This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TVT.2025.3591733, IEEE Transactions on Vehicular Technology

Efficient Driving Behavior Narration and Reasoning on Edge Device Using Large Language Models

Yizhou Huang, Yihua Cheng, Kezhi Wang, *Senior Member, IEEE*

***Abstract*—Large language models (LLMs) with robust reasoning capabilities have significantly advanced the development of autonomous driving technologies, particularly in the narration and reasoning of driving behaviors, which hold substantial importance for accident analysis and traffic management. However, traditional deployment of these models relies on cloud servers, resulting in high latency and training costs, making it challenging to meet the stringent real-time requirements of autonomous driving scenarios. Recent studies suggest that edge computing, by deploying models closer to the data source, offers a promising solution to these issues. While existing general-purpose LLMs excel in video understanding and task reasoning, their generalization capabilities in rapidly changing traffic scenarios remain questionable. This paper provides a valuable reference for deploying LLMs at the edge in autonomous driving contexts. By leveraging real-world 5G networks for rapid deployment, we validate the performance and response speeds of various models in autonomous driving scenarios. Furthermore, we introduce an innovative prompt engineering strategy that enhances model performance by 25% without changing model parameters through minimal prompt tuning. Experimental results demonstrate that LLMs deployed on edge devices achieve satisfactory response times. Tests on the OpenDV-YouTube dataset further confirm that our prompt strategy significantly improves the performance of driving behavior narration and reasoning.**

***Index Terms*—Autonomous driving, Large language model, Edge computing.**

I. Introduction

In the field of autonomous driving [1], deep learning models [2] play a pivotal role due to the powerful feature learning capabilities, end-to-end learning processes, and the ability to integrate multi-modal data. These strengths contribute to the increased reliability, safety, and efficiency of autonomous driving technologies in real-world scenarios. Driving behavior description [3], a critical sub-task within autonomous driving, involves the deep understanding and precise interpretation of vehicle behavior in various traffic environments. In this task, LLMs [4]–[7] demonstrate exceptional performance, particularly due to their strong reasoning and contextual understanding abilities. This makes LLMs highly effective for addressing high-level decision-making challenges. Additionally, LLMs possess the capability to combine driving rules with natural language, enabling them to generate explanatory narratives for driving behaviors.

Although LLMs have demonstrated remarkable performance in autonomous driving scenarios, their deployment typically relies on cloud servers. This reliance results in end-to-end latency for cloud-based LLMs that often fails to meet the stringent delay requirements [8] in autonomous driving scenarios. Cloud-based LLM research, leveraging hardware optimization [9], [10] and clustered computing power, has demonstrated its potential for latency reduction, making real-world deployment in autonomous driving feasible. In contrast, edge computing addresses the high latency and computational costs of LLMs by deploying resources closer to data sources, enabling lightweight and efficient single-GPU solutions [11], [12] as an alternative for autonomous driving tasks. On the other hand, existing studies rely on simulated environments for evaluation [13], [14], overlooking the impact of real-world network conditions on LLM response time. This suggests that the effectiveness of LLMs in real-world networks for autonomous driving tasks still requires further exploration.

In this paper, we propose a framework that integrates LLMs with a laptop to simulate the use of 5G networks to evaluate the performance of edge LLMs in specific driving behavior tasks. This approach combines the strengths of LLMs in understanding complex semantics and reasoning with the powerful image-processing capabilities of visual encoders, resulting in more efficient and flexible descriptions of driving scenes. Specifically, the visual encoder analyzes and extracts key visual features from the driving environment, such as environment change and motion change, which are then fed into the LLM for further processing. We deploy LLMs across multiple roadside units (RSUs), each simulated on a laptop. These RSUs are interconnected via 5G NR/NSA technology, enabling decentralized deployment. We deploy the LLMs across three laptops; each laptop performs as a roadside unit (RSU), covering a specific area and interconnected through 5G NR/NSA technology [15], enabling decentralized deployment. Within this framework, each laptop [1]processes only the traffic data from its coverage area, thereby avoiding redundant operations and mitigating data congestion.

