Project Proposal: CS159

The Path Not Taken: Investigating LLMs for Strategic Path Planning

Introduction of research area: Large Language Models (LLMs) have shown remarkable performance in generalizing to a wide range of natural language processing problems. However, applying these models to solve real-world problems can be quite challenging. Reasoning and planning to solve any given task is a critical ability of intelligent species and is crucial in solving AI. However, current LLMs fail in proper planning and reasoning. In this project, we make a comprehensive investigation on the ability of LLMs to reason and plan to solve a variety of path-planning tasks. For the study, we design a suite of benchmarks of path-planning tasks for evaluation. Each of the tasks is designed in a grid-form environment composed of start locations, target locations, rewards, obstacles, and constraints. We permute over the components to architect multiple scenarios, e.g. single target tasks, multiple target tasks, reward maximization, etc. Moreover, we design a "chain of paths" planning task in which the model is expected to reach a sequence of target locations in a specific order while satisfying the environmental constraints. We believe each of these tasks will be an important direction in leveraging LLMs towards multiple applications ranging from robotics, to playing games, and vacation planning.

Overview of previous work: There have been several investigations into the spatial reasoning capabilities of LLMs and their ability to complete path-planning tasks. In [1], the authors propose a new dataset and benchmark for evaluating spatial-temporal reasoning via "path planning tasks" in which the LLM navigates to target locations while avoiding obstacles and satisfying a set of constraints. They formulate the spatial reasoning task as a uniform grid, with each grid element either containing an obstacle, containing a target, or left empty. They test both few-shot prompting and fine-tuning, and show success on shorter-term tasks in which the model is continuously given feedback from the environment. However, the continuously-prompted models failed on longer-term planning tasks, and the fine-tuned models struggled to generalize to out-of-distribution environments.

In [2], the authors investigate LLMs' ability to perform path planning in the context of robotics. They convert the robotic path planning problem to a natural language problem on GPT3.5-turbo to provide an optimized path for robotic systems. They present both an LLM-powered robotic system and a probabilistic transformation mechanism for converting the input robotic signal into language and then back to robotic motions, evaluating against SOTA path planners using least distance traveled and shortest planning times as path metrics. As input, their system receives user-inputted natural instructions describing the goal location in the environment. This approach assumes an "ideal path" in order to assess the quality of the optimized path outputs, and the robots are equipped with several sensors (LiDar and odometry data) to continuously determine their position in the environment. They choose GPT-3.5-turbo due to its processing speed, but find that it comes at the cost of reduced path accuracy when compared to other SOTA methods. Similar to [1], they also find that their system breaks down with longer path times and would benefit from more real-time environmental information.

Research question(s): We want to address the following questions through the project:

1. How do out-of-distribution in-context examples affect LLM's path-planning capabilities? Can LLMs generalize from in-context examples?
2. Is an LLM's uncertainty in next-token prediction somewhat calibrated in path planning? Is it correlated to the number of free paths to choose? Or the ultimate correctness of the planned path?
3. How does input modality matter for LLM path planning? Does verbal description, markdown (tokenized) description, or visual modality work better?
4. How important is chain-of-thought in path planning (chain-of-paths) when the goals are sequential?

Plan of attack:

Prompt LLMs to output a sequence of tokenized actions, e.g. ["left", "up", "right", "down"]

1. OOD Generalization of In-Context Learning
   Dataset: generate 6x6 boards with initial state, k obstacles, and m rewards.
   OOD in-context examples:
   - Smaller boards of fixed size, e.g. 3x3
   - Smaller boards of different sizes, e.g. 2x2, 3x3 (induction)
   - Same-size boards with <k obstacles and <m rewards
   - In-distribution examples with k obstacles and m rewards

   Compare {1,2,3} in-context examples (except induction) for {GPT4, GPT3.5, Llama Gemini}

2. Investigate Uncertainty Calibration of LLMs in Path Planning
   Dataset: generate 6x6 boards with initial state, k obstacles, and m rewards.
   In-context examples: 2 in-distribution (6x6, k obstacles, m rewards) examples
   Observe:
   - Uncertainty of each action vs. possible directions at that time step that the agent can take, see whether they are correlated
   - Average uncertainty of the sequence vs. overall success (success defined as collecting all rewards without touching any obstacle). Bin this (e.g. get percentage of success given uncertainty in range[0.1,0.2]) and plot.

   Compare across {GPT3.5, Llama}

3. Investigate Input Format's Effect on LLM Path Planning
   Dataset: generate 6x6 boards with initial state, k obstacles, and m rewards.
   Input formats:
    - In language description
    - In markdown, e.g. x is obstacle, o is reward
      ```
      |   |   |   | x | o |
      | x |   | o |   | o |
      |   | x | o |   | o |
      ```
    - In images (VLM)

   Using 3 in-context examples in each, compare success rate (success defined as hitting all rewards without running into any obstacles) for 3 modes. See how this changes when increasing complexity (increasing k and m, potentially also board size).
   Compare {GPT4, GPT3.5, Llama, Gemini}

4. Chain-of-Path Planning
   Dataset: generate 6x6 boards with initial state, k obstacles, and m ordered rewards.
   Compare:
    - Plain ICL with 3 examples
    - Chain of thought prompting with 3 examples
    - Zero-shot Chain of thought

   Compare the success rate (success defined as hitting all rewards without running into any obstacles) for 3 methods. See how this changes when increasing complexity (increasing k and m, potentially also board size).
   Compare {GPT4, GPT3.5, Llama, Gemini}

References:

1. "Can Large Language Models Be Good Path Planners? A Benchmark and Investigation on Spatial-Temporal Reasoning". Available at: https://openreview.net/pdf?id=fe8CzLTMG.
2. "3P-LLM: Probabilistic Path Planning Using Large Language Model for Autonomous Robot Navigation". Available at: https://arxiv.org/pdf/2403.18778.
3. "Scene-LLM: Extending Language Model for 3D Visual Understanding and Reasoning". Available at: https://arxiv.org/html/2403.11401v1.
4. "Humanoid Locomotion as Next Token Prediction". Available at: https://humanoid-next-token-prediction.github.io/.