Faculty of Engineering
Albert-Ludwigs-Universitat
Autonome Intelligente Systeme - AIS
Wintersemester 2017/18

Deep Learning Lab Course

2017

Assignment 3

Author: Lior Fuks (4251285)
Markus Merklinger (4211461)
E-mail: liorfuks@gmail.com
merklingermarkus@gmail.com
Tutors: Aaron Klein,
Artemij Amiranashvili,
Mohammadreza Zolfaghari,
Maria Hgle,
Jingwhei Zhang,
Andreas Eitel
Data of submission: December 18, 2017

Deep Learning Lab Course 2017
Assignment 3

December 18, 2017

Abstract

In this Assignment we got familiar with the python Tensorflow framework. This was achieved by implementing a small convolutional neural network (CNN) and training an agent to preform a planner task. Furthermore the impact of different history length and agent local view on the performance were examined and evaluated. In addition we tested the cases where the map or goal position were changed.

Contents

1 Introduction **2**

2 Implementing and training an agent using CNN **2**

3 Results **2**

3.1 Testing different history lengths 2
3.2 Testing different local view sizes 3
3.3 Changing target after training 4
3.4 Changing map after training 4
3.5 Training the agent on a map with a random target location 5
3.6 Changing map after training 5

4 Discussion **5**

1 Introduction

In this exercise we used a convolution neural network to perform a planner task in a grid world based only on his local view. The training improves the agent ability to map the current local view to a movement action towards the goal.The network input data is created by using an A* algorithm given the entire map and collecting the local view images and corresponding action of the agent at each state. Each input data of the network is composed of several states representing the history of the agent. Using the current state and history the agent computes its next action.

2 Implementing and training an agent using CNN

The first task was to implement and train an agent using a CNN. We created a network with 5 layers followed by a dropout and a soft max classification output layer. We used a batch size of 32 with 500 batches and 5 epochs. The network structure is described in the following Table 1.

The implementation can be found under https://github.com/RedHeadM/dl_lab_2017.

Table 1: CNN layer structure

Convolutional Layer number	No. of filters	Kernel size	Stride	Activation function
1	32	3x3	2	Relu
2	64	3x3	1	Relu
3	128	3x3	1	Relu
4	128	3x3	1	Relu
Fully connected layer	No. of units	Drop out		
5	1000	0.5	-	Relu
Classification layer	Action classification 5			Softmax

3 Results

3.1 Testing different history lengths

The agent is trained and tested with different history lengths and a fixed map. The agent starting position is randomly chosen and its goal is to reach the goal given only its local view. The success rate, which represents how often the agent have reached the goal, as well as the mean difference between the number of steps using A* and the CNN is plotted in the following graph. Note that for the mean difference to the A* algorithm only runs where the agent reached the goal are taken into account.

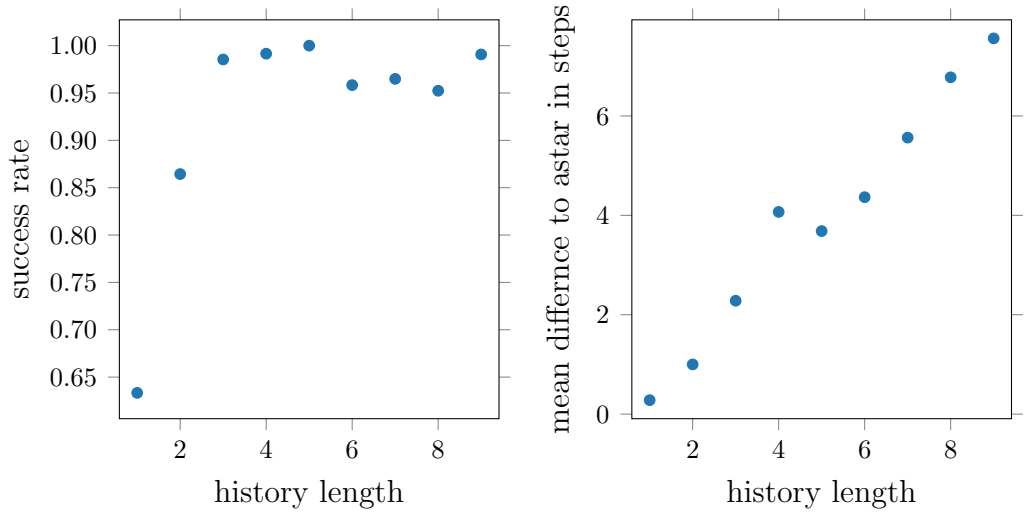

Figure 1: Success rate and mean difference to A* for a changing history length in a fixed map with a fixed goal

3.2 Testing different local view sizes

The agent is trained and tested with different local view sizes and the success rate as well as the difference between the number of steps using A* and the CNN is plotted in the following graph.

