

Improving the Resiliency of Embedded Networks in Heavy Vehicles - Towards Fault Tolerance

Chandrima Ghatak, Saira Jabeen, Indrakshi Ray
Department of Computer Science
Colorado State University

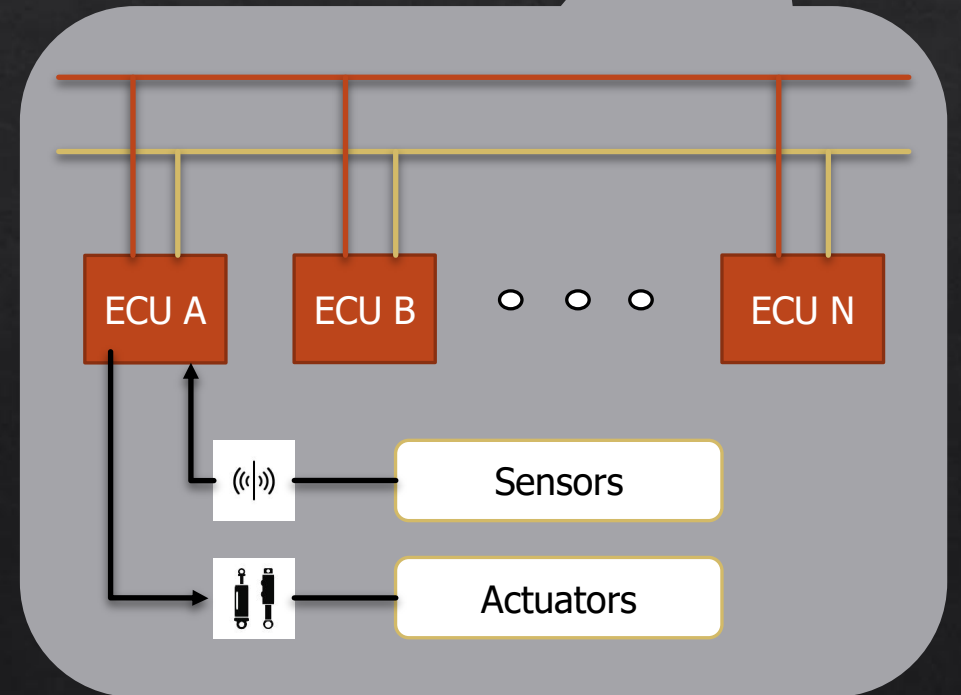
Hossein Shirazi
Department of Management Information Systems Department
San Diego State University



Colorado State University

Goal of the Research

- ◇ Problem:
 - ◇ MHD Vehicles: Crucial to modern infrastructure.
 - ◇ ECUs in MHDs rely on sensor data exchange.
 - ◇ Current protocols (e.g., SAE's J1939) are robust but vulnerable to sensor malfunctions & cyber-attacks.
 - ◇ Consequences: Vehicle goes into 'limp mode' or total shutdown.
- ◇ Limitations of Traditional Solutions:
 - ◇ Limited adaptability & predictive accuracy.
 - ◇ Manual updates; can't adapt to new threats in real-time.
- ◇ Our Solution:
 - ◇ Use Machine Learning (ML) for dynamic, real-time resilience.
 - ◇ Unified neural network model trained on diverse sensor data.
 - ◇ Reduces complexity, ensures broad applicability.



Kenworth T270 Research Truck

- ◆ Truck stalled on US I-80 , near Ogallala, NE
- ◆ Reason: One fuel injector malfunctioned
- ◆ Solution: Our predictive values for the sensor, reach the nearest mechanic without breaking down.



