Supervised Learning approach to Detect Anomalies in Blockchain using Federated Learning

Problem Statement:

As the interest in and use of machine learning for security applications increases, so will the awareness of cyber-criminals. When frequently updating an ML model to account for new threats, malicious adversaries can launch causative/data poisoning attacks to inject misleading training data intentionally so that an ML model becomes ineffective. Detecting anomalies in Blockchain needs each individual block data has to take to the central server which is very complex to get and train each block. And also in the testing phase, the model needs new block data it was complex to get the test data. As mentioned above attackers are intentionally deliver their data to the model which can't predict anomalies.

Proposed Solution:

So, I came up with a new solution using new concept called Federated Learning (A technique for training Machine models on data which do not have access). Federated learning is one of the most widely deployed techniques in the context of Private Deep Learning. An interesting blog about Federated Learning by Prof. Mi Zhang found here.

I had created 50 VirtualWorkers for training the model using PySyft (PySyft is a Python library for secure, private Deep Learning). I had added data to each VirtualWorker and then created Blockchain using Awesome Blockchains.

After creating the model the created model the created model has been sent to each VirtualWorker containing the data and train the model and goes to another Model. This process has iterated to all the blocks in the Blockchain trained their data.

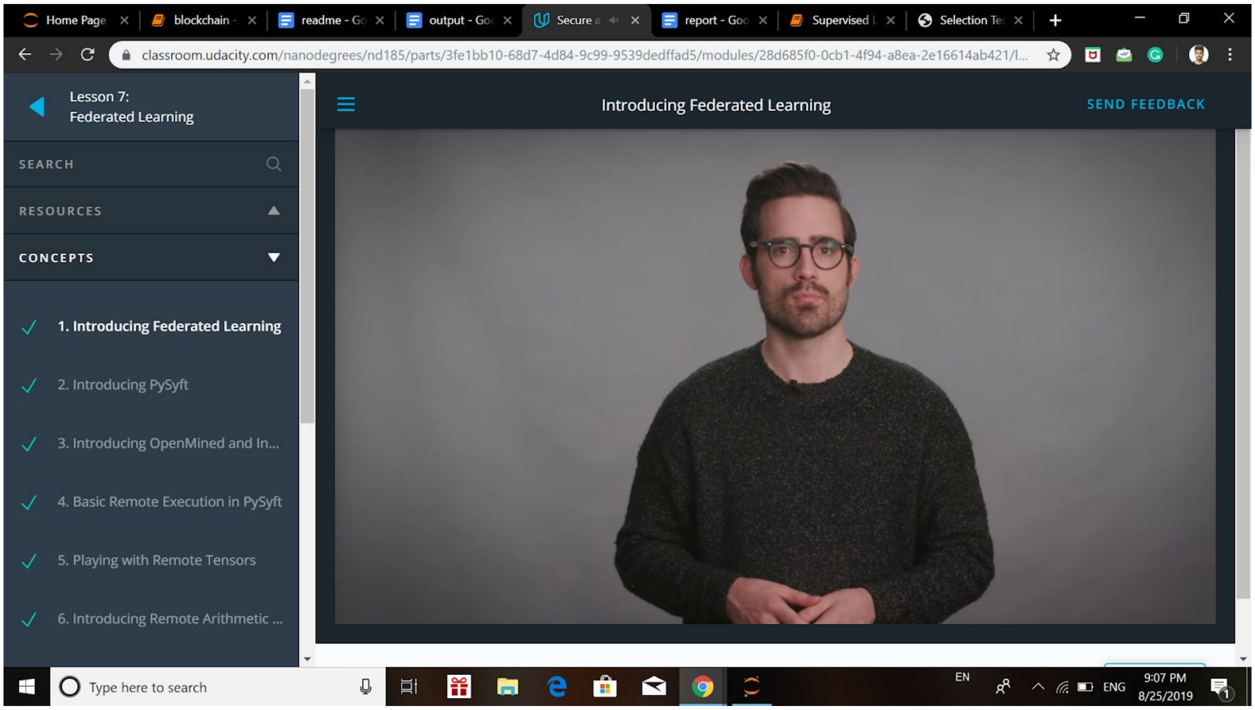
Now the model is ready to build for testing this model has been sent to any new block in a new Blockchain to testing.

Results :

I had iterated my model over 20 times the loss has been decreased from 0.6602 to 0.0001.

References :

1. I had got a Secure and Private AI Scholarship from Facebook. Facebook offers a course at Udacity. In this course, one of the main topics was Federated Learning using PySyft. This helps to solve this problem using Federated Learning. Here are the tutorials on Github.


Home Page
blockchain
readme - Go
output - Go
Secure a
report - Goo
Supervised
Selection Te
classroom.udacity.com/nanodegrees/nd185/parts/3fe1bb10-68d7-4d84-9c99-9539dedffad5/modules/28d685f0-0cb1-4f94-a8ea-2e16614ab421/l...
Lesson 7:
Federated Learning
SEARCH
RESOURCES
CONCEPTS
1. Introducing Federated Learning
2. Introducing PySyft
3. Introducing OpenMined and In...
4. Basic Remote Execution in PySyft
5. Playing with Remote Tensors
6. Introducing Remote Arithmetic ...
Introducing Federated Learning
SEND FEEDBACK
Type here to search
EN
ENG
9:07 PM
8/25/2019

2. Chained Anomaly Detection Models for Federated Learning: An Intrusion Detection Case Study this paper contains *Anomaly Detection through Federated Learning* using the CICIDS2017 dataset. I got an abstract idea to dealt with this problem.

3. BAD: a Blockchain Anomaly Detection solution

4. Tried these articles and courses

- Introduction to Anomaly Detection in Python
- Anomaly Detection | Python - Course Outline - DataCamp

5. Awesome Blockchains