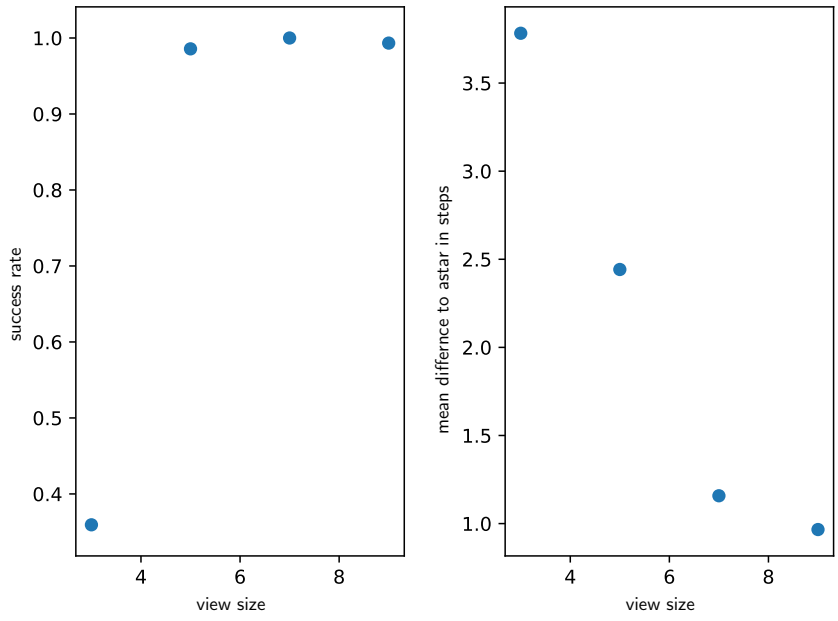

Figure 2: Success rate and mean difference to A* for a changing view size in a fixed map with a fixed goal

3.3 Changing target after training

When randomly changing the goal position in the same training map, the agent goes near to the original goal position as he did before. However, when the original goal position is visible in the local view the agent starts to loop, going back and forth near the goal.

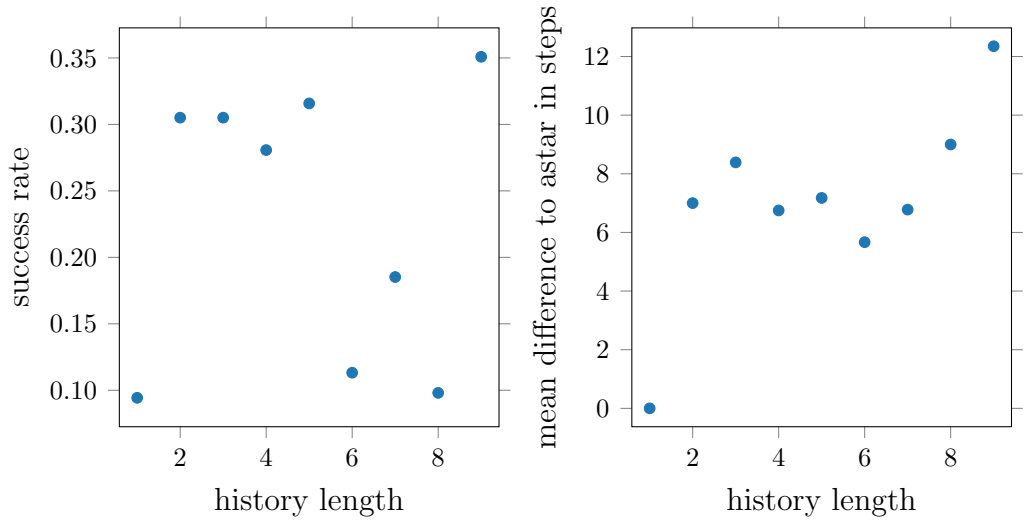

Figure 3: Success rate and mean difference to A* for a changing history length in a fixed map with a randomly changing goal

3.4 Changing map after training

The following graph shows the success rate as well as the difference in the number of steps between A* and CNN for training on map 0 but testing with map 1.

