The 4th Neural MMO Challenge: Open-Endedness

Joseph Suarez

October 2022

Abstract

We propose a competition on the Neural MMO platform that challenges participants to create policies capable of performing tasks not seen during training, on maps not seen during training, with allies and against opponents not seen during training. We will host separate tracks that incentivize progress in open-endedness and task-conditional reinforcement learning (TRL). This proposal builds on the experience of three previous competitions on Neural MMO with a combined total of 1120 participants and 2377 submissions.

1 Introduction and Prior Work

Neural MMO (NMMO) is a platform that simulates populations of agents in procedurally generated virtual worlds. You can learn more about the project at neuralmmo.github.io. NMMO has been in development for over 5 years, receiving six major updates since the initial release at OpenAI. The project is now developed at MIT, and all aspects are free and open-source immediately upon development with no private forks. NMMO balances complexity of the environment with computational efficiency, adapting techniques used to make classic massively multiplayer online (MMO) games able to host thousands of players in a shared virtual world on 90s hardware.

Previous Competitions include the IJCAI 2022 Neural MMO Challenge and the NeurIPS 2022 Neural MMO Challenge. We previously hosted a smaller pilot competition with AICrowd that was not affiliated with a conference. These competitions had a combined total of 1120 participants and 2377 submissions. Each of them advanced the state of the art beyond all expectations. The top policy submitted to the first competition had individual rollouts that achieved maximum score, something that the designer of the environment did not expect to be possible. In the second competition, pure reinforcement learning outperformed scripted and hybrid approaches, yielding policies capable of cooperating, specializing, and achieving multimodal objectives — all against opponents not seen during training and on maps not seen during training. Thus far, participants have quickly defeated all of the baselines we have released. It is important

to note that the version of the environment used in this competition is much more complex than any previous version of Neural MMO: rather than a toy research environment, it actually resembles a stripped-down MMO with progressive skill leveling, emergent markets and trade, and more. Several exploits that made the environment easier than intended were also patched in this version.

XLand is a high-profile DeepMind release in which the authors train on a curriculum of tasks to produce policies that generalize to new tasks not seen during training. They do so using a domain-specific language over unary and binary task predicates that produces a combinatorial number of possible tasks. There are a few problems with this work as a basis for future research in open-endedness. First and foremost, the project and environment were not made open-source. Second, the environment is slow and required DeepMind scale compute to train upon. This lack of efficiency also limited the complexity of the environment design: games are 1v1, span approximately two minutes, and are basic, intuitive physics tasks.

The Proposed Competition leverages the speed and complexity of Neural MMO to present both an accessible competition and, if successful, a significant extension of the XLand result with open-source implementations. For comparison, the NMMO competition at NeurIPS uses 128 agents on maps 16-64x larger than XLand's and time horizons of 10 minutes. Good baselines are trainable within 6 hours on a single GPU, and a planned infrastructure patch is projected to increase efficiency several fold, enabling scaling to the platform's current maximum tested scale of 1024 agents on 1024x1024 maps with horizons of over an hour.

2 Deliverables

Preparation for this competition will consist of three phases: development, baselines, and integration. Initial work can begin on baselines before the development phase is complete, but the final result of each phase is serial.

In the development phase, contributors will expand the current Neural MMO task system, improve stability and performance, and perform all pure-engineering tasks relating to the competition. This phase is estimated to require 3-4 person-months of work, not including time to spin up on Neural MMO. We aim to complete this phase in December with the following deliverables

1. Modular task system enabling the specification of single-agent, cooperative, and adversarial goals, similar to XLand
2. Generate 25K tasks for training that provide good coverage of the task space. This should be procedural such that it can be rerun with different random seeds to obtain different tasks.
3. Completion of an infrastructure patch that enables parallel computation of observations over all agents, increasing overall simulation speed several X, with a more pronounced effect in larger environments

4. If time allows, improve the efficiency of current NMMO pathfinding implementations, which are a major bottleneck

In the baselines phase, contributors will maximize performance with two separate baselines. Initial work on constructing these baselines can be performed in parallel with the development phase, but the final experiments and tuning will require the competition build of the environment. The duration of this phase is dependent upon the experience of the contributors to these baselines. We aim to complete this phase in early February with the following deliverables:

1. Port the torchbeast baseline from the NeurIPS competition to RLlib and integrate goal attention. This model should attain reasonable performance in at most 8 A100 hours. Use remaining time to increase efficiency.
2. Create the held-out task sets by rejection sampling the baseline with increasingly strict thresholds. Depending on the level of performance degradation, we may have to train a larger version of the baseline for this purpose.
3. Incrementally implement REPAIRED with a LLM diff model to generate new tasks. Start with just the diff model and prioritized level replay. The goal of this model is to generate a set of tasks better than the hand-curated set.
4. Save an offline dataset of replays generated by the baseline. This is to increase participation based on our experience from running previous competitions.

