1st Report of the project “ns-3 network simulator running Machine Learning Algorithms in python”

Participants:
Gabriel Aldair Villagrán Saucedo
UASLP
267572@alumnos.uaslp.mx

Jose Saul Castillo Ipiña
UASLP
292461@alumnos.uaslp.mx

Advisor
Dr. Pedro David Arjona Villicaña
UASLP
david.arjona@uaslp.mx

Teacher
Dr. Juan Carlos Cuevas Tello
UASLP
cuevas@uaslp.mx

Abstract
Artificial Intelligence is a vast world that involves different algorithms that makes easier the way that we do things, In this project we are going to use a module of ns-3 simulator called **ns3-ai,** which allows to implement AI algorithms in the network like Machine Learning (ML), Reinforced Learning (RL) and Deep Learning (DL) that are implemented in an easy way and in combination with ns-3 simulator that simplifies the works of developing protocols such as TCP, IP, UDP, etc. and communication technologies such as Ethernet, WiFi, LTE, WiMAX, etc.

Keywords
ns-3, AI, Network Simulation

1 Introduction

This project is being worked on by Jose Saul Castillo Ipiña and Gabriel Aldair Villagrán Saucedo for the Machine Learning course, under Dr. Juan Carlos Cuevas Tello and Dr. Pedro David Arjona Villicaña as advisor, our main goal is to understand how the simulator works and how we can implement the AI algorithms using the ns3-ai module.
Ns-3 is an open source simulator which is very helpful for research and educational use in networking. Inside of ns-3 we can use a lot of modules such as ndnSIM, NetSimulyzer, ns3-gym, etc. For this project we are going to use the module ns3-ai that will help us to use AI algorithms in an easier way.

2 NS3 and NS3-AI

ns3-ai is a module developed for the ns-3 simulator. It is an open source networking simulation tool implemented in C++ and python and is highly used for network research and education and in combination with ns3-ai provides a high-efficiency solution to enable the data interaction between ns-3 and other python based AI frameworks.
Ns3 can be installed in different operating systems and it has different ways to do it, in this case I'm going to use **Ubuntu 20.04** and the detailed installation guide step by step that I use to install ns-3 and ns3-ai can be found here GabrielVillagran/ns3-ai: Installation guide to install ns3-ai (github.com).

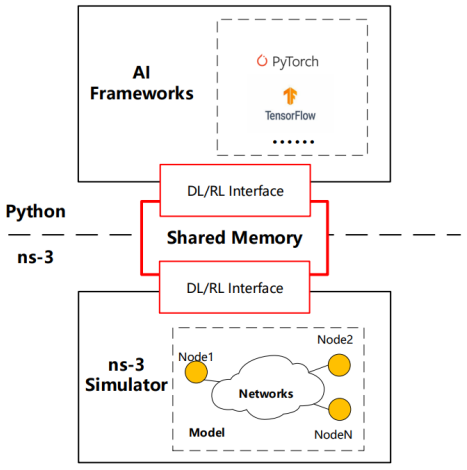

Fig.1

Figure 1 shows us a brief diagram of how ns-3 and ns3-ai works together.
ns3-ai is a share module between open source AI frameworks and ns-3 that helps us to access all the data to interact with AI models in real-time so we can quickly reproduce and verify the algorithm, provide us a high-level interface in Python and C++ for different algorithms, which makes it easier to adapt to different requirements
and present different examples to implement and test AI algorithms in network simulation.

3 How it works

The simulator and AI frameworks run in different processes and the data transfer is required in two situations:

1. Sending data for training and testing the AI model in ns-3.
2. The ns-3 simulator is used to set up the network and topology, which generates data for the training in AI algorithms. The AI frameworks provide functions to train the models using the data from the ns-3 and then to return the data from the trained model back to ns-3 for testing.

The simulator provides us with environments for testing AI algorithms on most of all network layers by building simulation scenarios.

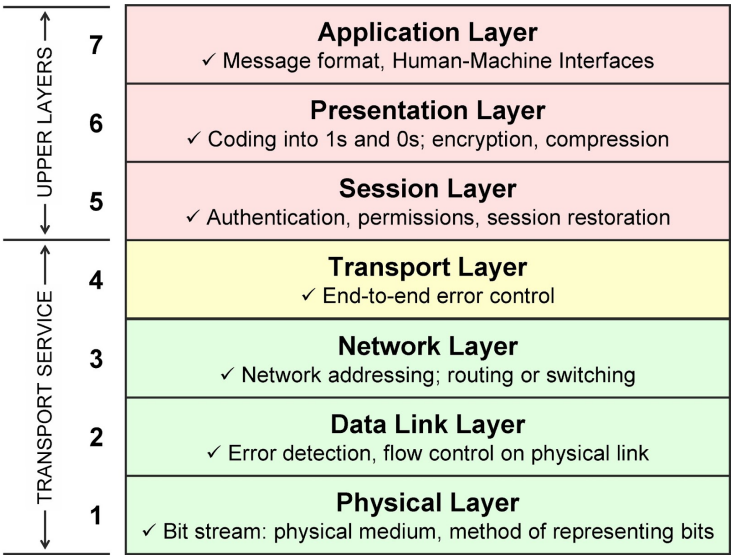

Fig.2

Figure 2 give us a brief information about the network layers based on OSI model, you can find more information about the OSI model (Network layers) on this link [OSI Model Reference Chart (cisco.com)]

