Another laundry list for continual learning

Martha White
Assistant Professor
University of Alberta
in collaboration with Tom Schaul, Hado van Hasselt, Joseph Modayil, Adam White, Pierre-Luc Bacon, Jean Harb, Shibl Mourad, Marc Bellemare, Doina Precup

amii

UNIVERSITY OF ALBERTA

What is this talk about?

- A summary of some points that arose during a workshop, with many authors
    - Tom Schaul, Hado van Hasselt, Joseph Modayil, Adam White, Pierre-Luc Bacon, Jean Harb, Shibl Mourad, Marc Bellemare, Doina Precup
- Cannot represent all of them (so I won't): the statements in this talk are my own (so blame me not them)
- I will provide a bit of a summary and some concrete statements, to facilitate a discussion

What we expect in a Continual Learning setting

- Vast worlds (only a tiny fraction of states are ever visited)
- Single continuing life (cannot learn from death)
- Firehose of data (volume and velocity)
- Sparse reward signal (often zero reward)
- Non-stationarity (in dynamics and reward)
- Irreversibility

Continual Learning definition

- **Continual learning environments are vast worlds where the agent needs to predict and control its data stream**
- The name says we can never be done (learning forever)
- Other names: cumulative learning, lifelong learning

Separating the definition from the solution strategy

- Continual learning environments are vast worlds where the agent needs to predict and control its data stream
- We often include a hypothesized part of the solution as part of the definition: **accumulating** knowledge/skills/…
- **Hypothesis: bottom-up agents will be needed to learn in CL environments**
  - leverage previous learning to improve learning now
  - use learned predictions to make more predictions (composition)

We take a predictive knowledge approach to the problem

- Make many predictions about the world
- Predictive question is a question about the (cumulative) outcome into the future, conditioned on a way of behaving
- Examples of the utility of predictive questions
    - learning models can be framed as learning predictive questions
    - predictive representations of state

What might predictive questions look like?

- General Value Functions: policy contingent predictions about signals, discounted into the future
  - e.g., what is the probability of hitting the wall in the next 100 steps, if I drive forward?
  - e.g., what is the discounted sum of this feature value into the future (such as in successor features)
  - e.g., one-step transition dynamics (i.e, the model)
- Some GVFs might not be themselves useful, but could be useful to make it easier to answer other questions

Why GVFs?

- **Learnability**: We can design incremental learning algorithms for GVFs
    - often described as independent of span, unlike some other multi-step predictions into the future
- **Expressiveness**: A restricted class of predictions BUT nonetheless an expressive class of predictions about the future
    - whether this is sufficiently expressive remains to be seen
- **Useful?**

Open Issues for CL

- **Evaluating continual learning agents**
- Learning: using data effectively
- Representation: generalizing well, avoiding interference
- **Exploration**: gathering the experience needed to learn
- Discovery: choosing what to learn about and when

Open Issues for CL

➡**Evaluating continual learning agents**

- Learning: using data effectively
- Representation: generalizing well, avoiding interference
- **Exploration**: gathering the experience needed to learn
- Discovery: choosing what to learn about and when

Really want environments:

- That have rich observations but are still partially observable
- That are continuing, the agent is taking action and maximizing external reward, while constantly making many predictions
  - Not a sequence of tasks, with clear boundaries
  - Continuing environments, rather than episodic or goal-based
- That enable the agent to learn how to explore
  - e.g., might start with random exploration, but learns how should have acted to get data to learn more efficiently

Nice to have environments:

- That start simpler and progress to more difficult settings (like a curriculum)
- That have maturation constraints and pain
- That are non-stationary
  - e.g., the actuators change dynamics over time
  - e.g., have other agents that are learning

Which environments have these desired properties?

- Robots? Or at least Embodied agents?
- Can we build interesting simulation worlds?
- Fortunately, there is a workshop paper here on this topic!
  - “Environments for Lifelong Reinforcement Learning”, Khetarpal, Sodhani, Chandar and Precup

Metrics

- Typical metrics: overall reward during a lifetime or average reward per step
    - this metric should still matter! We do in fact have an external reward, and a continual learning agent should get better at maximizing it
- Additional metrics: probe/challenge questions
    - set of predictions that are hard to make
    - periodically measure competency in predicting on this challenge set

More ways to understand CL algorithms

- Evaluation will have to include some **qualitative**
    - e.g., visualizations showing visitation in an environment
    - e.g., visualization of learning quantities (predictions, stepsizes, …)
- Design toy environments that highlight particular subproblems

Really want agents:

- That can take actions quickly, but can exploit computation in the background
  - e.g., have an explicit policy that is fast to query, but might be learning a model and using it for planning in Dyna in the background
  - anytime planning
- That are sample efficient
- That can make/learn new predictions without forgetting (i.e., can accumulate not just replace)

I am not too worried about:

- Overfitting — the world is vast, and the agent limited
- Explicit transfer — taking previous policies/values to use for new situations
    - focus instead on learning a representation, to make learning faster or more sample efficient right now and later
- Agents that start from scratch
    - could initialize agents with some solutions, to simulate an earlier learning phase

Open Issues for CL

- **Evaluating continual learning agents**
- Learning: using data effectively
- Representation: generalizing well, avoiding interference

➡**Exploration**: gathering the experience needed to learn

- Discovery: choosing what to learn about and when

Exploration

- **Not an explore-exploit setting**
- Intentionally explore now **to facilitate more learning**
    - rather than to get to an optimal policy
- Learning how to explore: building exploration skills
- Avoid catastrophic events (e.g., self-damage)

Exploration drives (or intrinsic motivation)

- “Learning feels good” — care about amount of learning
  - related ideas of uncertainty reduction, compressibility, etc.
  - novelty is a different idea (novel if rarely or never seen)
- Seek empowerment states — states from which the agent can quickly/cheaply get to many other states
- Seek states where agent can influence the outcome (controllability)

Exploration mechanism

- Goal is to get the agent to take actions (gather data) to facilitate learning about many things
- **Common approach:** Rewards based on curiosity are added as reward bonuses
    - e.g., prediction error for current set of GVFs
- Curiosity and intrinsic motivation can be suitcase words; like this workshop, they are not well-defined but are worth understanding

Some comments/questions

- One of the most important parts of continual learning is what the agent should predict (discovery) — it cannot predict everything
- Many predictive questions (multi-prediction setting), but only one task (one extrinsic scalar reward)
- What should the extrinsic reward be?
- It is unlikely that we can obtain optimal continual learning agents — so what can we say about our agents?
- Curiosity as single task explorations heuristic seems to miss the point of learning for (a) the sake of learning and (b) building up knowledge about the world