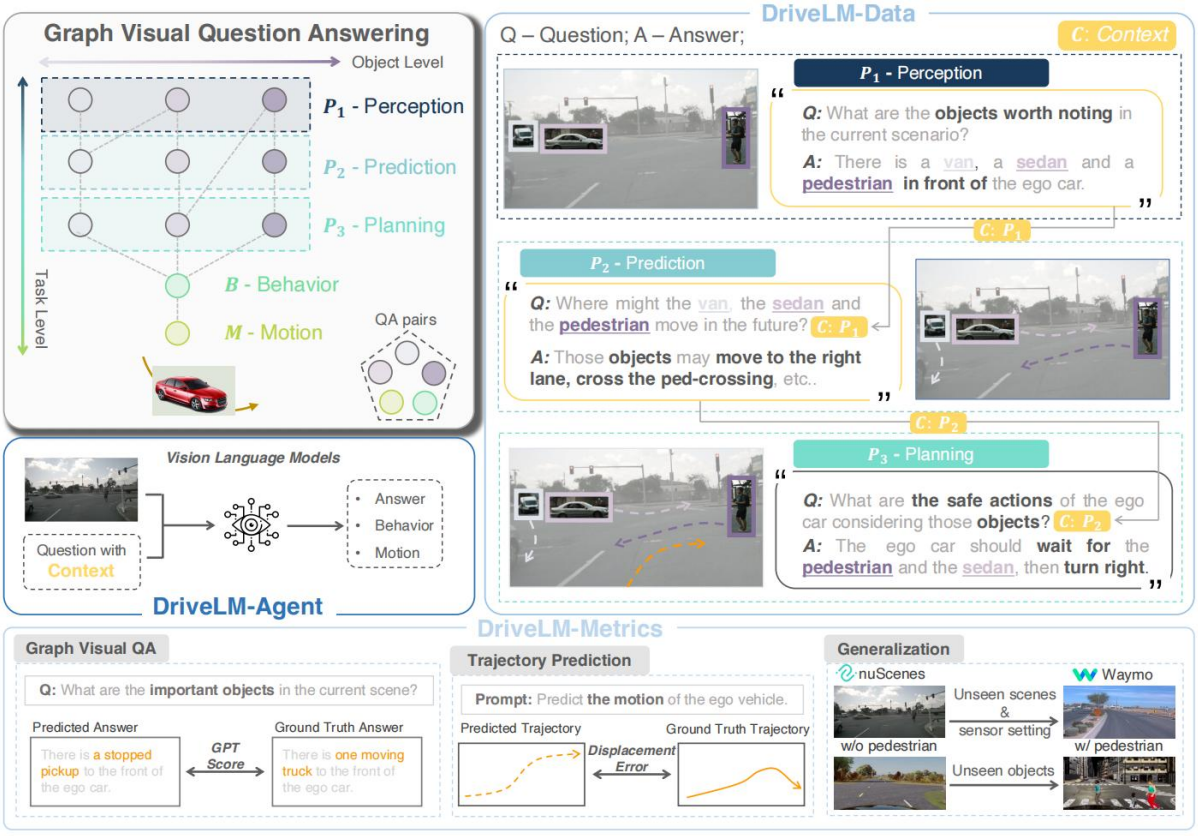
1. 项目概述

本项目的目标是优化 DriveLM 系统，使其能够高效运行。DriveLM 是一个面向自动驾驶任务的视觉语言模型（VLM），基于 DriveLM-nuScenes 数据集。DriveLM 通过图结构的视觉问答（Graph Visual Question Answering, GVQA）任务，模拟人类驾驶员的感知、预测和规划过程，从而提高模型的泛化能力和与人类的交互性。本项目的核心任务是探索多种优化技术，以减少 DriveLM 的计算占用并提高推理速度，同时保持其精度。通过本项目，学生将深入了解视觉语言模型在自动驾驶中的应用，并掌握模型优化的关键技术。

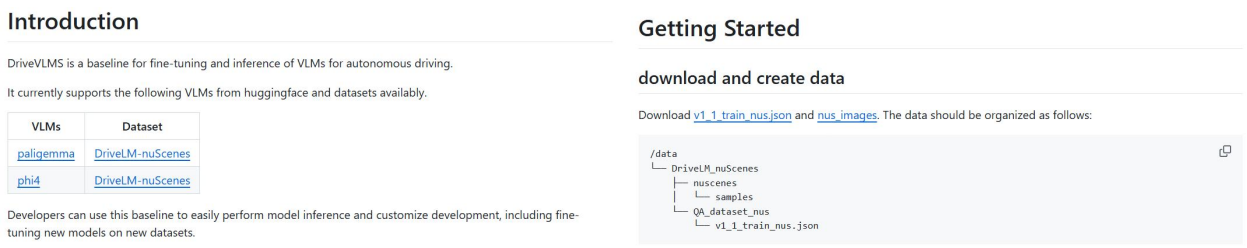


DriveLM 论文链接：https://arxiv.org/pdf/2312.14150

2. 项目细节

2.1 代码库与数据准备

我们提供了一个基于 DriveLM 模型和 DriveLM-nuScenes 数据的完整代码 (drivevlms.zip)，包含了数据及模型下载、模型微调、推理与演示，详细操作步骤请参考代码库的 readme。我们还提供了两个微调后的模型权重，后续给大家添加权限后可从 hugging face 下载，并基于微调后的权重进行优化。



Introduction

DriveVLMS is a baseline for fine-tuning and inference of VLMs for autonomous driving.