Agenda



Background
on Heavy
Vehicle
Networks and
SAE J1939




Data Preparation
and Pre-Processing



Experiments
and Results



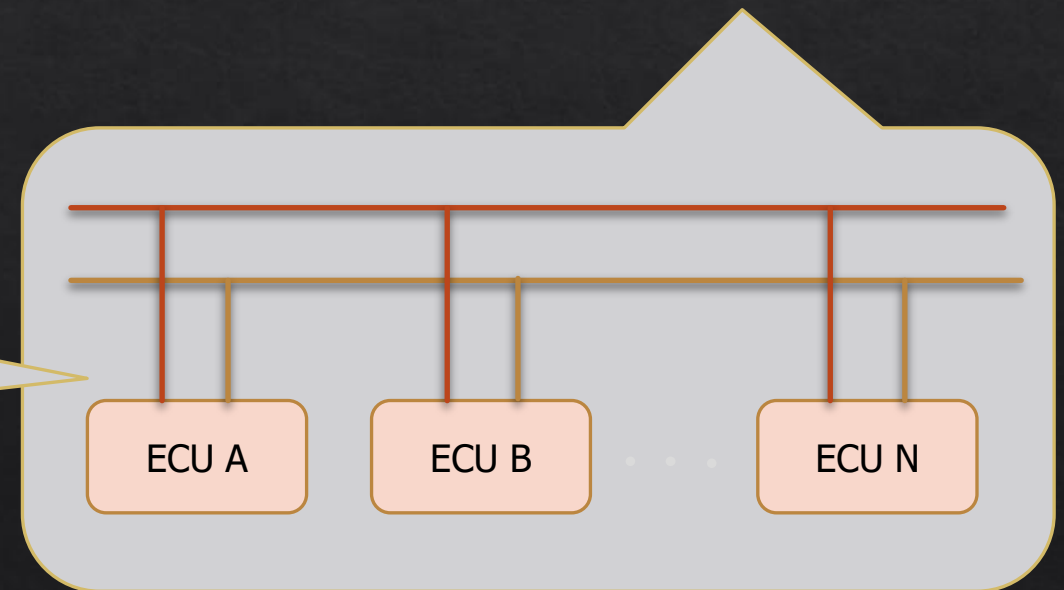
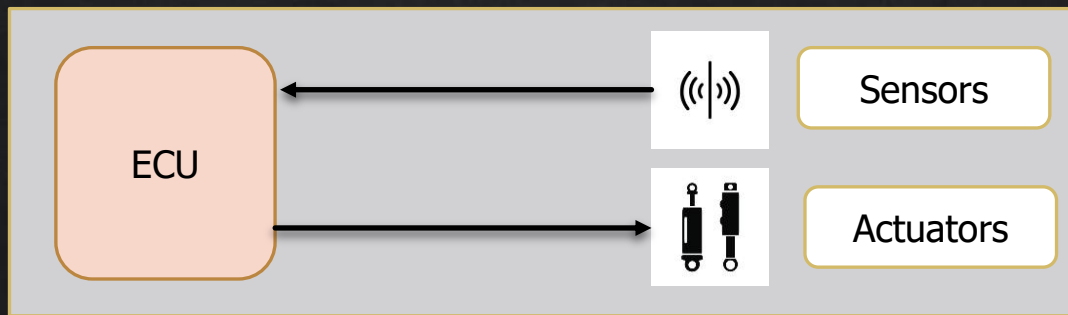
Future
Work



**Background on Heavy
Vehicle
Networks and SAE J1939**

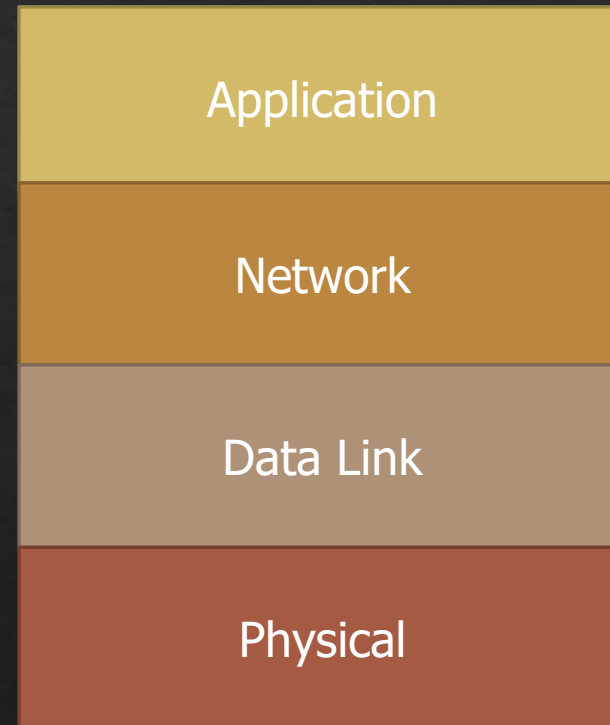
Heavy Vehicle Networks

- ◆ Most commercial vehicles are controlled by embedded computer called Electronic Control Units (ECUs).
- ◆ These ECU control most operations in a modern vehicle.
- ◆ ECUs gather data from sensors and send them over a two-wire multi master bus.
- ◆ ECUs use this sensor data to control various operations with the help of actuators.



