Universidade: presente!

UFRGS PROPESQ

XXXI SIC

Salão UFRGS 2019 – Conhecimento o Formação o Inovação

21 . 25 . OUTUBRO . CAMPUS DO VALE

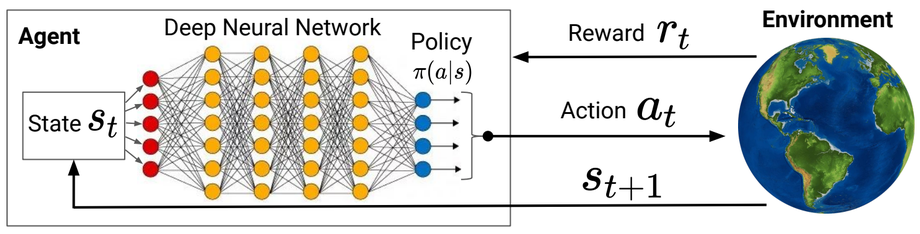
Multiagent Deep Reinforcement Learning Context Detection

Lucas N. Alegre - Orientadora: Ana L. C. Bazzan
{lnalegre, bazzan}@inf.ufrgs.br

INSTITUTO DE INFORMÁTICA UFRGS

INTRODUÇÃO

Deep reinforcement learning (DRL) é um campo da inteligência artificial que tem atingido performance sobre-humana em diversas áreas, desde tarefas de robótica a jogos complexos como *Go* e *Dota*. Combina **aprendizado por reforço**, onde um agente deve aprender uma política (mapeamento de estados para ações) que maximiza um sinal (recompensa) através de sucessivas interações com um ambiente dinâmico, com o poder de abstração das **redes neurais profundas**.



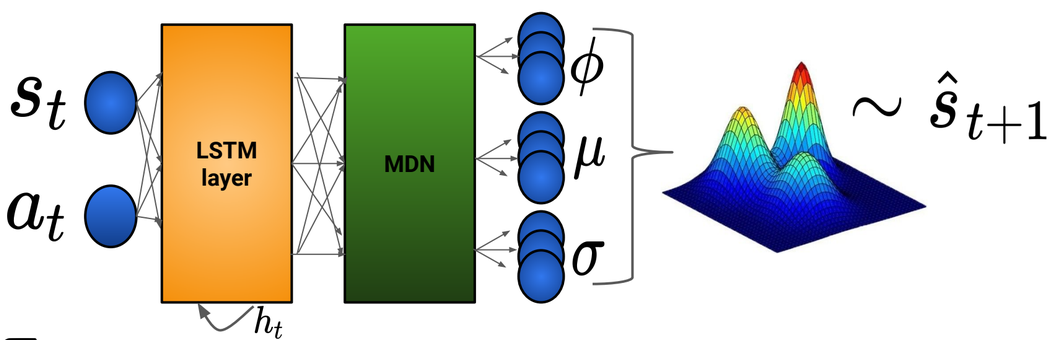
OBJETIVOS: Propor método capaz de agir em ambientes multiagentes com **estados contínuos** e **não-estacionários**, nos quais a dinâmica do ambiente (contexto) muda de forma imprevisível. Os agentes devem ser capazes de **aprender diferentes políticas para diferentes contextos e de detectar o contexto corrente**.

MÉTODOS

O **Multiagent Deep Reinforcement Learning Context Detection (MDRL-CD)** consiste para cada contexto i de:

- M_i : *Mixture Density Recurrent Neural Network (MD-RNN)*, que modela a distribuição probabilística do próximo estado como uma mistura de m Gaussianas multivariadas:

$$P(s_{t+1}|s_t, a_t, h_t) = \sum_{i=0}^{m} \phi_i g_i(s_{t+1}|\mu_i, \sigma_i^2)$$



- π_i : Política treinada com qualquer algoritmo de *RL model-free*. Ex: *Proximal Policy Optimization (PPO)*.

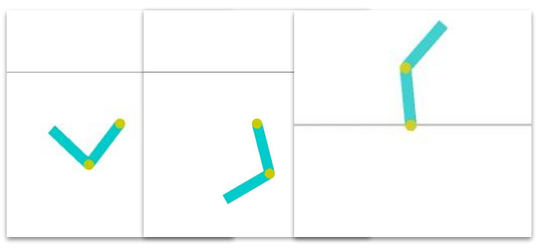
A cada instante, detecta-se se houve uma possível mudança do contexto 0 para o contexto 1 através do algoritmo *CUSUM*:

$$W_n = max(0, log(\frac{g_1(s_{t+1}|s_t,a_t)}{g_0(s_{t+1}|s_t,a_t)})),\ W_0 = 0$$