In the integration phase, the above deliverables will be handed off to Parametrix.AI, who will integrate them with the AICrowd infrastructure and perform final testing. This is projected to take 4-6 weeks, depending on the level of polish of the handoff. We aim to complete this phase in March

We have two options for the actual competition: either run the competition shortly after the above preparations are complete without conference affiliation or run a small preliminary competition and wait for IJCAI/NeurIPS next to run a larger, higher-profile competition. This proposal is agnostic to this choice.

3 Competition Structure

The competition will be split into three tracks with their own prizes. The first two track separately incentivize progress in reinforcement learning and open-endedness. The third track combines them both into a single objective. If we were to only run the first two tracks, we would gimp the final result of the competition. If we were to only run the third track, the participant pool would become much more limited, since RL researchers will not want to have to generate a task curriculum and open-endedness researchers will not want to tune RL policies. Running all three maximizes both the participant pool and the quality of the final result.

3.1 Curriculum Generation Track

Create a curriculum of 25K tasks that leads to robust policies. You will have access to an A100 for 8 hours on a machine loaded with a policy to which you can submit training tasks sequentially and a list of 25k tasks copied from the Generalization Track. Your objective is to submit a sequence of training tasks that train the policy to achieve maximum score on held-out tasks.

This track incentivizes ELM, PAIRED, and other open-endedness approaches. It explicitly fixes the reinforcement learning pipeline so no work on that portion of the training pipeline is required. This allows participants to focus only on task generation.

3.2 Generalization Track

Submit code that learns a policy over a fixed, known set of tasks. You have access to 8 A100 hours + some cores. Produce a policy that achieves maximum score on held-out tasks. Tasks are presented sequentially with no repeats.

This track incentivizes reinforcement learning approaches. It explicitly fixes the curriculum so that no work on that portion of the training pipleine is required. This allows participants to focus only on learning a robust policy.

3.3 No Holds Barred Track

Submit a policy that achieves maximum score on held-out tasks.

This track incentivizes joint work on task generation and generalization. No aspects are fixed and participants must develop all portions of the training pipeline. This allows participants to create stronger policies overall.

3.4 Round Format

Qualifier: Same as Round 1 but submissions will be evaluated using our fixed, publicly available baselines as opponents.

Round 1: Tasks will be sampled from the same distribution as training tasks. The random seed will be different and unknown to participants.

Round 2: Tasks will be sampled from the baseline policy such that there is a moderate drop in performance.

Round 4: We will rejection sample the baseline policy's worst performing tasks

Parametrix.AI will have substantial freedom to change this round structure because of their track record with orchestrating competitions in a manner that drives user engagement. The projected duration as per the above is 3-4 months, which is on par with our previous competitions.

4 Resources

4.1 Compute

1. **Development:** We will need a few thousand A100 hours for each the generalization and task generation baselines to develop and tune. This could be an underestimate for the task generation baseline since we do not have an existing implementation on the current version of NMMO.

2. **Competition:** Each evaluation in tracks 1 and 2 will take 8 A100 hours. We received 1679 submissions for the IJCAI competition and have received 537 submissions so far for the NeurIPS competition. These evaluations did not require training the submitted policy, so we implemented more lax submission constraints. The IJCAI competition was one of the most successful RL competitions ever hosted on AICrowd and did so with a complete lack of US marketing. With stricter submission criteria, some retweets from Stability and co-organizers, and good administration, our 90 percent confidence interval is 500 to 2000 submissions. Including final comprehensive evaluations at the end of the competition, this totals between 5000 and 20000 A100 hours. However, we also need to provide some compute credits to participantsin order to level the playing field (this was important in our NeurIPS competition). An easy way to do this with minimal room for exploitation is to simply increase the number of allowed submissions per participant, since evaluation for tracks 1 and 2 double as experiments. We are therefore requesting 50,000 A100 hours: 40000 to evaluate 5,000 submissions, 9000 for development of baselines, and 1000 for final experiments at the end of the competition.