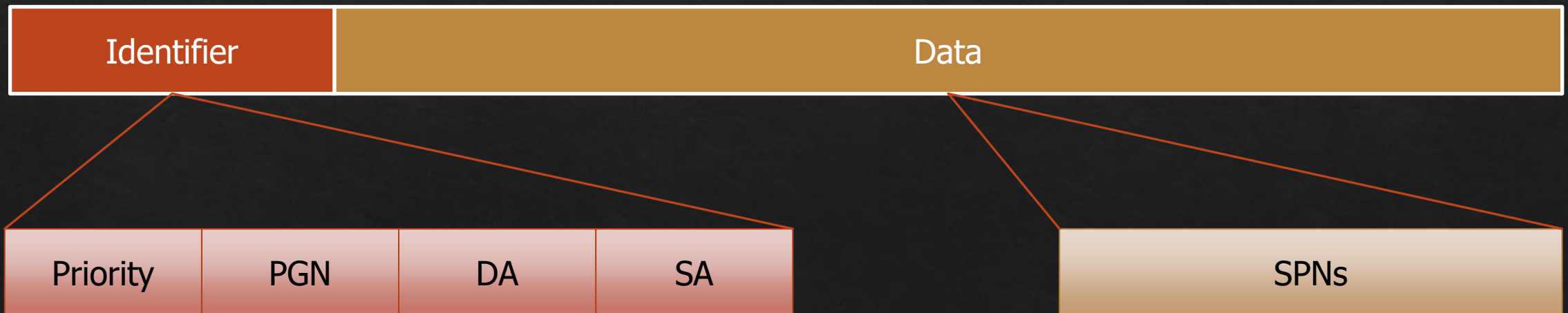
SAE J1939 Protocol

- ◆ Commercial vehicles in the United States use the SAE J1939 Protocol on top of the Controller Area Network (CAN) specifications to communicate amongst ECUs.
- ◆ The SAE J1939 Protocol is designed based on the ISO/OSI protocol stack.
- ◆ The SAE J1939 Protocol mainly use the application, network, data link and physical layer of the ISO/OSI protocol stack.

