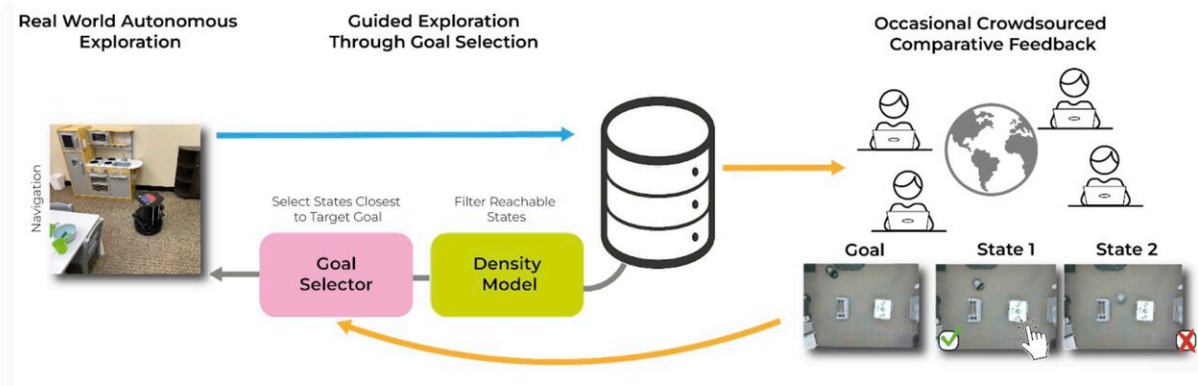# An approach that allows robots to learn in changing environments from human feedback and exploration

November 28 2023, by Ingrid Fadelli

Depiction of autonomous exploration with GEAR—the policy alternates between trying to go to a goal state and getting back to the initial state. In doing so the agent is commanded an intermediate sub-goal that is both proximal to the goal, and reachable under the current policy. When this is absent, the policy performs random exploration. The resulting policy learns to go back and forth, while efficiently exploring the space. Credit: Balsells et al.

To best assist humans in real-world settings, robots should be able to continuously acquire useful new skills in dynamic and rapidly changing environments. Currently, however, most robots can only tackle tasks that they have been previously trained on and can only acquire new capabilities after further training.

Researchers at University of Washington and Massachusetts Institute of Technology (MIT) recently introduced a new approach that allows robots to learn new skills while navigating changing environments. This approach, presented at the 7th Conference on Robot Learning (CoRL), utilizes reinforcement learning to train robots using human feedback and information gathered while exploring their surroundings.

"The idea for this paper came from another work we published recently," Max Balsells, co-author of the paper, told Tech Xplore. The current paper is available on the *arXiv* preprint server.

"In our previous study, we explored how to use crowdsourced (potentially inaccurate) human feedback gathered from hundreds of people over the world, to teach a robot how to perform certain tasks without relying on extra information, as is the case in most of the previous work in this field."

While in their previous study, Balsells and their colleagues attained promising results, the method they proposed had to be constantly reset to teach robots new skills. In other words, each time the robot tried to complete a task, its surroundings and settings would go back to how they were before the trial.

"Having to reset the scene is an obstacle if we want robots to learn any task with as little human effort as possible," Balsells said. "As part of our recent study, we thus set out to fix that issue, allowing robots to learn in a changing environment, still just from human feedback, as well as random and guided exploration."

The new method developed by Balsells and his colleagues has three key components, dubbed the policy, goal selector and the density model, each supported by a different machine-learning technique. The first model essentially tries to determine what the robot needs to do to get to a

specific location.

"The goal of the policy model is to understand which actions the robot has to take to arrive at a certain scenario from where it currently is," Marcel Torne, co-author of the paper, explained. "The way this first model learns that is by seeing how the environment changed after the robot took an action. For example, by looking at where the robot or the objects of the room are after taking some actions."

