Prediction of Pushing Affordances using Recurrent Neural Networks

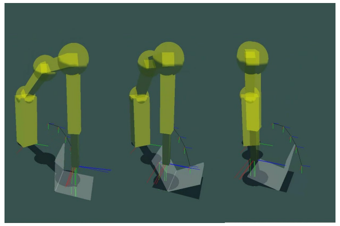
Sergio Roa and Geert-Jan Kruijff

Talking Robots@Language Technology Lab, German Research Center for AI, Saarbrücken, Germany
CogX EU project

Introduction

- Task: Prediction of pushing affordances with polyflaps: predict a sequence of polyflap poses given a pushing motor command of a Katana6M™ simulated arm.
- Approach: Offline and Active Learning using Recurrent Neural Networks, specifically *Long Short-Term Memory (LSTM)*.
- During the pushing movement, a sequence of rigid body poses are stored and used for learning. The LSTM then learns a regression function

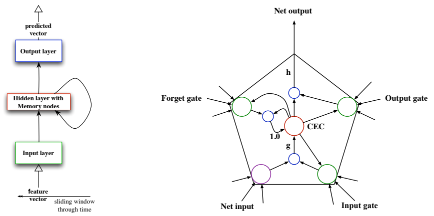
Scenario



Learning approach

- We first conducted offline experiments to test the generalization capability of the learners.
- Then, we used an active learning algorithm that selects an action to perform according to a learning progress measure maximization (c.f. Oudeyer et al.)
- We put the polyflap in a certain position and we define 18 different starting positions for the arm to start a pushing movement (i.e. 18 different sensorimotor regions).

Learning approach (LSTM)



Learning algorithms

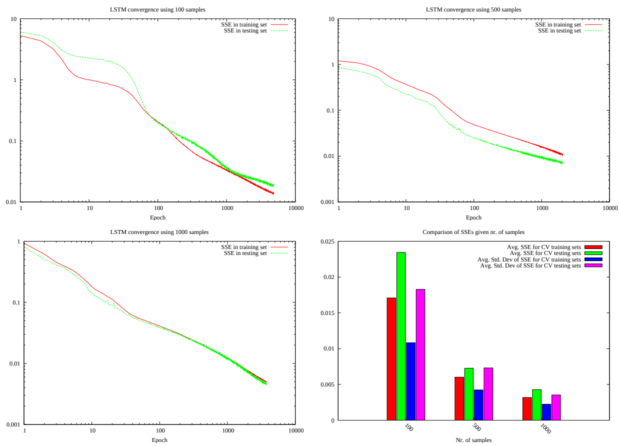
Long Short-Term Memory

Recurrent neural network training algorithm that allows prediction of long sequences by using a gradient-descent approach.

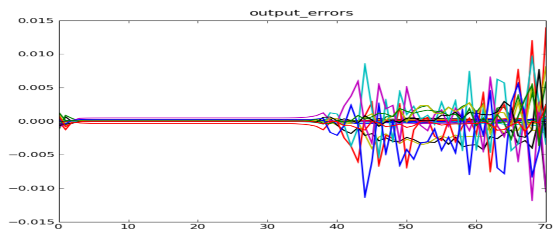
Interesting features:

- Constant error carrousel neurons: learn to get rid of non-relevant inputs or outputs by learning to close and open input and output gates.
- Forget gates: learn when previous inputs need to be forgotten.
- Peephole weights: improve the learner ability to predict exact timing during sequence processing.

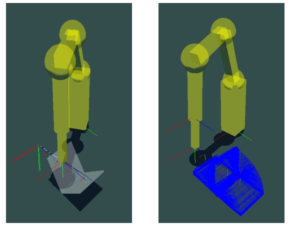
Experimental Results for Offline Learning



Output neuron errors for a sequence



Experimental results (predicted sequence)



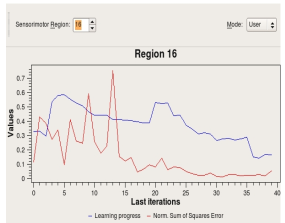
Active Learning (Intrinsic Motivation)

- The learning progress $L(t)$ can be calculated as the difference between the mean error values $e(t-\theta)$ and $e(t)$.
- By using a ϵ-greedy policy, given $L(t)$, select an action a from the set of possible actions $\{a_i\}$ in 18 sensorimotor regions $\{\mathcal{R}_i\}$ so that:

$$a = \arg\max_{a_i \in R_i} \{L_i(t)\}$$

- Update the weights of the LSTM.

Preliminary results (active learning)



Conclusions

- LSTM convergence is shown by offline experiments.
- LSTM performs well for sequences processing, at least when data do not include noise.
- Straightforward implementation of active learning techniques like intrinsic motivation systems.
- A comparison of convergence between offline and active learning approaches might be carried out.
- Alternative offline training algorithms might be considered.