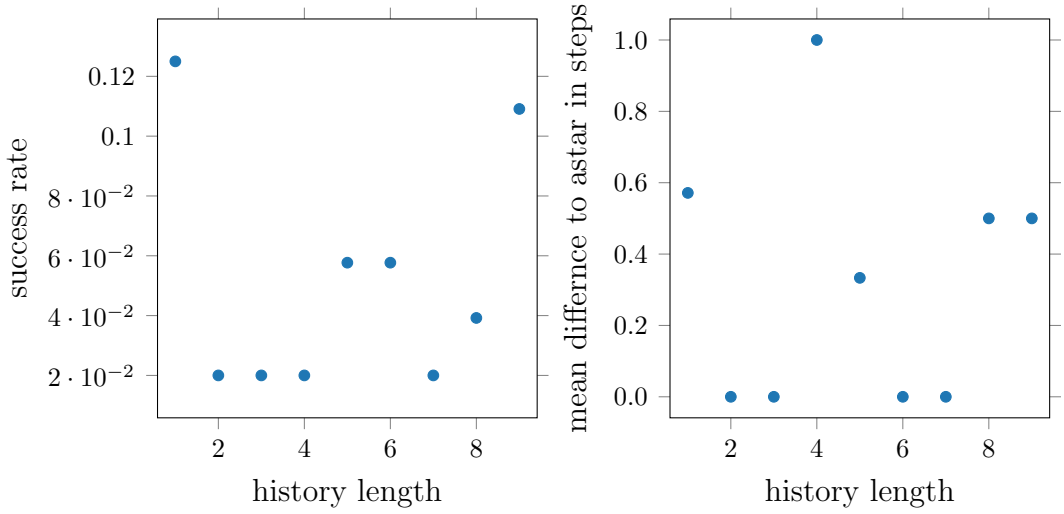

Figure 4: Success rate and mean difference to A* for a changing history length in a different map with a fixed goal

3.5 Training the agent on a map with a random target location

In order to improve the success rate of the agent for the case where it has a random goal point in a fixed map. We tried to train it on a random goal as well. The success rate as well as the difference between the number of steps using A* and the CNN is plotted in the following graph.

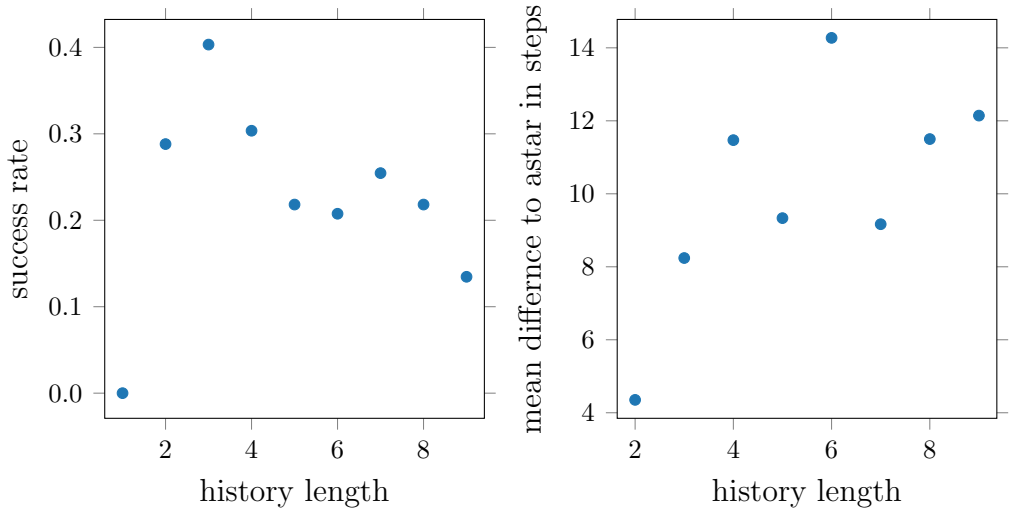

Figure 5: Success rate and mean difference to A* for a changing history length in a fixed map with a randomly changing goal both in training and testing

3.6 Changing map after training

4 Discussion

As one can see from the results above at Figure 1, increasing the history length improves the success rate of the agent but decreases the efficiency of the solution, which is described as the mean difference in steps between A* and CNN. In addition, as seen in Figure 2, increasing the agent local view size improves both the success rate and the efficiency of the agent. When testing the trained agent with a different map or goal position, the agent achieves a much lower success rate as seen in Figure 3 and Figure 5. When changing the goal in the training, one can see that the CNN performs worse in comparison to the fixed goal case, but it still shows some indications to generalize. This implicates that we over fit the training to the map and goal position in the training data. For the case where we change the map after training, the CNN almost shows no generality. Our proposal for improvement in order to overcome the over fitting to a goal position is to use a random goal in the training data. However this method did not increase the success rate.

List of Figures

1 Success rate and mean difference to A* for a changing history length in a fixed map with a fixed goal . . . 3
2 Success rate and mean difference to A* for a changing view size in a fixed map with a fixed goal . . . 3
3 Success rate and mean difference to A* for a changing history length in a fixed map with a randomly changing goal . . . 4
4 Success rate and mean difference to A* for a changing history length in a different map with a fixed goal . . . 4
5 Success rate and mean difference to A* for a changing history length in a fixed map with a randomly changing goal both in training and testing . . . 5

List of Tables

1 CNN layer structure . . . 2