Essentially, the first model is designed to identify the actions that the robot will need to take to reach a specific target location or objective. In contrast, the second model (i.e., the goal selector) guides the robot while it is still learning, communicating the moment when it is closer to achieving a set goal.
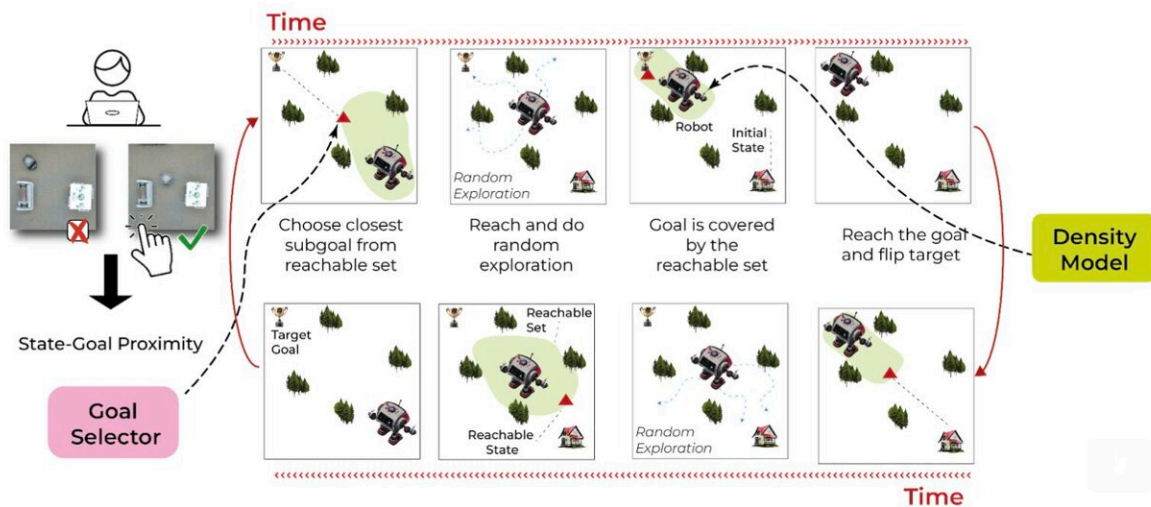
Problem setting in GEAR. The robot explores the world autonomously and reset-free only using cheap, occasional binary feedback from non-expert users to guide exploration. This allows for massive scaling of data experience and solving much more challenging tasks. The three main models of the method are: the policy used to control the robots in the real world, the goal selector, which

chooses which state is closest to the goal and the density model which filters the goals reachable by the policy. Credit: Balsells et al.

"The objective of the goal selector is to tell in which cases the robot was closer to achieving the task," Balsells said. "That way, we can use this model to guide the robot by commanding the scenarios that it has already seen, in which it was closer to achieving the task. From there, the robot can just do random actions to explore more that part of the environment. If we didn't have this model, the robot wouldn't do meaningful things, making it very hard for the first model to learn anything. This model learns that from human feedback."

The team's approach ensures that as a robot moves in its surroundings, it continuously relays scenarios it encounters to a specific website. Crowdsourced human users then browse through these scenarios and the robot's corresponding actions, letting the model know when the robot is closer to achieving a set goal.

"Finally, the goal of the third model (i.e., the density model) is to know whether the robot already knows how to get to a certain scenario from where it currently is," Balsells said. "This model is important to make sure that the second model is guiding the robot to scenarios that the robot can get to. This model is trained on data representing the progression from different scenarios to the scenarios in which the robot ended up."

The third model within the researchers' framework basically ensures that the second model only guides the robot to accessible locations that it knows how to reach. This promotes learning through exploration, while reducing the risk of incidents and errors.

"The goal selector guides the robot to make sure that it goes to interesting places," Torne said. "Notably, the policy and density models learn just by looking at what happens around, that is, how the location of the robot and the objects change as the robot interacts. On the other hand, the second model is trained using human feedback."

Notably, the new approach proposed by Balsells and his colleagues only relies on human feedback to guide the robot in its learning, rather than to specifically demonstrate how to perform tasks. It thus does not require extensive datasets containing footage of demonstrations and can promote flexible learning with fewer human efforts.

"By using the third model to know which scenarios the robot can actually get to, we don't have to reset anything, the robot can learn continuously even if some objects are no longer at the same location," Torne said. "The most important aspect of our work is that it allows anyone to teach a robot how to solve a certain task just by letting it run on its own while connecting it to the internet, so that people around the world tell it from time to time in which moments it was closer to achieving the task."

The approach introduced by this team of researchers could inform the development of more reinforcement learning-based frameworks that enable robots to improve their skills and learn in dynamic real-world environments. Balsells, Torne and their colleagues now plan to expand their method, providing the robot some 'primitives' or basic guidelines on how to perform specific skills.

"For example, right now the robot learns which motors it has to move at every time, but we could program how the robot could move to a certain point of a room, and then the [robot](#) wouldn't need to learn that; it would just need to know where to move to," Balsells and Torne added.

"Another idea that we want to explore in our next studies is the use of

big pre-trained models already trained for a bunch of robotics tasks (e.g., ChatGPT for robotics), adapting them to specific tasks in the real world using our method. This could allow anyone to easily and quickly teach robots to achieve new skills, without having to retrain them from scratch."

More information: Max Balsells et al, Autonomous Robotic Reinforcement Learning with Asynchronous Human Feedback, *arXiv* (2023). DOI: 10.48550/arxiv.2310.20608

© 2023 Science X Network

Citation: An approach that allows robots to learn in changing environments from human feedback and exploration (2023, November 28) retrieved 3 May 2024 from https://techxplore.com/news/2023-11-approach-robots-environments-human-feedback.html