Me:

As shown in the picture this is our hardware representation consisting of 2 main elements Bread board and Microprocessor the Bread Board representing the CAN BUS inside the car Where u can see the MCP chips and the wiring and our Microprocessor we used Raspberry pi 4 with 4 Leds connected from the CAN to the Raspberry pi to light on upon each packet received either a benign packet or one of the 2 attacks we are detecting.  
  
As with accuracy over 90% upon detection of each received packet.

Our future plan till the last discussion is that we will be trying to finish the simulation of the whole script on the Raspberry Pi as this is one of our main targets

Aya:

* A modern Vehicle can have more than 80 ECU continuously comminating as long as the vehicle is turned on
* CAN is a serial communication protocol implemented as a physical serial bus in the vehicle’s network to regulate this communication
* CAN was designed for old basic automotive so with the rises of model technologies, CAN has become a vulnerable target for remote and physical security breaches attacks can easily access it via any point and record the dataflow

Approach:

* We build an embedded software by 1ST developing a reactive deep learning-based intrusion detection system, 2ND deploy this system on a microcontroller. The MC is integrated into the CAN bus to monitor its environment and detect and deviant activity in real-time
* However , this is not as easy as it sounds and here lies the problem

Deep learning based IDSs require high computational resources which exceeds the limitations of embedded electronics and cause detection latency as well

* So, we try to accelerate and compress the deep learning model as much as possible which keeping its performance intact.

Mostafa:  
  
System overview

The system overview of our deep learning-based IDS for in vehicle network detects malicious attacks targeting the CAN bus which regulates communication between the Electronic Control units (ECUs) of the in-vehicle network

-1-In our Pre-processing we prepared the dataset to our model by:

1- Merging both attack free dataset and attack datasets together

2- labeling the unlabeled data instants to identify it whether it is malicious or benign and specifying it

3- Reducing the dimensionality of the dataset by extracting most informative features

4- Applied encoding methods on the trained dataset to suit the input format of our model

-2- Processing

Our training dataset is fed to the deep learning model so it can distinguish between malicious CAN packets and benign ones, to improve the performance of the trained model, its generalization ability is tested with validation datasets. During the process the model is optimized by adjusting and tuning the hyperparameters . Finally, the trained model is evaluated by feeding unseen data instant of datasets and obtaining results

- Second level of processing Compressing and Acceleration

The pretrained model passes through optimization processes aim to compress and accelerate the model achieving optimal computational and memory requirements that comply with microcontroller specifications.

- Third level of processing Deployment

Now that the deep learning model is fully prepared, we move on to the process of integrating it with micro-controller in order to be tested in live environment.

Fully implemented model

The micro-controller unit (MCU) that carries the deployed model is then wired to the CAN bus and the ECUs. The model streams the CAN traffic passing through the bus. Next step is preprocessing the received packets before passing it to the classification step which determines if the stream of packets is benign or malicious. If the stream is malicious, it detects which attack type it is, discards the packets and saves it in a log file for further diagnosis.

Impact

After the deployment of the models on the MCU it will not only increase the security level of the autonomous cars.

SOCIALLY will increase the safety rate of each and every driver or passengers in the vehicles and

INDUSTRIALLY freight transport the safety rate of the goods that are being shipped without causing any damage will increase, as the shipping company will not be affected as it will not pay any compensations due to no loss or destruction of the shipped cargo using the deep learning model.

Omar:  
  
Innovative Aspects:

Our IDS has somewhat similar counterparts in the field already, hence we felt the desire to differentiate and achieve something more than what was offered by them.

And indeed, we were able to create an IDS after several tests that yielded a high accuracy, real-time detection and low false positive and false negative rates.

Now the main problem we focused on was getting our proposed IDS to a state that is ready to be used on the electronic control units that exist within the CAN bus.

We were able to compress our model to a small size, and accelerate it to the needed point to be able to run it on microcontrollers. This makes our model the first of

its kind to be ready for use in the industry.  
  
  
  
  
  
Technology Used:

The technologies we used include the LSTM, a very prominent deep learning network that has made a great reputation for itself today in the machine learning field.

We use a Raspberry Pi 4 to act as the microcontroller holding the IDS, and have utilized the Tensorflow Lite API to compress and accelerate our model.