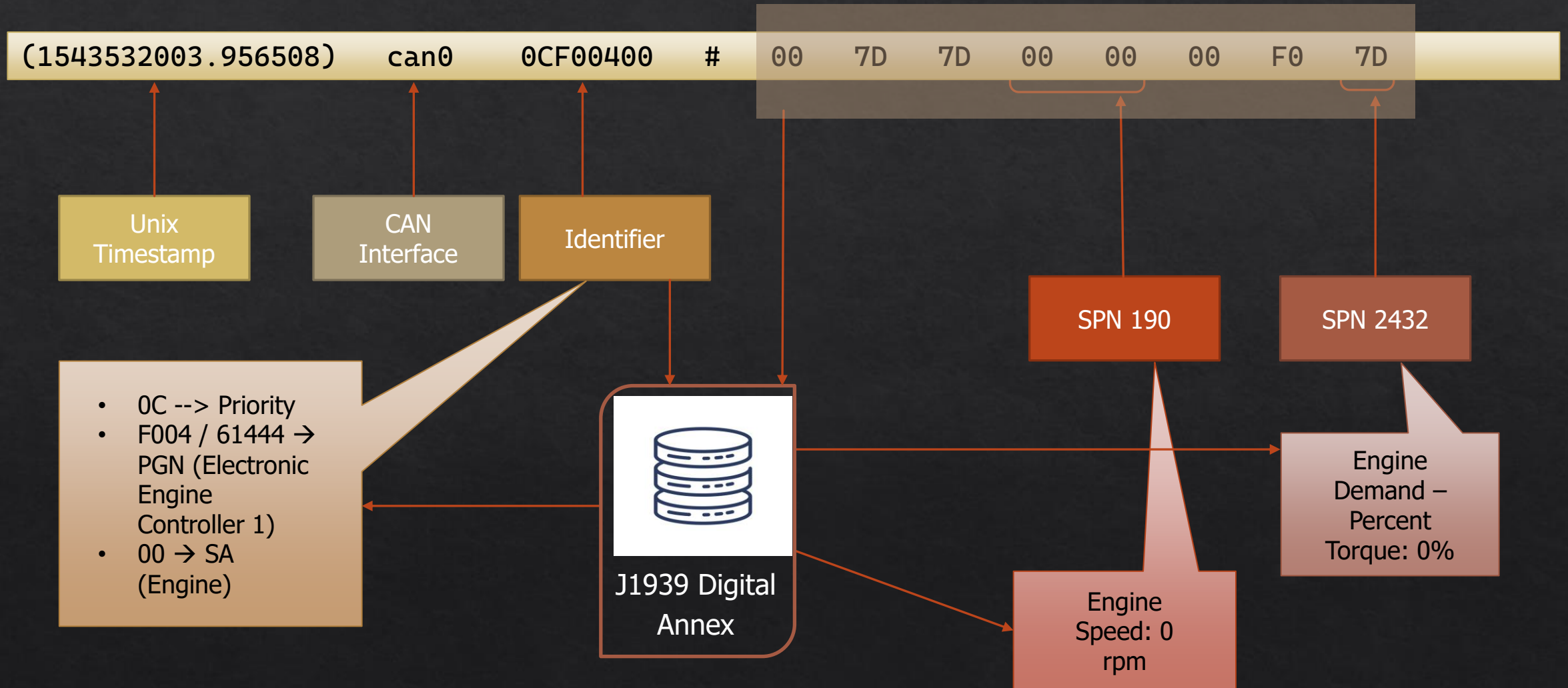


J1939 Frame Format

- ◇ A J1939 Frame contains an Identifier and a variable length data field.
- ◇ The Identifier field comprises of a Priority, a Parameter Group Number (PGN), a Destination Address (DA) and a Source Address (SA).
 - ◇ If first byte of PGN is < 240 the next byte denotes DA (signaling the message is destination specific).
 - ◇ If first byte of PGN is ≥ 240 the next byte denotes PGN extension(signaling the message is broadcast).
- ◇ The Data Field consists of Suspect Parameter Numbers (SPNs) which contain ASCII encoded data for physical sensor values, status or a command.



Understanding J1939 Frames



Related Work

Our research is based on previous efforts to reconstruct compromised sensor values and mainly takes inspiration from the paper '**Towards Resiliency of Heavy Vehicles through Compromised Sensor Data Reconstruction**' by Shirazi et al. [1]

- ◇ Most of the works in heavy vehicle security either focus on simulating attacks or develop Intrusion Detection Systems (IDS) that can detect attacks and notify the system. [2, 3, 4, 5, 6]
- ◇ Existing IDSs can only detect compromised devices through anomalous values generated by them.
- ◇ If an Electronic Control Unit (ECU) has been compromised, how its data values can be reconstructed from non-compromised values, which IDS fails to address.
- ◇ Long Short-Term Memory (LSTM) is used to train multivariate LSTM models when multiple values generated from ECUs have been used as input so that the models may learn the relationships between the different vehicle parameters.
- ◇ A target output can be reconstructed using only these alternate inputs while ignoring the original, compromised data. This can mitigate the effects of an attack if one is detected and has the added benefit that reconstructed data are immune from the effects of the compromised data.

Limitations of this Approach

1. Models were trained and tested on limited number of SPNs. There is a huge number of sensor data that is present in a heavy vehicle network.
2. This approach only works when one sensor or SPN value is compromised or missing. We need an approach that can predict when more than one SPN values are missing or compromised .
3. This approach was never tested in real time in a real heavy vehicle network.
4. Thus, in our research, we try to address these issues.

Our current research aims to address these limitations through the following:

1. We considered all measured sensor values present in a heavy vehicle network.
2. Our approach is designed to work when one or more sensor values are missing.
3. Our future goal is to deploy our approach in a real heavy vehicle and test it in real-time.