The ns-3 simulator could also serve as the data generator. For instance:

"Thinking of a simple cellar network, users may be connected or disconnected to the base station or move from one side to another. We concern about the location and speed of the user in a higher view but also desire to acknowledge its channel condition and user experiences. Then we could train a model to make the scheduling decision based on this information. In ns-3, it is straightforward to build a scenario from the user view and also receive the lower layer information to feed the deep learning model. Thus ns-3 could be regarded as a data generation tool as well as a test tool for various requirements.

Different AI frameworks using Python scripts are all supported to be integrated with this module. By applying the interfaces in the Python side, the users could extract the data from the shared memory then continue to train the model or return the results for testing. It only impacts the procedure of data processing and testing methods without interfering with the internal processing of the AI algorithms.

Important: The ns3-ai module interconnects the ns-3 and AI frameworks by transferring data through the shared memory pool. The memory can be accessed by both sides and controlled mainly in ns-3. From data using for training or testing to the control signals, all the information transmitting through the two parts could be handled by the ns-3 module. In this way, users could define different memories for complex usages, as all kinds of messages could convey through the ns3-ai module"

Yin, Hao and Liu, Pengyu and Liu, Keshu and Cao, Liu and Zhang, Lytianyang and Gao, Yayu and Hei, Xiaojun.
Ns3-Ai: Fostering Artificial Intelligence Algorithms for Networking Research 2020

3.1 Shared memory pool

The shared memory pool (fig. 3) is an important part of the simulator because the data is transferred into separate memory blocks at the beginning of each simulation and it's divided in three blocks, so different kinds of data could use different memory blocks.

1-. Main Control Block: Is in charge of the memory and updates its version after each modification.

2-. Control and memory blocks: This are one to one correspondence each time when you acquire a new memory block from the head, a new control block corresponding to it will be created automatically in the tail of the memory pool.

Control block includes the next information:

- Size and the address of the memory block.

It is readable but could not be changed until the free of the block while the memory block is readable and writable memory.

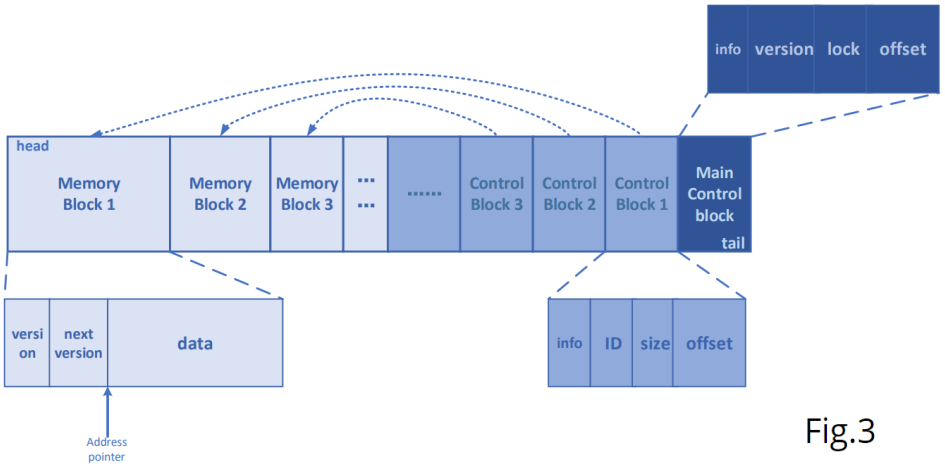

Fig.3

4 Validating the installation of ns-3 and the ns3-ai module

ns-3 has unit tests that can be run to verify the installation to run this tests you have to be on the ns-3.35 directory (fig. 4) :

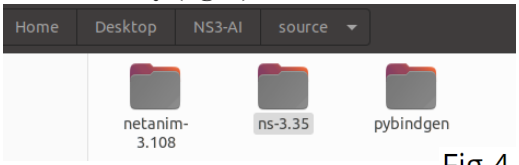

Fig.4

```
./waf configure --enable-tests
./test.py
```

```
PASS: TestSuite histogram
PASS: TestSuite ns3-wifi-interference
PASS: TestSuite ns3-tcp-cwnd
PASS: TestSuite ns3-tcp-interoperability
PASS: TestSuite sample
```

for using the different examples that are already created you need to be in the directory "**ns-3.35**"

```
./waf -d optimized configure; ./waf
```