Our work focuses on evaluating the latency and performance

Corresponding author: Kezhi Wang

This work was partly supported by Eureka i2D-MSW: intelligence to Drive — Move-Save-Win (with funding from Innovate UK project under Grant No. 10071278) and Horizon Europe COVER project, No. 101086228 (with funding from UKRI grant EP/Y028031/1). K. Wang would like to acknowledge the support in part by the Royal Society Industry Fellowship (IF/R2/23200104).

Yizhou Huang and Kezhi Wang are with the Department of Computer Science, Brunel University of London, UB8 3PH, UK (email: yizhou.huang2@brunel.ac.uk; kezhi.wang@brunel.ac.uk).

Yihua Cheng is with the School of Computer Science, University of Birmingham, B15 2TT, UK (email: y.cheng.2@bham.ac.uk)

[1]We omit mention of the laptop-based RSU simulation in the following sections.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TVT.2025.3591733, IEEE Transactions on Vehicular Technology

of LLMs in handling driving behavior narration and reasoning tasks on edge devices. We conduct real-world deployment and testing using actual network conditions. Experimental analysis shows that edge devices and 5G networks can effectively reduce the latency of LLMs. However, the performance of vanilla LLMs remains suboptimal, raising two key questions:

- Does the LLM struggle to recognize a new environment?
- Can the LLM recognize the new environment but fail to generate the desired outcome?

To address these questions, we propose a three-stream prompt strategy using multi-modal information, consisting of environmental, agent, and motion streams. These streams convert the extracted features into structured natural language descriptions and reasoning prompts, guiding the LLM to generate context-specific responses. Our prompt engineering allows LLMs to rapidly improve their understanding of driving scenarios. During prompting, we freeze the model parameters, relying solely on pretrained LLMs for evaluation. Compared to computationally intensive fine-tuning, our strategy is significantly more suitable for rapid deployment on edge devices.

Our work provides a valuable reference for the performance and response time of deploying general LLMs on 5G edge devices for driving behavior narration and reasoning tasks. We propose a three-stream prompting strategy that effectively leverages multimodal prompt engineering to improve LLM performance, enabling rapid edge deployment. We propose a three-stream prompting strategy that enhances LLMs' contextual understanding of driving scenarios through prompt learning. With minimal computational resources, this approach achieves a 25% performance improvement in specific driving behavior tasks.

II. Task Definition

This work aims to evaluate the performance and response time of LLMs in driving behavior narration and reasoning for specific autonomous driving scenarios. We develop a framework where LLMs deploy on edge devices in real-world conditions via cellular networks. We aim to input image sequences into LLMs to generate driving behavior narration and reasoning. The narration task focuses on generating contextually relevant keywords for environment, agents, and motion, updating its knowledge base. The reasoning task builds upon the learned knowledge base, requiring LLMs to generate explanations for driving behavior narrations. The reasoning process follows a causal inference rule, ensuring that explanations establish keyword-triggered causal relationships, for example, "Due to Keyword 1, Keyword 2 occurs, leading to Keyword 3's action."

III. Method

A. Overview

Our task aims to test the performance of LLM on edge devices, including the accuracy of narration and reasoning of driving behavior and the response time. We designed a framework where the LLM is deployed independently on the RSU with its parameters frozen, without fine-tuning. This setup avoids redundant computation and reduces queuing delays during data transmission. Freezing the parameters ensures that the LLMs can be deployed quickly without the need for complex parameter adjustments. The framework we designed is an LLM deployed on RSUs individually; this setup avoids redundant computations and reduces queuing delays during data transmission, minimizing response latency.

B. LLMs with Edge Device

We propose a framework to integrate LLMs with edge devices in this section. In our framework, we use three laptops to simulate three edge servers and a 5G router provided by a network operator for 5G cellular communication. We connected the three RSUs to the 5G router and deployed LLMs on each. A single video clip was split into three sequential parts and fed into the RSUs in order. Our goal is for the LLMs on each RSU to provide coherent reasoning and explanations based on the video content, which will be evaluated against a baseline of human annotations. Additionally, as illustrated in Figure 1(a), we designed a visualization window at the end of the LLM to simulate road condition information uploaded by pedestrians.

