Driving with LLMs: Fusing Object-Level Vector Modality for Explainable Autonomous Driving

Anika Raisa Chowdhury
araisa@umich.edu

Anubhuti Hiwase
anubhuti@umich.edu

Tanvi Shah
tanvisha@umich.edu

***Abstract*—This paper discusses a new approach to using Large Language Models (LLMs) in autonomous driving. The researchers created a unique LLM architecture that combines numeric information with a pre-trained language model to better understand driving situations. They also developed a dataset with 160,000 question-answer pairs based on 10,000 driving scenarios, where the questions were generated by both a reinforcement learning agent and a teacher language model (GPT-3.5). The researchers used a specific pretraining strategy to align numeric information with the language model's representations. They also introduced an evaluation metric for Driving Question-Answering (QA) and demonstrated that their LLM-based system is proficient in interpreting driving scenarios, answering questions, and making decisions. The findings suggest that LLMs could be valuable for generating driving actions compared to traditional methods like behavioral cloning. .**

***Index Terms*—LLM, Explainable AI, Autonomous Driving, Question-Answering**

I. Introduction

The researchers in this paper are exploring the use of Large Language Models (LLMs), which have shown advanced capabilities indicative of early signs of artificial general intelligence (AGI), in the context of autonomous driving and robotics. These LLMs demonstrate abilities like reasoning beyond known scenarios, understanding common sense, retrieving knowledge, and effectively communicating with humans. Autonomous driving systems often face challenges in terms of being opaque or ”black boxes” in their decision-making processes, making it difficult to provide them with out-of-distribution reasoning and interpretability. Traditional approaches struggle to address these issues. The researchers propose a novel approach to leverage LLMs by integrating a numeric vector modality, commonly used in robotics to represent speed, actuator positions, and distance measurements. This numeric modality is more compact than visual information, reducing scaling challenges. The researchers fuse this numeric information, specifically object-level 2D scene representation used in autonomous driving, into a pre-trained LLM with adapters. This integration allows the model to directly interpret and reason about complex driving situations, empowering LLMs to act as the ”brain” of autonomous driving systems. The LLM interacts directly with simulators to facilitate reasoning and predict driving actions.

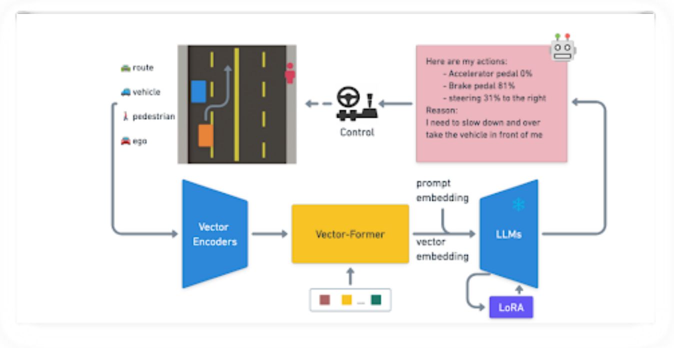

Fig. 1. An overview of the architecture for Driving with LLMs, demonstrating how object-level vector input from our driving simulator is employed to predict actions via LLMs

II. Driving with LLMs architecture:

In this paper, the authors make the following contributions: They propose a novel architecture that fuses an object-level vectorized numeric modality into any LLMs with a two-stage pretraining and fine-tuning method Driving scenario Q/A dataset: They came up with a 160k question-answer pairs dataset on 10k driving situations with control commands, collected with RL expert driving agents and an expert LLM-based question answer generator They introduce a unique approach to assess the performance of Driving Question-Answering by employing an expert LLM grader. Additionally, we offer initial evaluation outcomes along with a baseline utilizing our comprehensive multimodal architecture designed for end-to-end processing

Why we think this academic work is interesting: This work is extremely interesting to us as this is a first of its kind approach to integrate LLMs into driving scenarios in simulation. This includes a comprehensive framework encompassing the simulator, automatic data collection, integration of a new object-level vector modality into LLMs, and the GPT-based evaluations approaches. This work taps into advanced capabilities of LLMs and suggests potential application of these models beyond traditional language tasks.By merging numeric information crucial for robotics with language models, the research takes an interdisciplinary approach, addressing limitations in traditional methods used in autonomous driving.

Also this work and approach was published very recently (3rd October 2023). This means that this project group would be amongst the first to attempt the implementation of the researcher's work. This opportunity was extremely exciting for the team and thus inspired us to work on this project.

