Kalpa Publications in Computing

Volume 14, 2023, Pages 57–59

Proceedings of V XoveTIC Conference. XoveTIC 2022

Task-oriented autonomous representation of visual inputs to facilitate robot goal achievement

José J. Rodríguez[11], Alejandro Romero[1], and Richard J. Duro[1]
[1] University of A Coruña
jose.j.rodriguez@udc.es, alejandro.romero.montero@udc.es, richard@udc.es

Abstract

State Representation Learning (SRL) is a field in Robotics and Artificial Intelligence that studies how to encode the observations of an environment in a way that facilitates performing specific tasks. A common approach is using autoencoders and learning to reproduce the same state from a low-dimensional representation [1, 2, 3]. Although very task-independent, this method learns to encode features that may not be relevant to the task in which the encoding will be used. An alternative is to use some elements related to the goal to achieve and/or some knowledge about the environment and the problem [1] to produce an appropriate low-dimensional encoding that captures only the relevant knowledge. In this paper, we propose an approach to autonomously obtain latent spaces of the appropriate (low) dimension that permit an efficient representation of the sensorial inputs using information about the environment and the goal. To measure the performance of this methodology, we show the results of a series of simulations of robots performing a task consisting in catching a ball in different environments. In these cases, we have found that the models required for the prediction of the final position of the ball, taking as input the learned encoding, are much simpler than those that would be required using the sensing information directly.

1 Introduction

This research is part of the PILLAR [5] project. It is a project developed by the Integrated Group of Engineering (GII) of the University of A Coruña and financed by the European Union as part of the HORIZON initiative. The goal of the PILLAR project is to create a new generation of robots that are able to autonomously recognize the tasks they have to accomplish in any environment and situation.

To achieve this goal, a cognitive architecture is proposed [4]. Its approach to learning is based, among others, on Reinforcement Learning (RL) techniques. For this type of Machine Learning (ML)

[1] Corresponding author

algorithm, the developers are required to define state and action spaces, as well as rewards, that are fixed during the entire learning process. Defining these spaces is often difficult and, above all, implies a lot of intervention on the part of the designer, hindering the open-ended learning capabilities autonomous robots should display..

This work is a first approximation to an autonomous state representation system that is able to learn efficient representations of the state for the current task directly from the primary sensorial inputs of the robot. We propose an approach based on a Neural Network architecture and an Optimization algorithm that, together, find representations of the states that are both low-dimensional and informative enough to achieve the goal of the task. Low-dimensional representations are desired since they remove most of the noise from the inputs and facilitate further processing.

To validate the proposal, several simulations are executed. In those simulations, a robot has to catch a ball in different environments. The representations found in all cases present a dimension that is less than the one we expected, taking as a reference Newtonian physics, but that representation allowed the robot to perform the task with an acceptable error taking into account the size of the ball and the environment dimensions.

2 Details of the proposal, results and further work

The proposed Neural Network architecture consists in an encoder that produces the state representation and a decoder that is specific for each task. As the task, we are experimenting with is to catch a ball, the decoder is a single-layer network that predicts the coordinates of the ball from the representation of the state in three previous instants of time. The Efficient Net pre-trained model [6] is used for the encoder, but a last dense layer is added to produce the final representation. The inputs of the network are three video frames, thus three copies of the encoder are used and their outputs are concatenated before producing the final representation.

We made the assumption that the error when performing the task will monotonically decrease as the dimension of the representation increases. That's due to the fact that a larger representation contains more information from the input than a smaller one. So, a bisection algorithm was implemented to find the smallest dimension that allows the robot to perform the task with tolerable levels of error. The neural network is trained as part of the optimization algorithm. On each iteration, the performance of the network is measured and if it is acceptable, then we try with a smaller dimension (i.e. fewer neurons in the last layer of the encoder) in the next iteration, otherwise, we try with a larger dimension.