3. **Prizes:** The past two competitions have had a 20k prize pool split across tracks. This has become standard for large competitions. Parametrix.AI may be able to sponsor this again. If not, we request it as part of the budget. Compared to the value of development and experiments for this competition, prize money is a very cost effective way to drive participation.

5 Expected Findings

A constant throughout all three competitions on Neural MMO has been the success of standard methods on new problems. Given quality infrastructure, reasonable observation preprocessing, and basic reward shaping, the same reinforcement learning methods have succeeded on all three competitions with few to no algorithmic changes. While some may find this trend intellectually unsatisfying, we argue that this is the best possible situation for long-term progress in AI: that we are able to solve increasingly hard problems without having to make major algorithmic innovations. It is only by pushing environment complexity to the point where current methods fails that we can discover where new approaches are truly required. Taking this into consideration, we expect:

1. Goal-conditioned versions of the same, standard RL approahces used in previous competitions will achieve substantial generalization to new tasks
2. Improvements to population based training methods will result in increased policy robustness to both new opponents and new tasks
3. Human-created scripted models will not work because of the difficulty of creating task-conditioned models a-priori
4. Simple task-selection heuristics will produce large improvements to training efficiency and generalization.
5. Task curricula to ELM will also improve training efficiency. If ELM outperforms heuristic approaches in the first competition of this kind, it will be strong evidence of long-term impact.
6. The competition will drive significant interest in task-conditional RL and open-endedness in the academic community – more so with good administration and advertising

6 Failure Case

While task-conditioned scripted models are reasonably simple in NMMO, they are still much more complex than those in e.g. sodaracer. It is possible that this is a level of complexity that ELM is not yet sufficiently advanced to achieve. We will mitigate this risk by defining macros that ELM can use in its code generation to specify high-level instructions, such as pathing to an objective or evading an opponent.

In order for the competition to succeed, the baseline must be quick to train. The largest improvement to our previous competition was the release of a baseline capable of training in 6 hours on a single GPU, vs. the previous baseline which took several days. It is unclear whether training over a curriculum of tasks will be quicker or slower than training on a single task. Intuitively, one would expect it to be harder, but other forms of randomization have substantially decreased training time in the past.

The success of the ELM component of the competition, assuming a quality baseline, will largely be dependent upon competitors' access to compute. We will either need to make ELM work with a small LM or provide API access to a LLM.

7 CarperAI Impact

This project aims to establish NMMO as a platform for open-endedness research in addition to its current popular role as an RL environment. We expect this competition to enable future research in open-endedness with a new level of

environment complexity, including many-agent interactions and detached conditional things in the form of persistent items. This also leaves room for future competitions to spur specific research on additional aspects of open-endedness. Compared to existing environments used in ELM, NMMO is both more complex and more visually interesting, both of which are likely to help popularize this line of research.

8 Broader Impacts

We anticipate no immediate and direct ethical considerations in this line of work: this research occurs within the confines of a specific game-based environment. Even with extreme success in the competition, applying the solutions developed in new, nefarious ways would be difficult and require dedicated and talented personnel.

We acknowledge that this is capabilities research and, long-term, technology that endows agents with the ability to generalize to new tasks can and almost certainly will be misused. However, this is a problem with almost all scientific progress, and nothing about this project suggests a higher-than-normal risk to progress trade-off in the immediate future.

9 Disclosure of Previous Partners

Joseph began Neural MMO independently, but the first version was published at OpenAI as part of an internship. He continued development independently thereafter until starting a PhD at MIT. The project is now maintained in Phillip Isola's group at MIT.

The first competition was partnered only with AICrowd as a joint effort to popularize many-agent environments as good targets for RL research.

The second competition was partnered with both AICrowd and Parametrix.AI, a Tsinghua affiliated startup. The impetus for this was the first competition, in which Parametrix.AI handily won first place and offered to sponsor a larger competition, both to advance the science and to attract talent in China to their company.

The third competition was also partnered with both AICrowd and Parametrix.AI. It was additionally sponsored by AWS, which provided 20k in credits to be distributed to under-resourced participants to enable them to compete. Parametrix.AI additionally focused on increasing the efficiency of the baseline model for this purpose, releasing a model capable of training from scratch on 6 hours with a single GPU mid-competition. This competition is still ongoing and will be hosted at NeurIPS in December.

We may have the option to bring some or all of these sponsors aboard for the fourth competition. AICrowd has been generous thus far in waiving platform fees, and Parametrix.AI has a wealth of experience in managing and administering these sorts of competitions, as well as the experience in RL, devops, and

user support to make them successful.