III. Related works:

A. End-to-End Autonomous Driving Systems (2.1):

Recent advances in autonomous driving systems using end-to-end deep learning lack interpretability in decision-making.[1], [2], [3]. A fundamental challenge that remains with modern autonomous driving systems is the lack of interpretability in the decision making process [4]. Understanding why a decision is made is crucial for identifying areas of uncertainty, building trust, enabling effective human-AI collaboration and ensuring safety [5].The paper responds to this issue by introducing a novel approach that incorporates both vector and textual modalities, along with pre trained Large Language Models (LLMs) to enhance understanding of the decision-making process.

B. Interpretability of Autonomous Driving Systems (2.2):

Various methods have been developed [6] to comprehend the decision-making processes of deep neural networks. Established model-agnostic interpretability techniques, like those outlined in [7], [8], and [9], provide explanations for individual predictions. Additional approaches, such as gradient-based methods [10], saliency maps [11], and attention maps [12], delve into the internal workings of models to elucidate the decision-making process. In the realm of autonomous vehicles, [13] introduced visual attention maps highlighting influential areas in driving images. In [14], attention-based methods were combined with natural language to craft a controller offering action descriptions and explanations based on image frames. This work was refined in [15], incorporating part-of-speech prediction and special token penalties. Recognizing the limitations of attention alone [16], efforts have been made to integrate this approach with other explanatory methods. For instance, [17] suggests explaining transformers by simultaneously leveraging attentive class activation tokens, encoded features, gradients, and attention weights. Expanding on this research, our proposal advocates for the use of textual modality to enhance explainability in the context of autonomous driving.

C. Multi-modal LLMs in Driving Tasks (2.3)

Recently, a prominent trend involves integrating various modalities into comprehensive models. Notable instances include Visual Language Models (VLMs) like [18], [19], [20], and [21], effectively merging language and images for tasks such as image captioning and visual question answering. Another significant advance [22] combines information from six modalities, expanding content generation possibilities and enabling diverse multi-modal search capabilities. In the domain of autonomous driving, incorporating language has been natural through VLMs. For instance, [22] employs images and language for training a driving policy, while [23] proposes a method for vehicle control with human assistance, utilizing natural language to explain visual observations and predict appropriate actions. Similarly, in robotics, [24] integrates language with point clouds for object identification. Although [25] utilizes LLMs for low-level robotics control, our novel methodology involves fusing numeric vector modality with language for interpretable control and driving QA tasks, a unique contribution in the domain of autonomous vehicles.

D. Integration of Language in Robotics (2.3)

While efforts in robotics have fused language with various modalities, the paper distinguishes itself by introducing a pioneering methodology. Unlike previous works confined to vision modalities, this research uniquely fuses numeric vector modality with language in the domain of autonomous vehicles. The goal is to facilitate interpretable control and enhance the understanding of decision-making processes in autonomous driving.

MODEL ARCHITECTURE

Training the Driving LLM Agent: Training the LLM-Driver involves framing it as a Driving Question Answering (DQA) challenge, incorporating an object-level vector modality into pre-trained LLMs. The process has two stages:

- **First Stage:** Grounding the vector representation into an embedding
  - Freeze the language model
  - Optimize vector encoders and transformers
- **Second Stage:** Fine-tuning the model for DQA tasks
  - Enable the model to answer driving-related questions
  - Take appropriate actions

The model architecture includes three key components as demonstrated in Fig 2:

- **Vector Encoder:** Processes four input vectors through Multilayer Perceptron (MLP) layers, incorporating a cross-attention layer to move them into a latent space. The ego feature is added to emphasize ego states.
- **Vector Former:** Contains self-attention layers and a cross-attention layer working with latent space and question tokens, transforming latent vectors into an embedding decodable by the LLM.
- **LLM with Adaptor:** Injects trainable rank decomposition matrices (LoRA) into linear layers of the pretrained LLMs for efficient fine-tuning, utilizing LLaMA-7b as the pretrained LLM.
- **LLM with Adaptor:** Injects trainable rank decomposition matrices (LoRA) into linear layers of the pretrained LLMs for efficient fine-tuning, utilizing LLaMA-7b as the pretrained LLM.

Vector Representation Pre-training:: Integrating a new modality into pre-trained Large Language Models (LLMs) poses challenges. A novel approach leverages structured language to bridge the gap between vector space and language embeddings, focusing on numerical tokens. During pre training, the language model is frozen, and the entire framework

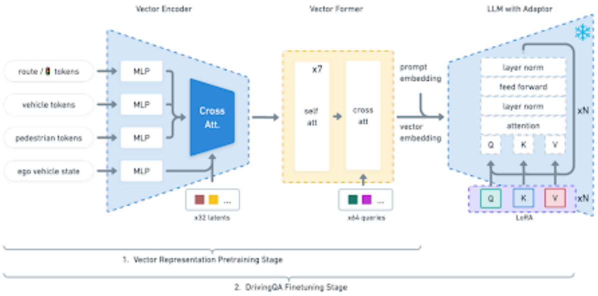

Figure 3: The architecture of the Driving LLM Agent

Fig. 2. The Model Architecture

is optimized end-to-end to ground the vector representation effectively. The pretraining process involves using perception structured-language labels and avoids reasoning tasks, focusing solely on representation training.

