [30] benchmark for optimizing energy while ensuring adequate lighting. The LLM-enabled system in work [28] is proposed for the traffic signal control tasks. The pre-trained LLM's inference ability is exploited and applied to understand how weather conditions, traffic states, and road types influence traffic dynamics, then takes the action produced by the control policy to provide efficient transportation and mitigate congestion waste. Within LLMind [17], LLM designs control scripts through interaction with users and machines to multiple domain-specific AI modules and IoT devices in smart homes.

## IV. Evaluation

As LLM-enabled CPS continue to evolve, evaluating the effectiveness of these technologies is also a crucial issue. We primarily focus on analyzing evaluation techniques in the fields of robotics and autonomous driving from two aspects: metrics and benchmarks. Other application fields currently lack a unified benchmark and mainly rely on custom methods defined by researchers.

**Metrics.** To effectively evaluate these systems, metrics are very important, as they can influence the accuracy and persuasiveness of the evaluation results. In most studies [1], [2], [10]–[16], accuracy or plan success rate are used to measure the precision of plans generated by LLM compared with ground truth. The execution success rate is used to assess the specific execution of the plan by robots or cars. Additionally, full sequence accuracy and subgoal completion rates are utilized for measuring the accuracy and success rate of sub-tasks in some long-horizon tasks [10], [14]. RMSE and some other metrics are used to measure the control performance of LLM-enabled CPS in autonomous driving. Beyond these metrics related to planning and execution, accuracy is also used to access how LLM-enabled CPS understand multimodal data.

**Benchmarks.** As for robotics and embodied systems, Alfred [31] and Behavior [32] are two of the most popularly used benchmarks for interpreting grounded instructions. In the field of autonomous driving, datasets BDD-X [33] and DRAMA [34] which include multi-modal data such as images, control signals, and vehicle states, have been widely applied. Some other datasets, such as Nuprompt [35] and MAPLM [36] are also been considered since they contain point cloud data. Some studies have constructed their own datasets from simulators for specific scenarios [16] [20]. Beyond simulators and datasets, existing works in the field of robotics and embodied systems extensively use real mobile manipulators for experiments in real-world scenarios, such as robotic arms [11]–[13] and robotic dogs [37]. Among them, SayCan [11] constructs a dataset for mobile manipulators based on Alfred [31] and Behavior [32]. It has been widely used by robotics researchers. In the field of autonomous driving, only [2] has conducted experiments with real vehicles. Figure 2 gives an illustration of some simulators and real-world testbeds. This figure sequentially showcases simulated scenarios of robots [31], real-world scenarios of robots [11], simulated scenarios of autonomous vehicles [16], and real-world scenarios of autonomous vehicles [2].

Fig. 2. Illustration of some test scenarios

## V. Research Opportunities and Challenges

In this section, we explore the potential research opportunities for LLM-enabled CPS and give the corresponding challenges.

**Security and safety.** With the rapid development of LLM-enabled CPS, security should be considered as a important research direction. Malicious attackers can modify the instructions or data uploaded to the LLMs, causing deviations in the LLMs' output. Such deviations can lead to serious safety problems in CPS due to their interaction with the physical world [38]. For example, an autonomous vehicle may cause an accident due to deviations in the LLM's plan. In addition, LLMs can harbor biases even without being attacked. Hallucination [39] in LLMs is a widely studied phenomenon where LLMs generate information that is incorrect confidently. It occurs due to issues in the training process, such as insufficient training data or biases within the training dataset itself. When embedding LLMs into CPS, hallucination can lead these systems to confidently execute incorrect plans, thereby raising significant safety concerns. Moreover, LLMs inherently lack the capability to understand the physical world. This could lead to plans generated by LLMs violating the constraints of the physical environment in which the CPS operate, such as a robotic arm colliding with obstacles. LLMs may also be hard to understand some temporal constraints in CPS, such as deadline and events order. In conclusion, it is both necessary and urgent to design additional methods to make the LLM-enabled CPS more secure and safe because of the strong interaction between CPS and the physical world.