One key strength of this system is its ability to enable information sharing between RSUs. When an LLM on one RSU detects an event, such as speeding or an accident, this information is immediately communicated to neighboring RSUs. For instance, if the first RSU detects a speeding vehicle, it can alert the second RSU, which can then warn nearby vehicles and pedestrians. This inter-unit communication helps create a safer driving environment by predicting potential dangers before they escalate. The warnings and information are transmitted over the 5G network, ensuring timely and reliable message delivery.

C. Multi-modal Prompt Strategy

The LLMs deployed on the RSUs are not specifically utilized for autonomous driving and driving behavior tasks. To further enhance the LLM's narration and reasoning capabilities, we design a multi-modal prompt strategy to improve the performance of the LLM in narration and reasoning.

We input images that contain different environments, agents (pedestrians and vehicles), and motion information into general LLMs. The task's objective is to ensure that the LLMs can perform driving behavior analysis using relevant information. During training, we notice that general models may inherently encode relevant information but are not explicitly utilized in generic tasks. To guide the general models to better gain relevant information for this specific task, as illustrated in Figure 1(b), we formulate the task definition so that the generated driving behavior descriptions must incorporate all three elements, which are designated as keywords. The task's rule requires each description to include the corresponding stream-specific keywords. For example, in a rainy environment, the generated description must contain the keyword indicating rain. Through the three-stream prompting strategy, the model is directed to focus on the environment, agent, and motion elements.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI: 10.1109/TVT.2025.3591733, IEEE Transactions on Vehicular Technology

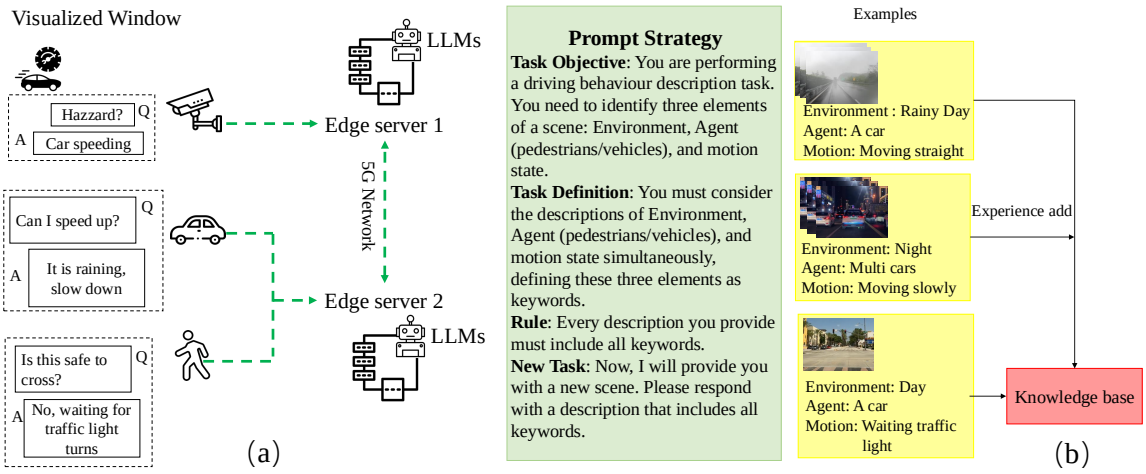

Fig. 1. (a). Overall system framework. The LLM deployed on the edge receives input data from edge users through a 5G cellular network. It analyzes input data to generate corresponding driving behavior narration and reasoning. LLM generates text-based outputs, which can be accessed through a real-time visualization window for backend queries. (b). Prompt details: LLMs learn environmental, agent, and motion information through prompt engineering to accomplish the narration task. According to the predefined prompting rules, the model continuously updates its knowledge base until it correctly learns all required features and generates appropriate responses.