Driving Question-Answer Fine tuning: After pre-training, the trainable LoRA module is integrated, and weights for Vector Encoder, Vector Former, and LoRA module are optimized end-to-end on collected Driving QA data. Action-triggering questions are added to the VQA dataset to train the LLM-Driver to generate actions. These questions require the agent to infer actions based on vector input. The model is refined on 10,000 scenarios, resulting in a driving LLM agent capable of reasoning about actions and responding to driving-related questions based on object-level vectors.

Implementation and Evaluation

After pre-training, the trainable LoRA module is integrated, and weights for Vector Encoder, Vector Former, and LoRA module are optimized end-to-end on collected Driving QA data. Action-triggering questions are added to the VQA dataset to train the LLM-Driver to generate actions. These questions require the agent to infer actions based on vector input. The model is refined on 10,000 scenarios, resulting in a driving LLM agent capable of reasoning about actions and responding to driving-related questions based on object-level vectors.

In the evaluation of the model's performance in our implementation, several key metrics were analyzed. The Traffic Light Accuracy (TL Accuracy) achieved a commendable 70 percent, indicating the model's ability to correctly identify the state of traffic lights. The Mean Absolute Error (MAE) for Traffic Light Distance prediction stood at 0.7733, showcasing the model's proficiency in estimating distances with a relatively low margin of error. Notably, the model exhibited accuracy in predicting errors related to cars, with a Car Error MAE of 0.2, and errors related to pedestrians, achieving a Pedestrian Error MAE of 0.5. The Average Control Error, measured at 0.06999999999999999 for one aspect and 0.018000000000000006 for another, further highlights the model's effectiveness in control-related predictions. The evaluation process, including run time, demonstrated robust computational efficiency, with a total time investment of 1613.987 seconds. These results collectively underscore the model's promising performance across various aspects of traffic prediction and control error estimation. Fig 3 shows the the reported metrics, the authors calculated the Mean Absolute Error (MAE) for the predictions of the number of cars and pedestrians, denoted as Ecar and Eped respectively. Additionally, we measure the accuracy of traffic light detection as well as the mean absolute distance error in meters for traffic light distance prediction, represented as AccTL and DTL. Furthermore, we compute the MAE for normalized acceleration and brake pressure denoted as Elon., and normalized steering wheel angle denoted as Elat.. Lastly, we report the weighted cross-entropy loss for the token prediction on the evaluation set, indicated as Ltoken.[26].The original work used 10k Q-A pairs for training and 1k Q-A pairs for testing.

	agents count		*traffic light*		*action*		*loss*
	$E_{car}\downarrow$	$E_{ped}\downarrow$	$Acc_{TL}\uparrow$	$D_{TL}\downarrow$	$E_{lon.}\downarrow$	$E_{lat.}\downarrow$	$L_{token}\downarrow$
Perceiver-BC[46]	0.869	0.684	**0.900**	**0.410**	0.180	0.111	n/a
LLM-Driver$_{\text{w/o pretrain}}$	0.101	1.668	0.758	7.475	0.094	0.014[a]	0.644
LLM-Driver$_{\text{w/ pretrain}}$	**0.066**	**0.313**	0.718	6.624	**0.066**	**0.014**[b]	**0.502**

[a] Exact value: 0.01441
[b] Exact value: 0.01437

Fig. 3. Results from the reference study

Comparing the our evaluation results and with that of the authors reveals notable improvements in the performance of the LLM-Driver model. In terms of Traffic Light Accuracy (TL Accuracy), the previous evaluation achieved 70 percent, while the LLM-Driver, both with and without pretraining, surpassed this with scores of 71.8 percent and 75.8 percent, respectively. The Mean Absolute Error (MAE) for Traffic Light Distance (DT) prediction witnessed a substantial decrease from 0.7733 to 0.066, indicating enhanced precision in estimating distances. Notably, the model demonstrated superior performance in various aspects, including control-related errors. The Average Control Error, previously at 0.06999999999999999, improved to 0.01437 (with pre-training) and 0.01441 (without pre-training). These results showcase the model's advancements, affirming its effectiveness in traffic prediction and control-related tasks.