However, ensuring the security and safety of LLM-enabled CPS is nontrivial and can face several challenges. For example, when hallucination occurs, it is difficult to judge based on the corresponding probabilities to the output of LLM, because LLMs are confident in these incorrectly generated answers. Although multiple methods are used to feed LLMs with environmental data, LLM itself performs poorly in abstracting knowledge from continuous data for decision-making, which is widely used in CPS, like speed and position. For the safety and temporal constraints, some researches [37], [40] make progress on generating constraints-guaranteed plans. They iteratively query LLMs and validate the plans using external validation tools and providing LLMs with counter-examples. However, the ability of LLMs to consider constraints in the planning process has not seen significant improvement. These systems frequently fail to produce the correct plan after reaching the iteration limit.

**Runtime Checking/Verification of LLM.** In addition to applying methods to enhance system security and safety, it is necessary to evaluate and guarantee the security and safety of LLM-enabled CPS both prior to and throughout deployment. When applying neural networks to safety-critical applications, researchers conduct runtime checking and verification to ensure the safety of systems. For LLM-enabled CPS, the demand is even more pronounced due to the broad application areas of LLM-enabled CPS.

Real-time monitoring of LLMs and verification of LLMs present new challenges. Due to the massive number of parameters and complex network structures of LLMs compared to traditional neural networks, conventional runtime checking algorithms would lead to significant time overheads, making real-time monitoring impossible. Traditional methods for DNN verification, such as methods Reluplex [41] and reluval [42], are also unfeasible due to immeasurable computational costs. Furthermore, applying traditional runtime checking and verification to multimodal LLMs is also challenging. For example, traditional methods of verifying neural networks typically involve calculating the range of the neural network's output results after giving a certain range of inputs. However, for multimodal LLMs, the inputs may include both images and text, which have vastly different scales of input ranges. At the same time, the outputs of LLMs are often not categorical or numerical like those of traditional neural networks, but textual in natural language, which poses additional challenges for verification.

**Autonomous perception and response.** Most present researches are dedicated to creating LLM-enabled CPS capable of interacting with humans through natural language. In some cases, we aim for these systems to have the ability to perceive and autonomously respond in real-time to meet our immediate needs or maintain some

abstract objectives [27], [28]. We aspire for these systems to operate autonomously, without the need for human instructions.

Reducing humans in the loop is an important challenge in achieving this objective. This requires that LLMs not only perform reasoning and planning like humans when given specific instructions but also possess common sense similar to humans. For instance, when a teacup falls from the table and breaks, the robot is expected to automatically detect and clean up the fragments.

**LLM deployment.** Several significant problems in efficiency and accuracy emerge when deploying LLMs in CPS. Due to existing technical constraints like computational and storage resource limitations, deploying LLMs locally on CPS is infeasible. The common approach is to utilize a cloud-based LLM for complex functions such as reasoning and planning, while the local machine is responsible for transmitting data and performing pre-processing.

However, this deployment architecture still has several challenges to overcome. First, the cloud-based LLMs bring up latency issues. In some real-time systems with high requirements for response time, tasks or issues may not be processed by LLM in a timely manner due to transmission delays. Second, LLMs like GPT-4 have limitations on the size of prompt. In multi-turn dialogues, users cannot include all the content of previous prompts in a new prompt, which may lead to challenge in maintaining dialogue consistency. Existing methods summarize the content of the prompt, but this inevitably results in the loss of information, leading to inaccurate answers from LLMs.

