Bachelor Thesis / Master Thesis

Decision: Follow current lane
Reason: Required to drive on the right in Germany, too close to traffic light for lane change.

An exemplary scenario where a LLM decides between two possible maneuvers.

Leveraging Large Language Models for Decision-Making in Autonomous Driving

Arbitration graphs [1] are a decision-making framework where the behavior of an autonomous system is broken down into atomic behavior components and organized into a hierarchical structure. Arbitrators at each level of the hierarchy decide which behavior to execute based on their implemented policy.

This thesis investigates the potential of using large language models (LLMs), such as GPT, to improve decision-making processes within arbitration graphs. The reasoning and common-sense capabilities of LLMs present a unique opportunity to replace traditional cost arbitrators with more sophisticated, context-aware decision makers.

Unlike conventional algorithms, LLMs are not only capable of understanding complex scenarios, they also offer explanations for their decisions. This opens up new possibilities for transparant and explainable decision-making in autonomous systems.

I am happy to answer any questions you might have, just send me an email!

References
[1] Orzechowski, P., Burger, C., & Lauer, M. (2020). "Decision-Making for Automated Vehicles Using a Hierarchical Behavior-Based Arbitration Scheme"

Institute of Measurement and Control Systems (MRT)
Prof. Dr.-Ing. Christoph Stiller

Advisor:
Nick Le Large, M.Sc.

Programming language(s)[1]:
C++   proficient
Python   advanced

System, Framework(s):
Linux, Docker, PyTorch

Required skills:
- Prior experience with machine learning
- Prior knowledge of motion planning
- Capable of working independently

Language(s):
German, English

For more information please contact:

Nick Le Large

Room: 234
Phone: +49 721 608-43794
Email: nick.lelarge@kit.edu

Or directly send in your application including your current grades as well as our questionnaire!

---

[1] **skill levels:**
beginner   < 500 lines of code (LOC)
advanced   500 – 5000 LOC
proficient   > 5000 LOC