NOTE: You can choose the directory that you want to try, on figure 5 you can see the different examples that you can run

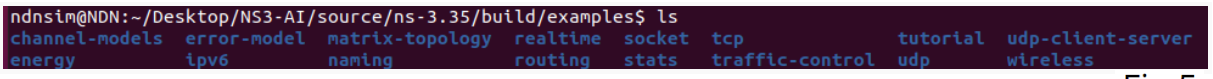

Fig.5

In this example I am using the **tutorial** directory that shows us the output of figure 6.

```
./waf shell
cd build/examples/tutorial
./<name of the executable> (e.g.):
./ns3.35-hello-simulator-debug
```

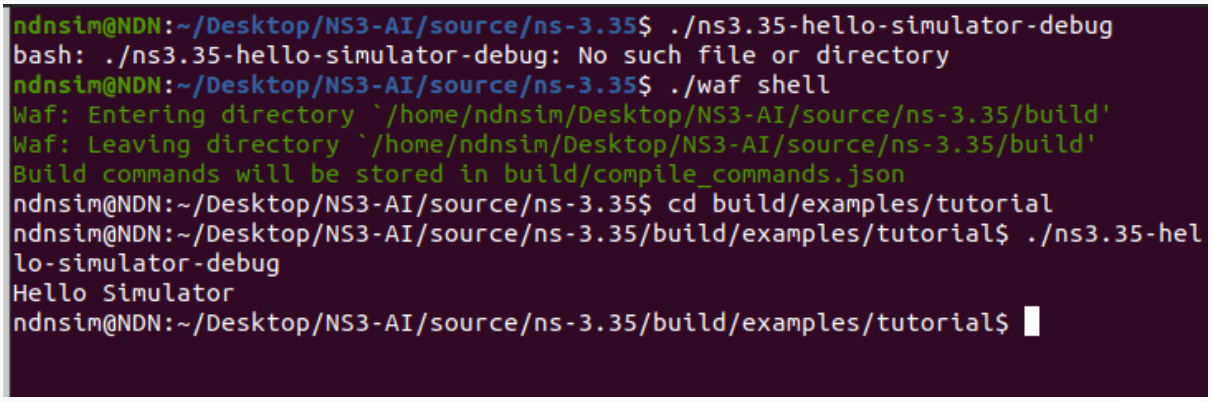

Fig.6

4.1 Running ns3-ai examples

ns3-ai provide us examples like a_plus_b, RL-TCP, LTE_CQI and Rate-Control I'm going to use the first one which is a very simple case that transfer data from ns-3 to python side and calculate a + b in the python to put back the results:
The output of the examples can be seen on figure 7.

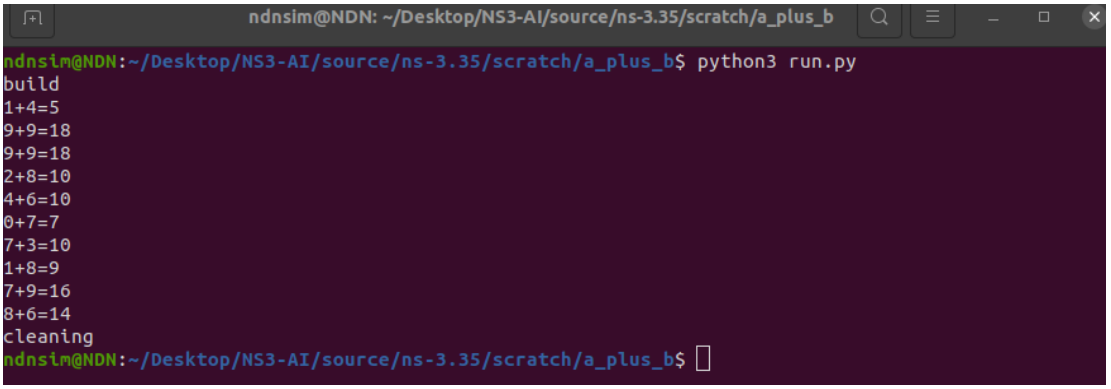

Fig.7

5 Algorithms

In this project we are going to use Deep Learning (DL) and Reinforcement Learning (RL) algorithms for developing our simulations and examples.

Conclusions

This is a very interesting project because internet networking is a fundamental part of our everyday life and combining it with deep and reinforcement learning will be very useful to develop more efficient routes to our protocols and one of the most important characteristics of this project is that we can work in all network layers which is very interesting for research and study.

Bibliography

(1) (PDF) ns3-ai: Fostering Artificial Intelligence Algorithms for Networking Research (researchgate.net)

Communications-Caching-Computing Tradeoff Analysis for Bidirectional Data Computation in Mobile Edge Networks (nsnam.org)

hust-diangroup/ns3-ai: Enable the interaction between ns-3 and popular frameworks using Python, which mean you can train and test your AI algorithms in ns-3 without changing any frameworks you are using now! (github.com)