## VI. Related Surveys

The rapid advancement of LLMs-enabled systems has given rise to numerous comprehensive surveys. [43] initially examines research in the field of LLM-based agents, focusing on their construction. The authors propose a unified framework that encapsulates much of previous work. Subsequently, they provide an overview of the wide range of applications for LLM-based autonomous agents across social science, natural science, and engineering. Finally, they discuss the prevalent evaluation strategies for LLM-based autonomous agents. They also present several challenges and future directions. In addition, [44] gives more detail on the capabilities of LLMs. This survey offers detailed insights into general aspects of the field, like natural sciences, universal autonomous agent, social sciences, and engineering systems. For each major application direction, they have made more detailed subdivisions and analyzed the current status and prospects of LLMs in these subdivided fields. For instance, in the context of engineering systems, they further categorize into industrial control systems, medical systems, military systems, and so on.

As for specific areas, [45] provides an overview of the integration of LLM intro robotic systems. This survey focuses on analyzing the capabilities required by robotic systems and offered by LLMs. It also discusses the challenges and promising directions of LLM-enabled robotic systems. [46] sheds light on evaluating LLM-enabled robotics systems. [47] conducts a literature review on autonomous driving integrated with multimodal LLMs.

## VII. Conclusion

As LLMs continue to evolve, LLM-enabled CPS will become more intelligent and efficient. However, we must also pay attention to the new security and safety issues that arise from embedding LLMs into CPS. In this survey paper, we first give an overview of LLMs' functions and roles in LLM-enabled CPS. Then we systematically summarize existing applications of LLM-enabled CPS across various fields. Subsequently, this paper provides some commonly used metrics and benchmarks for evaluating LLM-enabled CPS. In addition to reviewing the previous works, we also give a vision of potential future research opportunities and the corresponding challenges. We hope this survey paper can provide some inspiration to researchers and promote the development of the field.

## Acknowledgment

This work was supported in part by NSF CNS-2333980. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the National Science Foundation (NSF).

## References

[1] K. Rana, J. Haviland, S. Garg, J. Abou-Chakra, I. Reid, and N. Suenderhauf, "Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning," in *Conference on Robot Learning*. PMLR, 2023, pp. 23–72.

[2] C. Cui, Z. Yang, Y. Zhou, Y. Ma, J. Lu, and Z. Wang, "Large language models for autonomous driving: Real-world experiments," *arXiv preprint arXiv:2312.09397*, 2023.

[3] M. Liu, L. Zhang, P. Lu, K. Sridhar, F. Kong, O. Sokolsky, and I. Lee, "Fail-safe: Securing cyber-physical systems against hidden sensor attacks," in *2022 IEEE Real-Time Systems Symposium (RTSS)*. IEEE, 2022, pp. 240–252.

[4] M. Liu, L. Zhang, V. V. Phoha, and F. Kong, "Learn-to-respond: Sequence-predictive recovery from sensor attacks in cyber-physical systems," in *2023 IEEE Real-Time Systems Symposium (RTSS)*. IEEE, 2023, pp. 78–91.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[6] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[7] R. Dale, "Gpt-3: What's it good for?" *Natural Language Engineering*, vol. 27, no. 1, pp. 113–118, 2021.

[8] K. Sanderson, "Gpt-4 is here: what scientists think," *Nature*, vol. 615, no. 7954, p. 773, 2023.

[9] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[10] P. A. Jansen, "Visually-grounded planning without vision: Language models infer detailed plans from high-level instructions," *arXiv preprint arXiv:2009.14259*, 2020.

[11] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," *arXiv preprint arXiv:2204.01691*, 2022.

[12] W. Huang, F. Xia, D. Shah, D. Driess, A. Zeng, Y. Lu, P. Florence, I. Mordatch, S. Levine, K. Hausman *et al.*, "Grounded decoding: Guiding text generation with grounded models for robot control," *arXiv preprint arXiv:2303.00855*, 2023.

[13] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu *et al.*, "Palm-e: An embodied multimodal language model," *arXiv preprint arXiv:2303.03378*, 2023.

[14] K. Lin, C. Agia, T. Migimatsu, M. Pavone, and J. Bohg, "Text2motion: From natural language instructions to feasible plans," *arXiv preprint arXiv:2303.12153*, 2023.