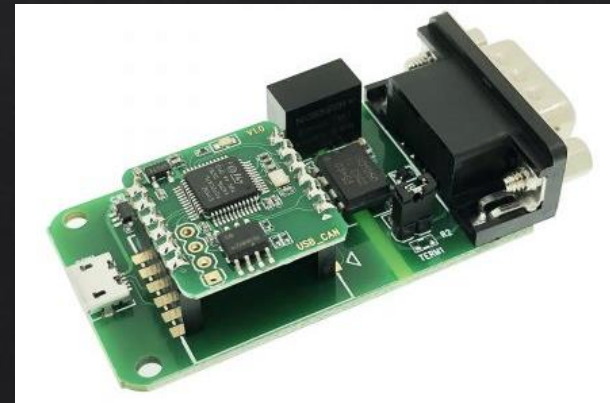
Data Preparation and Pre- Processing

Data Preparation

- ◇ Data Collection:
 - ◇ From a 2014 Kenworth T270 truck, during a 2018 trip from Fort Collins to Detroit.
 - ◇ Data in 'candump' format with time-stamped ASCII values.
- ◇ Data Processing:
 - ◇ Isolated relevant PGNs using SAE-J1939 standard.
 - ◇ Focused on 52 non-static sensors for effective training and testing.
 - ◇ Sampled sensor data at 500ms intervals for a time-series dataset.
 - ◇ Normalized dataset values between 0 and 1.



CAN Logger



CAN-2-USB

Experimental Setup

- ◇ We have conducted experiments with the following neural networks, to compare and study which gives optimum results to be used in real time.
- ◇ All these methods have been used for time series forecasting and have shown promising results for predicting erroneous inputs.
- ◇ LSTM was shown to have good results for J1939 sensor data by Shirazi et al. [1]
- ◇ Current research at CSU on Dense and Sparse Binary Transformers have also shown promising results.

Long Short-
term Memory
(LSTM)

Dense Binary
Transformers
(DBT)

Sparse Binary
Transformers
(SBT)

Experimental Setup (Contd.)

- ◇ Algorithm Optimization:
 - ◇ Training under varied conditions, focusing on missing values.
 - ◇ Average Percentage Error (err) used for accuracy measurement.
- ◇ Training Strategies:
 - ◇ Full-data Training: Models trained with the complete dataset.
 - ◇ Training with Missing Values: Simulated sensor data unavailability.
- Dataset and Training:
 - 16,000 instances, split into 75% training and 25% testing.
 - Models trained for 100 epochs using NVIDIA TITAN V.
- Performance Evaluation:
 - Mean Squared Error (MSE) used for comparison.
 - Optimal performance observed at a window size of 10.

Window Size	Training with all data			Training with 5% of missing data		
	DBT	SBT	LSTM	DBT	SBT	LSTM
10	0.2446	0.5901	0.4810	0.2432	0.4969	0.4952
50	0.2767	0.6391	0.5710	0.2661	0.5886	0.6721
100	0.3174	0.6784	0.6312	0.2638	0.6453	0.7611
200	0.4850	0.7537	0.7005	0.4209	0.6892	0.9023

Table 1: Mean Squared Error on Test Data

Model Architecture and Hyperparameters

- **Randomized Model Search:**
 - For finding best hyperparameters (optimizer, learning rate, etc).
- **Best-Performing Architectures:**
 - LSTM:
 - Input layer with specific shape for past observations.
 - Two encoder LSTM layers, returning sequences and states.
 - A ‘RepeatVector’ layer and two decoder LSTM layers.
 - ‘TimeDistributed’ wrapper around a ‘Dense’ layer.
 - ◊ DBT and SBT:
 - ◊ Positional encoding size: 64 units.
 - ◊ Four encoder layers and self-attention heads.
 - ◊ Hidden layers size: 128 units.

Average Percentage Error Calculation

◇ For Error Calculation, we considered Percentage error based on Range of each SPN to be optimum for this project.

◇ For the equation here:

$$err = \frac{1}{n} \sum_{i=1}^n \frac{e_i}{z} \times 100,$$

- err = Average Percentage Error
- e_i = error in prediction for each data point for a particular SPN value
- z = range of each SPN
- n = number of all data points in the dataset for the particular SPN



