RL and AGI

Why traditional Reinforcement Learning will probably not yield Artificial General Intelligence (S. A. Alexander)

Remy Chaput

October 16, 2020

LIRIS - SMA

Outline

Summary of the paper [1]

In RL, we (traditionally) use real numbers as rewards. They are constrained, which can prevent learning some tasks. Alternatives exist, which would remove one obstacle toward AGI.

Plan:

- The Archimedean Property
- Examples of tasks we cannot solve
- Possible solutions

The Archimedean Property (Maths!)

Archimedean Property

Let $r > 0$ be any positive real number.
For every real number y, there is some natural number n such that $nr > y$.

Generalized Archimedean Structures

A significantly-ordered structure is a collection X with an ordering $\ll$.
For $x_1, x_2 \in X$, we say x_1 is significantly less than x_2 if $x1 \ll x2$.

A significantly-ordered structure is Archimedean if:
for every X-sequence $x_0 \ll x_1 \ll x_2 \ll$ ..., for every $y \in X$, there is some i such that $y \ll x_i$.

Example

$x_0 = 1, x_1 = 2, x_2 = 4, \ldots$
I cannot find y greater than any number in the X-sequence

x_0 $\quad$ x_1 $\quad$ x_2 $\quad$ $\cdots$ $\quad$ $y?$ $\quad$ x_i

Examples of tasks with non-Archimedean structures

Known examples (maths again)

Sets

Suppose $x_i \ll x_{i+1} \iff$ "the set x_{i+1} contains all elements of set x_i".
There is a sequence $x_0, x_1, \ldots$ of sets and $y = \cup_{i=0}^{\infty} x_i$.
Thus, $x_0 \ll x_1 \ll \ldots \ll y$

Asymptotic complexities

Suppose $x_i \ll x_{i+1} \iff \Theta\left(x_i\right) < \Theta\left(x_{i+1}\right)$.
$x_0 = \Theta(n^0), x_1 = \Theta(n^1), x_2 = \Theta(n^2), \ldots$ and $y = \Theta(2^n)$

Speculative examples (not maths) i

Musical beauty

Agent's actions = one for each piano key + "stand and bow", reward is based on song's beauty.

$S_0, S_1, \ldots$ is a sequence of songs where $S_i \ll S_{i+1}$. If the rewards are real-valued, we cannot assign a reward to a "perfect" song y.

Robot surgeon

Actions = medical procedures. Assume there is a sequence of bad procedures B_i, each worse than the previous, but better than killing the patient. We cannot find y the "reward" for killing the patient such that $B_0 \ll B_1 \ll \ldots \ll y$.

Speculative examples (not maths) ii

Delayed gratification

2 buttons in a room: red $= +1$; when blue is pushed for the i-th time, infinite reward. If we approximate by a large number k, the agent might learn to push red for $k + 1$ times.

Possible solutions

Method 1: Preference-based RL [2]

Instead of giving a real-valued reward for each <State,Action>, we give preferences over the set of actions. No need for real numbers, no problem!

Method 2: Rewards with other number systems

The author describes 3 other number systems we can use instead of real numbers:

- Formal Laurent series
- Hyperreals
- Surreals

Definition

$$S = \sum_{i=-m}^{\infty} a_i \epsilon^i$$

ϵ^{-1} is a "1st-order infinite number" (bigger than every real)
ϵ^{1} is a "1st-order infinitesimal number" (smaller than every real)
ϵ^{2} is a "2nd-order infinitesimal number" (smaller than every 1st-order)

Comparison between 2 series

$A = 5\epsilon^{-1} - 2\epsilon^0 + 3\epsilon^1 + 4\epsilon^2$
$B = 5\epsilon^{-1} - 2\epsilon^0 + 1\epsilon^1 + 4\epsilon^2 + 5\epsilon^6$
$\implies A > B$

(But ... they still have limitations)

Hyperreals

- Used in the field of *non-standard analysis* (computing on infinite and infinitesimal numbers)
- We need a specific mathematical object to construct them
- The existence was proven … but we cannot exhibit one
- $\implies$ not really useful in RL

Surreals

- Every known extension of $\mathbb{R}$ which are non-Archimedean are surreals
  (including formal Laurent series and hyperreals)
- Surreals are a union of hierarchy S_a where a is an ordinal number
- We need symbolic machinery to compute these sets

Questions?

References

Samuel Allen Alexander. "The Archimedean trap: Why traditional reinforcement learning will probably not yield AGI". In: *arXiv preprint arXiv:2002.10221* (2020).

Christian Wirth et al. "A survey of preference-based reinforcement learning methods". In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 4945–4990.