Imitation-based agents

Tom Rochette <tom.rochette@coreteks.org>

August 30, 2025 — 861fb9d0

0.1 Context

0.2 Learned in this study

0.3 Things to explore

- Desire to imitate
- Learn how to imitate/repeat, understand what are the critical points to reproduce
- RL-based interaction between the AGI and its human counterpart, where the goal of the AGI is to imitate as well as possible its counterpart

1 Overview

Humans learn from imitation. It takes us approximately 1 year to learn to talk. Before that, we learn to make noises, cry for attention, stand up to reach things and so on. Some of those behaviors are guided by imitating other human beings around us.

In the beginning, most of our training is related to motor abilities: how to pick blocks, crawl to our favorite toys, learn to make noise, to speak.

Once this first phase is completed (we've learned their basics and can improve on them), the second aspect we learn to improve is reasoning. Why does this block fit in this hole and not this other hole? Why does this happen when I do that?

If we base ourselves on the fact that machines will use imitation as their first form of learning tool, they will most likely start by integrating existing code into their software.

They'll look through the internet for code they can integrate within themselves. For instance, they'll go onto github and take every piece of code available.

The main issue at this point for them will be to figure out how to integrate this code within their own. They'll also have to figure out what are appropriate inputs/outputs.

They'll want to improve the software they've just obtained. Through static analysis they will be able to determine the exact logic that was used and potentially refactor it using more appropriate instructions and data structures. This means that AGI will become masters of program perfection.

Much like a software programmer, they'll have to be able to assess the quality of the software they integrate to themselves. One way to do that would be to have an objective metric that can determine whether or not the addition of a set of instructions improves or deteriorates the machine's expected value over an amount of simulations.

2 See also

3 References