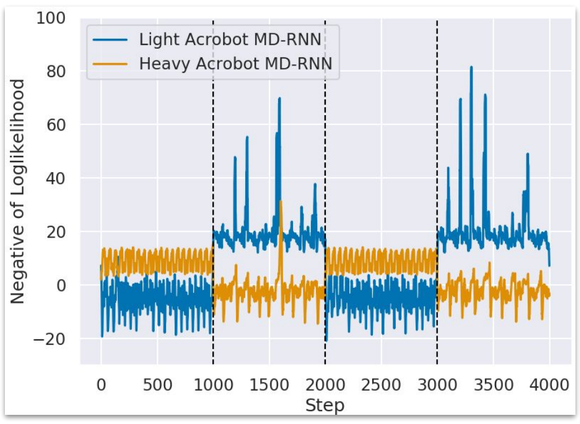
Onde W_n maior ou igual a um limiar A indica a ocorrência da mudança de contexto no passo n.

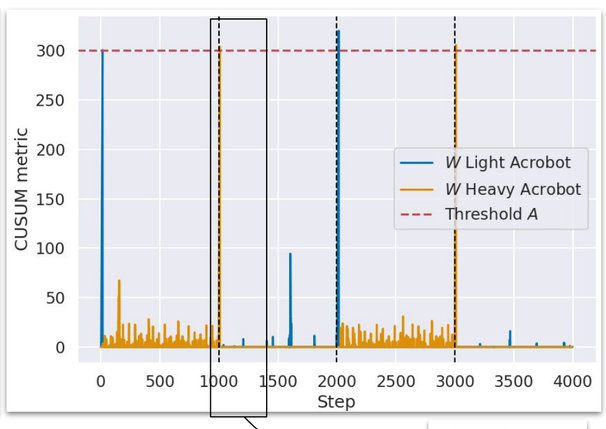
EXPERIMENTOS E RESULTADOS

- Extensões dos environments *Acrobot (light e heavy)* e *Hopper (low and high gravity)* do *OpenAI Gym* foram usados como *benchmark*.



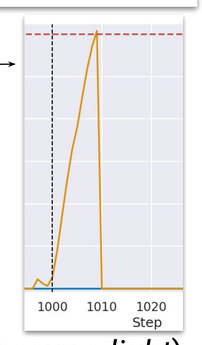
- Quanto menor o negativo do *log-likelihood* previsto pela *MD-RNN*, mais provável que o agente esteja nesse contexto.

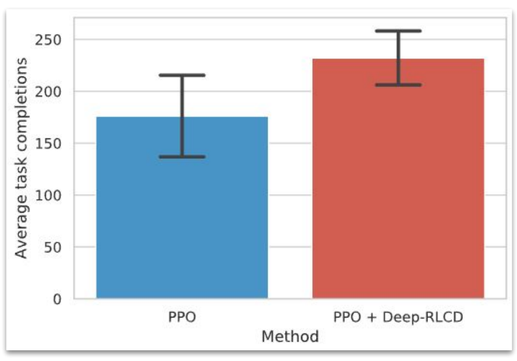




*Linhas pontilhadas verticais indicam o momento da mudança de contexto.

- Ao detectar uma troca de contexto pela *MD-RNN* M_i, o *Deep-RLCD* alterna para a política π_i. Evitando assim o uso de uma política sub-ótima.





- Número médio de vezes em que a tarefa foi completa no *Hopper* alternando-se a massa (*heavy* e *light*) a cada 1000 passos durante 20000 passos em 10 execuções.
- O nosso método evita o **esquecimento catastrófico** da política ótima de um contexto anteriormente visto pelo agente.

CONCLUSÕES

- O método desenvolvido foi capaz de **detectar mudanças de contextos em um pequeno número de passos (~10)**, rapidamente alternando para a política adequada e evitando quedas drásticas de performance.
- É necessário o desenvolvimento de uma métrica de confiança que diferencie erros de previsão causados por falta de dados e por mudanças de contextos, permitindo o treinamento da política e do modelo de transição de forma *online*.
- Como próximos passos, iremos realizar experimentos em ambientes **multiagentes**, nos quais os agentes podem compartilhar experiências para treinamento das *MD-RNNs*.

Referências: [1] Sutton, R.S. and Barto, A.G., 1998. Introduction to Reinforcement Learning. MIT Press. [2] Bruno Castro da Silva, Eduardo W. Basso, Ana L. C. Bazzan, and Paulo M. Engel. 2006. Dealing with Non-Stationary Environments using Context Detection. In Proceedings of the 23rd International Conference on Machine Learning ICML. [3] D. Ha and J. Schmidhuber, 2018. Recurrent World Models Facilitate Policy Evolution. In Advances in Neural Information Processing Systems 31. [4] K. O. Ellefsen, C. P. Martin, and J. Tørresen, 2019. How do Mixture Density RNNs Predict the Future? CoRRabs/1901.07859