The entire solution is implemented in Python[8, python]. The neural network is built with Tensorflow [8]. As the training is very resource-consuming we used the Kaggle [9] platform that offers free but not unlimited GPU access [10]. For simulation, we used Pybullet[11], a simulation engine that can be used as a Python library. Pybullet is used for both, training dataset generation and final validation.

Although the goal is always to catch a ball, a diversity of environments was used to test the proposal. On the first evaluation there was just the ball in an empty environment, then some objects were added, after that, we experimented with changing the colors of the environment and, in the last experiment, we added a second ball with a different color that the robot should ignore. In all cases, the dimension of the representation the algorithm obtained was smaller than the one we would need to make the prediction from the equations that describe the motions of projectiles and obviously much smaller than the dimension of the direct observation space which was 128x128 pixels. The error is calculated as the euclidean distance from the prediction to the actual final position of the ball. In all cases, this error was less than 3 units. Those units are the ones defined in the simulation engine. The diameter of the ball is 1u and the area of the surface where the ball can potentially fall is $400u^2$. The distribution of errors is

shown in figure **2.1.** Another important result is that there is no statistical evidence of dependence between the representation and the error or the prediction.

The goal of this work was to make a first approximation to a task-oriented autonomous state representation system for the new generation of robots that the PILLAR project aims to build. The results show that such a system seems possible to implement. The methodology proposed in this work found a state representation that is both lower-dimensional and informative enough to achieve good performance in the task.

Further work should take two directions. One of them is the application of the methodology to real robots and the second is to explore the generalization power of this methodology. The last one can be approached by experimenting with new tasks and environments.

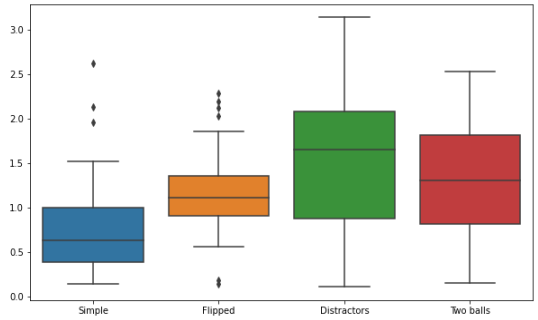

Figure 2.1: Distribution of the errors for every simulated task.

References

[1] Lesort, T. et.al. (2018). State representation learning for control: An overview. Neural Networks, 108, 379-392.

[2] Ahishali, M. et.al (2022). SRL-SOA: Self-Representation Learning with Sparse 1D-Operational Autoencoder for Hyperspectral Image Band Selection. arXiv preprint arXiv:2202.09918.

[3] Caselles-Dupré et.al. (2018). Continual state representation learning for reinforcement learning using generative replay. arXiv preprint arXiv:1810.03880.

[4] Becerra, J. A. et. al. (2020). Motivational engine and long-term memory coupling within a cognitive architecture for lifelong open-ended learning. Elsevier.

[5] Duro, R. et.al. (2020). PILLAR-RobotsPurposeful Intrinsically motivated Lifelong Learning Autonomous Robots, HORIZON-CL4-2021-DIGITAL-EMERGING-01.

[6] TAN, M., Y LE, Q.; Efficientnet: Rethinking model scaling for convolutional neural networks, In International conference on machine learning, (2019).

[7] RASCHKA, S. et .al.; Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence., Information, (2020).

[8] ABADI, M.; TensorFlow: learning functions at scale., In Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming, (2016).

[9] GOOGLE LLC.; 2010, Kaggle. Environment specifications, https: // www. kaggle. com/ docs/notebooks

[10] UPADHYAYA, S. R.; Parallel approaches to machine learning—A comprehensive survey., Distributed Computing, (2013)

[11] COUMANS, E., BAI, Y., Y HSU, J.; Pybullet. Available on: https: // pypi. org/ project/ pybullet/ , Retrieved, (2019)