Due memory limitations, our implementation was on a small subset of the data using 100 pairs for training and 10 pairs for testing. We used Wandb[1] to demonstrate our results. The Wandb demonstration of the LLM driver utilizes object-level vector input from our driving simulator can be found on this link[2].

Conclusion

In conclusion, the LLM-Driver represents a leap in the fusion of artificial intelligence and driving technology.In

[1] https://shorturl.at/axBJP
[2] csv file link

the project we explored the integration of numeric vector modality with LLMs in the context of autonomous driving. A major challenge was dealing with the entirety of the data due to limitations in processing and memory capabilities. Additionally, the scarcity of references hindered the ability to explore comparable bases or alternative implementation methods within the given time constraints. Despite these challenges, we achieved a significant milestone by working on a subset of the data, attaining a commendable 70 percent accuracy,. We intend to extend the work on a high GPU based systems to work on a larger chunk of data.In the realm of autonomous driving, the project's future scope is promising and multifaceted. One key avenue for exploration involves overcoming current limitations in memory and processing capabilities to enable a thorough analysis of a more extensive dataset. Concurrently, the project aims to delve into alternative implementation methods and architectures, seeking optimizations that can further enhance the efficiency of integrating numeric vectors with Large Language Models (LLMs). Additionally, the scope extends to real-time applications, envisioning the adaptation of the model for deployment in autonomous vehicles and driving simulations. This expansion ensures the model's responsiveness in dynamic driving scenarios, marking a pivotal step toward practical and real-world implementation.

github: https://shorturl.at/uENQS
dataset: https://shorturl.at/huyRV

Acknowledgments

We extend our heartfelt gratitude to Professor Jaerock Kwon for his invaluable guidance and support throughout our exploration of neural networks. His expertise and insights significantly enriched our understanding and contributed to the success of this engaging implementation.

References

[1] M. Bansal, A. Krizhevsky, and A. Ogale, "Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst," arXiv preprint arXiv:1812.03079, 2018.

[2] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "Vectornet: Encoding hd maps and agent dynamics from vectorized representation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11 525–11 533.

[3] J. Hawke, V. Badrinarayanan, A. Kendall, et al., "Reimagining an autonomous vehicle," arXiv preprint arXiv:2108.05805, 2021

[4] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," Information Fusion, vol. 58, pp. 82–115, 2020.

[5] W. Xu, "From automation to autonomy and autonomous vehicles: Challenges and opportunities for human-computer interaction," Interactions, vol. 28, no. 1, p. 48–53, dec 2020. [Online]. Available: https://doi.org/10.1145/3434580

[6] A. Holzinger, A. Saranti, C. Molnar, P. Biecek, and W. Samek, Explainable AI Methods - A Brief Overview. Cham: Springer International Publishing, 2022, pp. 13–38. [Online]. Available: https://doi.org/10.1007/978-3-031-04083-2-2

[7] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.

[8] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," Advances in neural information processing systems, vol. 30, 2017.

[9] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in International conference on machine learning. PMLR, 2017, pp. 3145–3153.

[10] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.

[11] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," arXiv preprint arXiv:1312.6034, 2013.

[12] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in International conference on machine learning. PMLR, 2015, pp. 2048–2057.

[13] J. Kim and J. Canny, "Interpretable learning for self-driving cars by visualizing causal attention," 2017.

[14] J. Kim, A. Rohrbach, T. Darrell, J. Canny, and Z. Akata, "Textual explanations for self-driving vehicles," 2018.

[15] M. A. Kühn, D. Omeiza, and L. Kunze, "Textual explanations for automated commentary driving," arXiv preprint arXiv:2304.08178, 2023.

[16] S. Jain and B. C. Wallace, "Attention is not explanation," arXiv preprint arXiv:1902.10186, 2019.

[17] Y. Qiang, D. Pan, C. Li, X. Li, R. Jang, and D. Zhu, "Attcat: Explaining transformers via attentive class activation tokens," in Advances in Neural Information Processing Systems, 2022.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021.

[19] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan, "Flamingo: a visual language model for few-shot learning," 2022.

[20] J. Li, D. Li, S. Savarese, and S. Hoi, “Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models,” 2023.

[21] OpenAI, “Gpt-4 technical report,” 2023.

[22] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, “Imagebind: One embedding space to bind them all,” 2023. [23] J. Roh, C. Paxton, A. Pronobis, A. Farhadi, and D. Fox, “Conditional driving from natural language instructions,” in Conference on Robot Learning. PMLR, 2020, pp. 540–551.