It currently supports the following VLMs from huggingface and datasets availably.

VLMs	Dataset
paligemma	DriveLM-nuScenes
phi4	DriveLM-nuScenes

Developers can use this baseline to easily perform model inference and customize development, including fine-tuning new models on new datasets.

Getting Started

download and create data

Download v1_1_train_nus.json and nus_images. The data should be organized as follows:

```
/data
└── DriveLM_nuScenes
    ├── nuscenes
    │   └── samples
    └── QA_dataset_nus
        └── v1_1_train_nus.json
```

Model	gradient_accumulation_steps:	GPU	GPU Memory	Time/Epoch	Total Epoch	DDP
paligemma	8	V100 32GB	26-30GB	18h/per epoch	2~6 个	speed*num_gpus 倍
phi4	8	V100 32GB	28-31.5GB	58h/ per epoch	1 个	Yes

2.2 优化方向

我们提供了以下八种基本优化方向，小组可以选择其中的几种进行实现：

(1) Prompt Engineering

- 任务介绍：针对已经训练好的模型参数，通过文本提示来引导预训练好的模型产生期望的输出。
- 要求：(1) 针对测试数据，提供一套有效的 prompt。 (2) 控制输出 token 长度，避免过多废话。

(2) 量化优化

- 任务介绍：对已经训好的 fp16 模型权重进行量化。
- 要求：(1) ptq 类量化方法 (2) int4 或更低 bit 位量化 (3) 相对精度不能低于 98%

(3) Self-Speculative Decoding

- 任务介绍：提出一种无需训练的有效的推理加速方案。
- 要求：加速比不低于 x2

(4) Encoder & Prefill 优化

- 任务介绍：可采用 pruning 等或者 early exit 等方法降低图像 encoder 和 llm prefill 耗时。
- 要求：(1) 相对精度不低于 98% （2）相对计算量降低到 80%以下。

(5) 模型 Backbone 替换

- 任务介绍：除了目前提供的 2 个模型，可以使用 huggingface 上其他开源模型进行模型优化。
- 要求：llm 部分参数量小于 3B 。

(6) 减少 COT 次数

- 任务介绍：COT Prompting 是 Prompt Engineering 的一种解决方法，将复杂问题分解为一系列小步骤。但需要降低 COT 的次数提高推理速度。
- 要求： QA 次数不能超过 3 次。

(7) 多 LoRA 数据集效果优化

- 任务介绍：集成其他智驾数据集，使用 lora 进行模型训练。
- 要求：(1) 统一数据加载格式 (2) 新场景模型训练效果与同等参数量性能保持一致。

(8) 多帧间信息关联优化

- 任务介绍： 目前 BEV 感知模型范式已经发展至时序模型，并将自动驾驶的感知问题建模成 Video 序列的全面场景感知。
- 要求：对大模型的视觉输入融合多帧间信息（2-8 frames）并对比 demo 效果。

2.3 开放性探索方向

(1) Trajectory Prediction module 的优化

为了生成 action sequence，如 future trajectories, DriveLM 的做法是单独引入轨迹预测模块来实现，如论文中给出的例子：引入 VLM:blip-2-flan-t5-xl，这为 DriveLM 的推理时间带来额外的开销。轨迹预测模块在端到端的 BEV 感知 pipeline 中是不可或缺的，但是在自动驾驶大模型中是否具有存在的必要还值得探讨，因为大模型更擅长于处理从 feature 到 action 的映射关系，对于 3D geometry 的先验和物理空间的感知是薄弱的。可以去思考轨迹预测模块在自动驾驶大模型中存在的必要性，并且如果轨迹预测模块是必要的，如何对该模块进行优化，尽量将轨迹预测任务集成到多轮 QA 当中去。

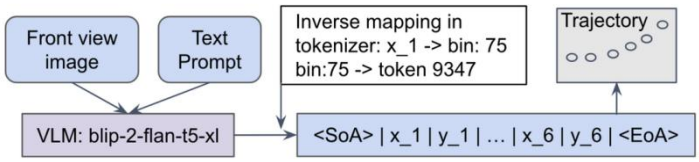


Fig. 12: Detailed architecture of DriveLM-Agent. An inverse mapping is designed to embed the real-world ego-vehicle trajectory into the token space of blip-2-flan-t5-xl.

(2) Multi-view Feature Backbone 的优化

CLIP 模型的 Vision Encoder 是视觉语言多模态大模型常用的特征提取器，然而在自动驾驶领域，无论在 BEV 感知模型或者是端到端自动驾驶模型中，使用 BEV 通用特征处理下游感知任务已经是业界通用的做法。在 DriveLM 模型中，其对 LLama_adatper_v2 模型进行微调，沿用了 llama_adatper_v2 的 clip vision transformer 的做法，现在为了增强自动驾驶大模型对 BEV 空间的感知能力，能否对 Multi-view images 的 backbone 进行优化，使用自动驾驶领域的轻量级的 BEV 感知模块替换 CLIP，并验证替换后的精度是否提升，推理时延是否具有较大的下降。

3. 项目要求

组队：4 人一组。

任务：完成八种基本优化方向中的几种，并鼓励探索给定的或其它开放性方向。

时间：现在即可开始调研项目背景和方法，后续服务器分配后约 1 个月提交作业。

提交内容：代码与文档；实验结果与分析报告。

评分标准：小组模型优化的综合性能以及实验分析报告（占比 70%）；个人工作量（占比 30%）。