Once the model's responses satisfy the task requirements, we proceed to reasoning analysis. Specifically, we define explicit task objectives and rules for the reasoning process. Since the model has already internalized the three-stream features within its knowledge base, we further guide it to retrieve narration memory from past learning to generate contextually aligned explanations. For example, if the model has learned the characteristics of a nighttime driving scenario, but during a reasoning task incorrectly attributes vehicle deceleration to rainy conditions, we intervene by reinforcing that the slowdown is due to low visibility in a nighttime environment. This correction mechanism further strengthens the model's knowledge base, ensuring improved consistency and accuracy in reasoning.

Environmental information contains weather and lighting, which significantly impact driving behavior and decisions. It ensures the model considers changes in visibility, road friction, and other environmental factors that may affect vehicle dynamics or necessitate more cautious driving. Agent information directs the model's attention to other entities in the driving environment, such as nearby vehicles and pedestrians. It aids in detecting interactions like lane merging, overtaking, or pedestrian crossings—crucial aspects of safe driving that are challenging to identify without specific prompts. Motion information analyzes the movements of the surrounding agents to identify events like sudden braking, sharp turns, and speeding. It enables the model to assess these situations and generate responses, such as recommending slowing down when a vehicle ahead brakes suddenly.

During prompting, we manually select the first frame that marks an environmental change. For instance, if frame N represents a street condition and frame N+1 transitions to a tunnel condition, frame N+1 is identified as the beginning of the new environment. We apply the same strategy for agent-related prompts. Our observations suggest that LLMs are more sensitive to sudden changes in these two types of prompts. When provided with repeated environmental prompts, LLMs do not continuously update their internal knowledge. In contrast, for motion prompts, we input a continuous sequence of frames, as motion states vary at each frame, enabling LLMs to dynamically update their knowledge.

We select 23 environmental information to serve as prompts, with each information representing a specific environmental condition, such as a rainy day, fog, or nighttime. We first input images containing one or more of these environmental information, along with their corresponding textual descriptions, into the LLM, allowing it to learn and retain this information and their related descriptions. Similarly, we select 15 agent information as prompts, including pedestrians, vehicles, and cyclists, and 47 motion information, such as turning, stopping, lane changing, and crossing. After the LLM forms a memory of individual information, we combine environment, agent, and motion information in pairs to enrich scene representation and evaluate the performance improvement of dual-prompt combinations. After confirming the effectiveness of this strategy, we integrate all three prompts to achieve the optimal results. This iterative and layered approach ultimately forms a comprehensive multimodal prompt, enabling the LLM to describe and interpret complex scenes from the perspectives of environment, agents, and motion.

As shown in Figure 2, when the prompt strategy is enabled, the LLM is able to generate more specific, context-aware outputs that better capture the complexities of driving scenarios. For example, the model can describe a situation where visibility is reduced, a vehicle ahead is prohibited from passing, and no pedestrians are crossing, while also providing an analysis of the underlying driving behaviors.

IV. Experiment and Discussions

A. Dataset

The OpenDV-YouTube dataset [16], developed by OpenDriveLab, constitutes a large-scale multimodal benchmark

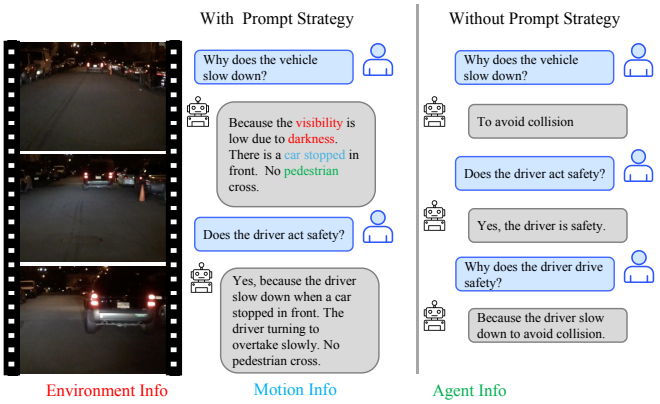

Fig. 2. Comparison between enabling and disabling three-stream prompts. LLM is able to generate descriptions of driving behavior based on three streams. The observed results can trigger various keywords, such as the environment keyword "visibility," the agent keyword "pedestrian crossing," and the motion keyword "stop," among others.