[15] Z. Xu, Y. Zhang, E. Xie, Z. Zhao, Y. Guo, K. K. Wong, Z. Li, and H. Zhao, "Drivegpt4: Interpretable end-to-end autonomous driving via large language model," *arXiv preprint arXiv:2310.01412*, 2023.

[16] L. Wen, D. Fu, X. Li, X. Cai, T. Ma, P. Cai, M. Dou, B. Shi, L. He, and Y. Qiao, "Dilu: A knowledge-driven approach to autonomous driving with large language models," *arXiv preprint arXiv:2309.16292*, 2023.

[17] H. Cui, Y. Du, Q. Yang, Y. Shao, and S. C. Liew, "Llmind: Orchestrating ai and iot with llms for complex task execution," *arXiv preprint arXiv:2312.09007*, 2023.

[18] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15818*, 2023.

[19] D. Fu, X. Li, L. Wen, M. Dou, P. Cai, B. Shi, and Y. Qiao, "Drive like a human: Rethinking autonomous driving with large language models," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 910–919.

[20] V. Dewangan, T. Choudhary, S. Chandhok, S. Priyadarshan, A. Jain, A. K. Singh, S. Srivastava, K. M. Jatavallabhula, and K. M. Krishna, "Talk2bev: Language-enhanced bird's-eye view maps for autonomous driving," *arXiv preprint arXiv:2310.02251*, 2023.

[21] C. Cui, Y. Ma, X. Cao, W. Ye, and Z. Wang, "Drive as you speak: Enabling human-like interaction with large language models in autonomous vehicles," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 902–909.

[22] Y. Xia, M. Shenoy, N. Jazdi, and M. Weyrich, "Towards autonomous system: flexible modular production system enhanced with large language model agents," *arXiv preprint arXiv:2304.14721*, 2023.

[23] T. Inagaki, A. Kato, K. Takahashi, H. Ozaki, and G. N. Kanda, "Llms can generate robotic scripts from goal-oriented instructions in biological laboratory automation," *arXiv preprint arXiv:2304.10267*, 2023.

[24] N. Yoshikawa, M. Skreta, K. Darvish, S. Arellano-Rubach, Z. Ji, L. Bjørn Kristensen, A. Z. Li, Y. Zhao, H. Xu, A. Kuramshin *et al.*, "Large language models for chemistry robotics," *Autonomous Robots*, vol. 47, no. 8, pp. 1057–1086, 2023.

[25] D. A. Boiko, R. MacKnight, and G. Gomes, "Emergent autonomous scientific research capabilities of large language models," *arXiv preprint arXiv:2304.05332*, 2023.

[26] D. A. Boiko, R. MacKnight, B. Kline, and G. Gomes, "Autonomous chemical research with large language models," *Nature*, vol. 624, no. 7992, pp. 570–578, 2023.

[27] N. Nascimento, P. Alencar, and D. Cowan, "Gpt-in-the-loop: Adaptive decision-making for multiagent systems," *arXiv preprint arXiv:2308.10435*, 2023.

[28] L. Da, M. Gao, H. Mei, and H. Wei, "Llm powered sim-to-real transfer for traffic signal control," *arXiv preprint arXiv:2308.14284*, 2023.

[29] C. R. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L. P. Kaelbling, and T. Lozano-Pérez, "Integrated task and motion planning," *Annual review of control, robotics, and autonomous systems*, vol. 4, pp. 265–293, 2021.

[30] N. Nascimento, P. Alencar, C. Lucena, and D. Cowan, "Toward human-in-the-loop collaboration between software engineers and machine learning algorithms," in *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 2018, pp. 3534–3540.

[31] M. Shridhar, J. Thomason, D. Gordon, Y. Bisk, W. Han, R. Mottaghi, L. Zettlemoyer, and D. Fox, "Alfred: A benchmark for interpreting grounded instructions for everyday tasks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 740–10 749.

