**Assignment 1**

**Develop Supervised Machine Learning Models**

Goal of the assignment: This assignment aims to have you practice exploring datasets, developing supervised machine learning models, and analyzing their performance. We will use in this assignment a dataset extracted from the in-vehicle network on an in-motion vehicle. The machine learning models that you should develop need to detect whether the vehicle is under-attack or not at different time steps.

**Context:**

On newer vehicles, Electronic Control Units (ECUs) communicate using in-vehicle net-works to control their behaviors, as shown in Fig 2.1 For example, the figure shows that the engine control unit communicates with other controllers such as the odometer and brakes over the Controller Area Network (CAN), among other bus systems. Several Intelli-gent Transportation Systems applications have been proposed and implemented recently to improve drivers’ experiences and road safety. These applications include infotainment systems, fleet management systems, parking assistance, remote diagnostics, eCall, remote engine start, and Cooperative Adaptive Cruise Control (CACC) systems. To operate correctly, these Intelligent Transportation Systems (ITSs) must communicate using the CAN bus to exchange messages between the ECUs, sensors, and applications. Technically, CAN is a time-synchronized broadcast, multi-cast reception message network bus. The figure 2.2 illustrates the CAN message format. Each message consists of a data frame, remote frame, error frame, or overload frame. The data frame is used to exchange data between the nodes, the remote frame is used to request the transmission of a specific identifier, the error frame is transmitted by any node detecting an error, and the overload frame is used to inject a delay between data or remote frames. The CAN was designed to operate as a closed network system; that is, all the communicating nodes are trusted. The CAN does not support security mechanisms to identify and isolate malicious nodes, such as authentication. This design was not a problem when automobiles were standalone entities; however, today, ITS applications communicate with external entities, such as other vehicles, in the case of CACC, which makes it possible for attackers to exploit vulnerabilities in these applications and attack the connected vehicles.

1. In normal driving.

2. Under spoofing the speed reading (by injecting fabricated speed CAN messages).

3. Under spoofing the RPM reading (by injecting fabricated RPM CAN messages).

12 CHAPTER 2. SUPERVISED LEARNING FROM EXAMPLES

**Assignment Tasks**

**Task 1:** Data preparation

**Step 1.** You are provided three files, i.e., text files containing bus logs of 3 different scenarios

· CAN Bus log - injection of FFF as the speed reading

· CAN Bus log - injection of RPM readings

· CAN bus log - no injection of messages

**Step 2.** A Python module to read the datasets is available. Download the module and use it

to create three separate data frames for the cases: (1) Injection of FF as Speed, (2) Injection of RPM, and (3) No injection. The last field of the data sets marks the CAN message as an injected message or not; it takes value 1 for attack/injection and 0 for the legitimate messages.