specifically designed for video-language understanding in autonomous driving contexts. To facilitate efficient model development, the authors provide OpenDV-mini—a curated subset containing 28 hours of driving. All LLMs in our study were rigorously evaluated using this standardized OpenDV-mini benchmark; we sampled 2,000 images for evaluation after prompt tuning.

B. Experimental Settings

We implemented all models in PyTorch and validated them using three NVIDIA RTX 3090 GPUs, each equipped with 24 GB of LPDDR4X memory. We compared edge devices such as the Mobilint MLA100, which supports up to 32GB of VRAM and is equipped with high-speed chips designed for AI inference. It can deploy LLMs up to 70B parameters at the edge; this enables the deployment of LLMs on edge devices.

In our study, we employ BERTScore to quantify the semantic similarity between texts generated by large language models (LLMs) and the annotations provided in the dataset. Specifically, we transform both the generated text from LLMs and the dataset annotations into token embeddings. Utilizing BERT, we then compute the token-level alignment between the generated text and the annotated text, subsequently generating an F1-score. A score approaching 1 indicates a high degree of similarity between the generated text and the dataset annotations, whereas a score of 0 denotes complete dissimilarity. Following this, we convert the 0-1 score into a percentage format, which serves as a performance metric for our evaluation. This approach allows for a nuanced understanding of the semantic fidelity of LLM-generated content relative to established dataset benchmarks.

C. Advantages of Multi-modal Prompt Strategy

We conducted a comprehensive series of tests on Video Chat [4], LLaMa Adapter [5], Video LLaMa [6], and Video-ChatGPT [7] to thoroughly assess their performance in terms of narration accuracy and reasoning correctness, as summarized in Table I. For consistency, the input provided to all four large language models was kept uniform, utilizing raw image data as the primary source. This consistency in input ensured that the results were directly comparable across different models. We designed two distinct sets of experiments: one set with the prompt strategy enabled and the other disabled. This approach allowed us to effectively evaluate the impact of the prompt strategy on the performance of these models, providing insights into how prompt-based optimization influences both narration and reasoning tasks.

TABLE I
THE EFFECT OF THE PROPOSED MULTI-MODAL PROMPT STRATEGY IN NARRATION AND REASONING TASKS, PS REFERS TO THE PROMPT STRATEGY.

Task	PS	Models			
		Video Chat	LLaMa-Ada	Video-LLaMa	Video-ChatGPT
Nar.	✓	76.9%	70.3%	74.1%	78.2%
	×	67.2%	56.3%	59.5%	64.9%
Rea.	✓	71.3%	68.1%	65.2%	81.7%
	×	51.4%	39.38%	44.7%	54.5%

Video-ChatGPT performs the best with the prompt strategy enabled, achieving a narration accuracy of 78.2% and a reasoning correctness of 81.7%. However, when the prompt strategy is disabled, these values drop to 64.9% and 54.5%, respectively. LLaMa Adapter and Video LLaMa, also show noticeable performance declines when the prompt strategy is disabled, especially in reasoning correctness, where LLaMa Adapter drops from 68.1% to 39.38% and Video LLaMa from 65.2% to 44.7%. This highlights the crucial role the prompt strategy plays in enhancing the models' understanding and handling of driving behaviors, particularly in reasoning.

To explore the performance differences between prompted edge LLMs and deep learning models on the same driving behavior task. We manually reproduced the ADAPT model and trained it using the OpenDV-YouTube dataset to establish a performance baseline. The results for ADAPT presented in Table II were obtained from the validation set, providing a benchmark for comparison against the LLMs.