[32] S. Srivastava, C. Li, M. Lingelbach, R. Martín-Martín, F. Xia, K. E. Vainio, Z. Lian, C. Gokmen, S. Buch, K. Liu *et al.*, "Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments," in *Conference on Robot Learning*. PMLR, 2022, pp. 477–490.

[33] J. Kim, A. Rohrbach, T. Darrell, J. Canny, and Z. Akata, "Textual explanations for self-driving vehicles," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 563–578.

[34] S. Malla, C. Choi, I. Dwivedi, J. H. Choi, and J. Li, "Drama: Joint risk localization and captioning in driving," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 1043–1052.

[35] D. Wu, W. Han, T. Wang, Y. Liu, X. Zhang, and J. Shen, "Language prompt for autonomous driving," *arXiv preprint arXiv:2309.04379*, 2023.

[36] X. Cao, T. Zhou, Y. Ma, W. Ye, C. Cui, K. Tang, Z. Cao, K. Liang, Z. Wang, J. Rehg, and C. Zheng, "Maplm: A real-world large-scale vision-language dataset for map and traffic scene understanding," https://github.com/LLVM-AD/MAPLM, 2023.

[37] Z. Yang, S. S. Raman, A. Shah, and S. Tellex, "Plug in the safety chip: Enforcing constraints for llm-driven robot agents," *arXiv preprint arXiv:2309.09919*, 2023.

[38] L. Zhang, K. Sridhar, M. Liu, P. Lu, X. Chen, F. Kong, O. Sokolsky, and I. Lee, "Real-time data-predictive attack-recovery for complex cyber-physical systems," in *2023 IEEE 29th Real-Time and Embedded Technology and Applications Symposium (RTAS)*. IEEE, 2023, pp. 209–222.

[39] J. Li, X. Cheng, W. X. Zhao, J.-Y. Nie, and J.-R. Wen, "Halueval: A large-scale hallucination evaluation benchmark for large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023, pp. 6449–6464.

[40] S. K. Jha, S. Jha, P. Lincoln, N. D. Bastian, A. Velasquez, R. Ewetz, and S. Neema, "Neuro symbolic reasoning for planning: Counterexample guided inductive synthesis using large language models and satisfiability solving," *arXiv preprint arXiv:2309.16436*, 2023.

[41] G. Katz, C. Barrett, D. L. Dill, K. Julian, and M. J. Kochenderfer, "Reluplex: An efficient smt solver for verifying deep neural networks," in *Computer Aided Verification: 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part I 30*. Springer, 2017, pp. 97–117.

[42] S. Wang, K. Pei, J. Whitehouse, J. Yang, and S. Jana, "Formal security analysis of neural networks using symbolic intervals," in *27th USENIX Security Symposium (USENIX Security 18)*, 2018, pp. 1599–1614.

[43] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin *et al.*, "A survey on large language model based autonomous agents," *arXiv preprint arXiv:2308.11432*, 2023.

[44] Y. Cheng, C. Zhang, Z. Zhang, X. Meng, S. Hong, W. Li, Z. Wang, Z. Wang, F. Yin, J. Zhao *et al.*, "Exploring large language model based intelligent agents: Definitions, methods, and prospects," *arXiv preprint arXiv:2401.03428*, 2024.

[45] F. Zeng, W. Gan, Y. Wang, N. Liu, and P. S. Yu, "Large language models for robotics: A survey," *arXiv preprint arXiv:2311.07226*, 2023.

[46] J. Wang, Z. Wu, Y. Li, H. Jiang, P. Shu, E. Shi, H. Hu, C. Ma, Y. Liu, X. Wang *et al.*, "Large language models for robotics: Opportunities, challenges, and perspectives," *arXiv preprint arXiv:2401.04334*, 2024.

[47] C. Cui, Y. Ma, X. Cao, W. Ye, Y. Zhou, K. Liang, J. Chen, J. Lu, Z. Yang, K.-D. Liao *et al.*, "A survey on multimodal large language models for autonomous driving," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 958–979.