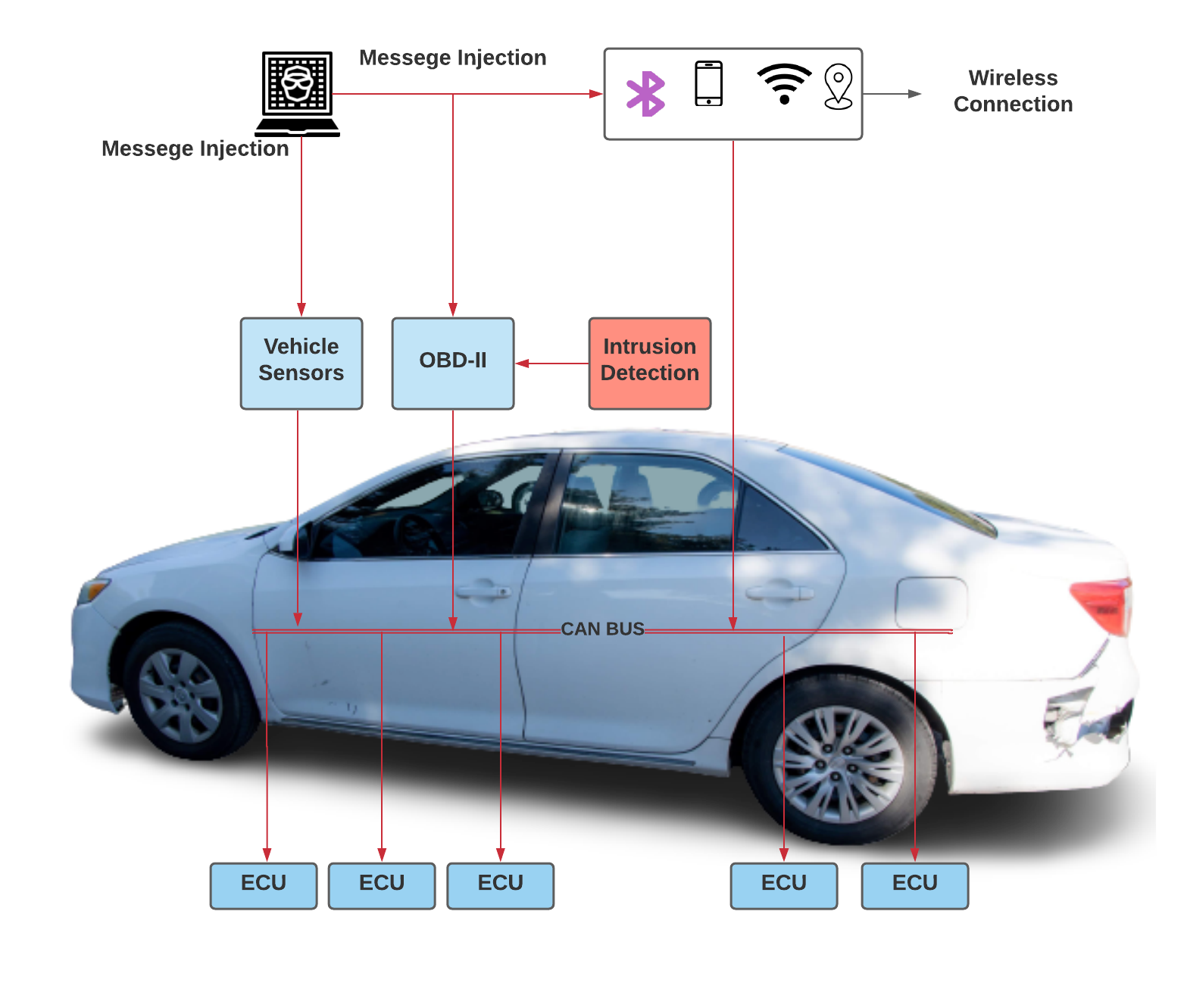


Figure 2.1: engine control unit

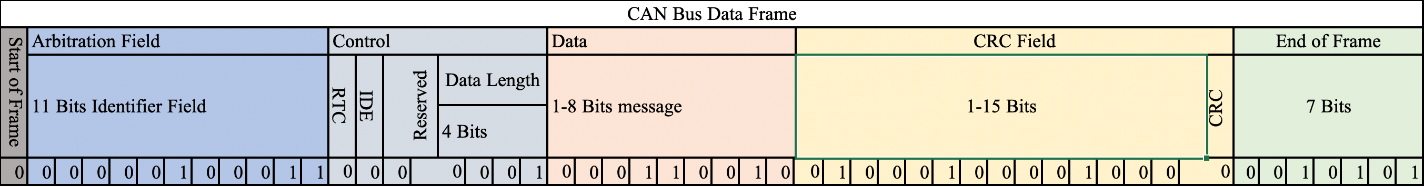


Figure 2.2: CAN

Cyber-attacks goal on connected vehicles is to inject CAN messages that aim to change the vehicle’s behavior, such as increasing/decreasing the speed or disabling the break. You are given in the following datasets collected from the CAN bus of a vehicle:

**Task 2:** Explore and analyze the data

**Step 1** - Create scatter plots for the three scenarios below. Remember to filter the data and select only the appropriate records needed to answer the question. You should use ’254’ as the CAN ID (represented by arbitration field in the CAN frame above) for the speed reading and ’115’ as the CAN ID for the RPM reading.

1. Change of speed over time. Plot the change of speed readings (CAN ID 254) over time to observe how it changes over the data collection period. Note: Index of the data is considered as time
2. Change of RPM over time. Plot the change of RPM readings (CAN ID 115) over time to observe how it changes over the data collection period. Note: Index of the data is considered as time
3. Relationship between speed and RMP. To understand the interplay between speed and RPM, plot the speed readings on the x-axis and RPM readings on the y-axis. This scatter plot will help identify any potential connections between these two variables

Reflect on the nine plots (three plots for each scenario) and report your observations in about two paragraphs. As you examine the scatter plots, note patterns and trends that emerge. For example, sudden spikes or drops in speed readings might correspond to specific driving events, and changes in RPM readings may signal engine adjustments. The speed vs. RPM scatter plot might reveal whether there’s a consistent relationship between the two parameters

**Step 2** - Create frequency plots for speed and RPM readings. These plots should show the distribution of these readings.

1. Frequency Plot for Speed: Visualize the frequency of speed values using a histogram like plot, with speed ranges on the x-axis and their corresponding frequencies on the y-axis.

2. Frequency Plot for RPM. Similarly, create a frequency plot for RPM readings to understand their distribution.

3. Observations. These frequency plots reveal common speed and RPM ranges and highlight any unusual patterns. Distributions that differ significantly from typical driving conditions could suggest anomalies or potential attacks. Reflect on the plots and write your observations in one or two paragraphs.

**Step 3.** Use Pearson-Correlation to analyze the relationships between the Speed and RPM readings plots in all three scenarios. Create a correlation table depicting the correlation coefficients and the associated P-value. Reflect on the values you obtained and write one or two paragraphs to report your observations.

**Task 3** - Supervised Machine Learning Mode

You are tasked to apply a supervised ML method of your choice (Random Forest, XGBoost, Decision Tree, Support Vector Machine, Logistic Regression, etc.) to detect if there have been attacks in the vehicle or not for each of the three datasets.

1. Combine the three datasets into one large dataset.
2. Split the datasets into two sets: 3/4 as the training dataset and 1/4 as the testing dataset.
3. Train your choice of ML model on the training dataset.
4. Evaluate your model by testing it using the test data set. Create a confusion matrix that summarizes your results.
5. Report the results of your model and your analysis of its performance.

**Analysis.** Examine the confusion matrix for deeper insights into your model’s efficiency. While accuracy is important, consider metrics like precision, recall, and F1-score for a holistic view of the model’s performance in detecting attacks.

**Task 4** - Second Choice of ML model Select another supervised ML technique and repeat the steps of Task 3 with that.

**Task 5** - Compare and contrast models trained on Tasks 3 and 4

**Step 1:** Compare the models you obtained in tasks 3 and 4 and report your observations and analysis. Do not simply write, “Model 1 is better than Model 2 because Model 1 has higher accuracy”. One way to compare would be to use the confusion matrix created on Tasks 3 and 4 and compare False Negative Rates.

**Task 6** - Discussion Reflect on your learning from this assignment. Questions that you may consider are: What were the shortcomings/limitations of using supervised ML on this problem? What could be done to make the models perform better? What do you suggest to do to solve the problem better?

Report your reflection in one or two paragraphs