D. Response Speed of LLMs

We tested the response speeds of these four LLMs alongside the traditional deep learning method ADAPT [17]. For this aspect of the evaluation, we introduced a variety of conditions by dividing image frames into intervals of 1, 15, and 30. This was done to measure response times under different levels of input frequency.

We conducted 20 experiments and averaged the results to minimize error fluctuations. The result in Table III for all experimental subjects is the sum of the network transmission time and the LLMs processing time. LLaMa-Adapter performs the fastest under all conditions, taking 50 ms, 70 ms, and 75 ms to process 1, 15, and 30 frames, respectively. Video Chat-GPT, Video Chat, and Video LLaMa also demonstrate quick response times, particularly when handling smaller batches of images. These results demonstrate that well-deployed edge LLMs can effectively meet the low-latency and high-response

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
Citation information: DOI: 10.1109/TVT.2025.3591733, IEEE Transactions on Vehicular Technology

TABLE II
ACCURACY OF LLMS AND CONVENTIONAL METHOD ADAPT. LLMS ARE NOT TRAINED ON THE DATASET AND SHOW REMARKABLE GENERALIZATION ABILITY. .

Task	ADAPT	Video-GPT	Video-LLaMa	Video-Chat	LLaMa-Ada
Nar.	89.7%	78.2 %	74.1%	76.9%	70.3%
Rea.	90.3%	81.7%	65.2%	71.3%	68.1 %

TABLE III
TOTAL RESPONSE TIME OF LLMS ON EDGE USING 5G NETWORK. LLAMA ADA REPRESENTS LLAMA ADAPTER

Model	Frames	Response time		
		Uploading	Inference	Total
Video Chat	#1	25 ms	50 ms	75 ms
	#15	210 ms	85 ms	295 ms
	#30	430 ms	105 ms	535 ms
LLaMa-Ada	#1	30 ms	50 ms	80 ms
	#15	240 ms	70 ms	310 ms
	#30	450 ms	75 ms	525 ms
Video-LLaMa	#1	30 ms	55 ms	85 ms
	#15	230 ms	85 ms	315 ms
	#30	470 ms	90 ms	560 ms
Video-ChatGPT	#1	25 ms	50 ms	75 ms
	#15	220 ms	80 ms	300 ms
	#30	450 ms	95 ms	545 ms

speed requirements of autonomous driving scenarios. In addition, leveraging the efficient parallel processing of GPUs, the inference time of LLMs does not increase linearly with the number of image frames. Specialized chips designed for edge AI inference can further accelerate LLM execution. Notably, we did not directly compare LLMs' response time with traditional deep learning models such as ADAPT, as the latter requires parameter updates. Retraining a deep learning model on the OpenDV-mini dataset entails substantially higher computational costs than leveraging prompt engineering with LLMs, making the latter a more efficient alternative in deployment-sensitive applications.

E. Discussion

We validated the potential of edge LLMs in handling autonomous driving narration and reasoning tasks in real-world deployment. The combination of 5G high-speed networks and edge devices can meet the low-latency requirements of autonomous driving. While advanced image compression techniques may further reduce latency and enhance safety. However, edge devices are constrained by a single GPU, typically limiting LLM deployment to within 70B parameters. In contrast, cloud-based computing remains superior to edge devices, enabling the deployment of larger-scale LLMs (e.g., 128B models) to further enhance the accuracy of driving behavior narration and reasoning. Exploring the integration of cloud-based fine-tuned LLMs with rapidly deployed edge LLMs may be a promising direction.

V. CONCLUSION

In this paper, we propose a framework that integrates large language models and edge devices, along with a multi-modal prompt strategy to enhance the accuracy of narration and reasoning of driving behavior on edge devices. After enabling the multi-modal prompt strategy, the overall performance of the LLM improved significantly. Furthermore, deploying the LLM directly on RSUs via 5G communication technology allows for real-time data processing at the source, significantly reducing the latency associated with data queuing and processing delays. By handling data closer to the source, this approach minimizes waiting times and enhances overall system efficiency.

