Matheo Xenakis

8 August 2024

Project Proposal: Optimization of Multi Agent LLM Networks

Multi-agent systems of large language models (LLMs) have seen a spike in interest in recent years owing to their ability to complement some of the weaknesses of individual LLM agents and produce solutions to a wide variety of problems in different contexts (such as games, data analysis, and simple responses to user prompts). From encoding text-based environments so as to allow LLM agents to readily interpret their environment, to encoding natural-language feedback from LLM's to simulate rewards in a complex environment in the style of reinforcement learning, to even using classic reinforcement learning algorithms to optimize a network of communicating LLM agents by removing antagonistic nodes, experimenting with different multi-agent LLM architectures at both the agent and network level has led to a burgeoning field of practical research, leveraging the practicality of LLM's for a greater number of uses.

The project proposal will expand upon the current path of cutting edge research in looking for new ways to optimize multi-agent LLM networks. The end goal of the project is to publish a research paper with Prof. Constantine Dovrolis on the subject of the optimization of Multi-Agent LLM networks, possibly with a supplemental Python library. Currently, there are two possible directions for the paper under consideration:

<u>Option 1: Optimizing Multi Agent LLM Networks at the Network Level</u>

The idea of this option is to look for new ways to create dynamic, optimal networks of agents to perform well on different LLM testbeds.

Three recent papers will serve as a touchpoint for inspiration: The first paper, “Language Agents as Optimizable Graphs” (Feb 2024) uses the REINFORCE algorithm to optimize a network of multiple LLM agents for a specific task. More concretely, the communications among different agents are governed by a network so that connected nodes are able to transmit information; the REINFORCE algorithm uses a policy gradient to select the edges in the network that will maximize the network’s score on a particular task. The second paper, “DYNAMIC LLM-AGENT NETWORK: AN LLM-AGENT COLLABORATION FRAMEWORK WITH AGENT TEAM OPTIMIZATION”, introduces a framework for multi agent collaboration by an unsupervised "Agent Importance Score" to rank different LLM agents.A final paper, “Symbolic Learning Enables Self-Evolving Agents” (Jun 2024) describes a language-based loss function, which can then be used to apply back propagation across the agents in the network, similar to the way that back propagation is applied to the weights of different edges in a conventional forward/backward neural network.

In this option, our project will focus on looking for new ways to optimize a network of LLM’s. Possible innovations could involve using an online deep reinforcement learning algorithm such as Q networks to create a network that generalizes well to different tasks, or leveraging concepts from network science and to design a new network architecture that accomplishes a similar feat.

<u>Option 2: Optimizing on the Agent Level</u>

Another branch of research in the domain of multi agent LLMs involves designing the individual agent so that the resulting network produces interesting emergent behavior. The paper “Grounding Large Language Models in Interactive Environments with Online Reinforcement

Learning (ICML 23)" focuses on the creation of a system to create 'alignment' between an LLM's understanding of its environment & the true state of the environment (By default, an LLM is prone to misinterpreting its environment, which can cause problems). This is done within the agent. A second paper "Reflexion: Language Agents with Verbal Reinforcement Learning" (NeurIPS 23) innovates with a framework to reinforce language agents directly through linguistic feedback, as opposed to updating weights or shaping rewards, which is again done on the reinforcement level.

Our project approaches the problem of 'grounding' an LLM in its environment using a new approach: A neural network. There is a rich history of individual agents learning from their environment using the combination of a neural network and some form of selection. The same forces will be at play in our design. The outputs of the neural network will be available as inputs to the LLM. For example, in a mixed cooperation/competition game setting, the neural network of each agent can compute the trustworthiness of another player, which can affect the way that the agent 'talks' to the other player to coordinate behavior.

Timeline:

August - September '24: Select from these two options

September 1 - October 1: Extensive Literature review

October 1 '24- March ~15 '25: Create experiments, write code, and settle in on an initial draft of the paper

March ~15 2025 - Late April 2025: Edit paper, finalize experiments, beautify graphics

Late April 2025: Submit papers to journals