REFERENCES

[1] Y. Huang, Y. Cheng, and K. Wang, "Trajectory Mamba: Efficient attention-mamba forecasting model based on selective SSM," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, Jun. 2025. accepted for publication.
[2] 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.
[3] R. Bhattacharyya, B. Wulfe, D. J. Phillips, A. Kuefler, J. Morton, R. Senanayake, and M. J. Kochenderfer, "Modeling human driving behavior through generative adversarial imitation learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 2874–2887, 2022.
[4] K. Li, Y. He, Y. Wang, Y. Li, W. Wang, P. Luo, Y. Wang, L. Wang, and Y. Qiao, "Videochat: Chat-centric video understanding," *arXiv preprint arXiv:2305.06355*, 2023.
[5] S. Gao, J. Yang, L. Chen, K. Chitta, Y. Qiu, A. Geiger, J. Zhang, and H. Li, "Vista: A generalizable driving world model with high fidelity and versatile controllability," *arXiv preprint arXiv:2405.17398*, 2024.
[6] H. Zhang, X. Li, and L. Bing, "Video Llama: An instruction-tuned audio-visual language model for video understanding," *arXiv preprint arXiv:2306.02858*, 2023.
[7] M. Maaz, H. Rasheed, S. Khan, and F. S. Khan, "Video-ChatGPT: Towards detailed video understanding via large vision and language models," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, 2024.
[8] Q. Dong, X. Chen, and M. Satyanarayanan, "Creating edge AI from cloud-based LLMs," in *Proceedings of the 25th International Workshop on Mobile Computing Systems and Applications*, pp. 8–13, 2024.
[9] P. Patel, E. Choukse, C. Zhang, Í. Goiri, B. Warrier, N. Mahalingam, and R. Bianchini, "Polca: Power oversubscription in LLM cloud providers," *arXiv preprint arXiv:2308.12908*, 2023.
[10] C. Holmes, M. Tanaka, M. Wyatt, A. A. Awan, J. Rasley, S. Rajbhandari, R. Y. Aminabadi, H. Qin, A. Bakhtiari, L. Kurilenko, *et al.*, "Deepspeed-Fastgen: High-throughput text generation for LLMs via MII and Deepspeed-Inference," *arXiv preprint arXiv:2401.08671*, 2024.
[11] K. B. Kan, H. Mun, G. Cao, and Y. Lee, "Mobile-Llama: Instruction fine-tuning open-source llm for network analysis in 5g networks," *IEEE Network*, 2024.
[12] G. Qu, Q. Chen, W. Wei, Z. Lin, X. Chen, and K. Huang, "Mobile edge intelligence for large language models: A contemporary survey," *IEEE Communications Surveys & Tutorials*, 2025.
[13] D. Fu, W. Lei, L. Wen, P. Cai, S. Mao, M. Dou, B. Shi, and Y. Qiao, "Limsim++: A closed-loop platform for deploying multimodal LLMs in autonomous driving," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1084–1090, 2024.
[14] M. Wu, F. R. Yu, P. X. Liu, and Y. He, "Facilitating autonomous driving tasks with large language models," *IEEE Intelligent Systems*, vol. 40, no. 1, pp. 45–52, 2025.
[15] A. Dogra, R. K. Jha, and S. Jain, "A survey on beyond 5G network with the advent of 6G: Architecture and emerging technologies," *IEEE access*, vol. 9, pp. 67512–67547, 2020.
[16] J. Yang, S. Gao, Y. Qiu, L. Chen, T. Li, B. Dai, K. Chitta, P. Wu, J. Zeng, P. Luo, J. Zhang, A. Geiger, Y. Qiao, and H. Li, "Generalized predictive model for autonomous driving," 2024.
[17] B. Jin, X. Liu, Y. Zheng, P. Li, H. Zhao, T. Zhang, Y. Zheng, G. Zhou, and J. Liu, "Adapt: Action-aware driving caption transformer," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7554–7561, IEEE, 2023.