Experiments and Results

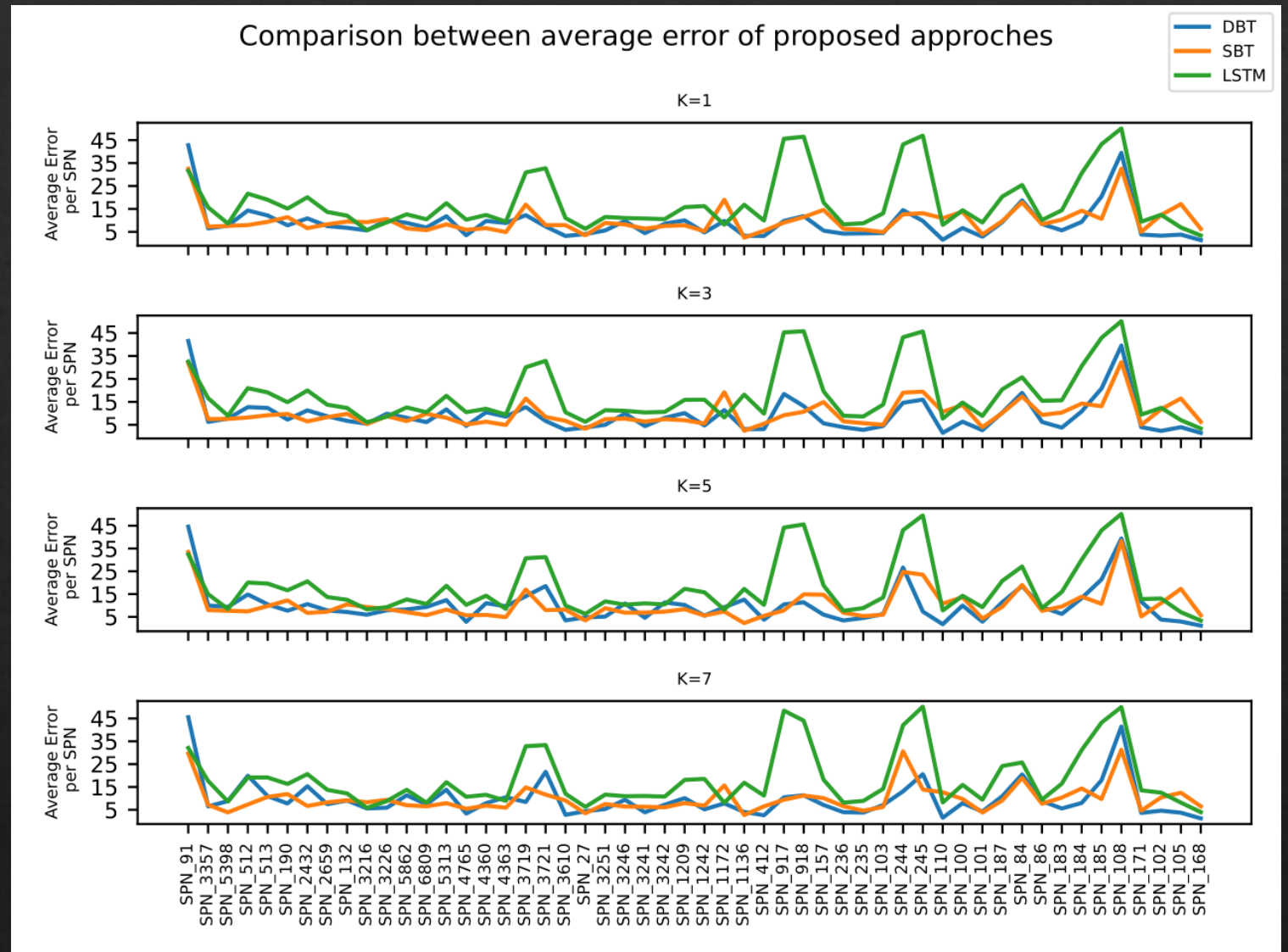
**Single Model to Predict All SPNs
with One or More Missing SPNs at a
Time**



Test Case 1: Missing one or more SPNs

1. Performance Variation:
 - ◇ Dependent on algorithm, training data nature, and specific SPN.
 - ◇ Highlighted importance of algorithm selection and training data nuances.
2. Testing Methodology:
 - ◇ Simultaneously missing data points and their previous values within a given window.
 - ◇ Random selection of $k-1$ SPNs from 51, predicting the k -th SPN.
3. Results:
 - ◇ LSTM shows highest error across all values of k .
 - ◇ DBT demonstrates lowest error, suggesting better feature learning.
 - ◇ Increased k value doesn't significantly affect average error for most SPNs.

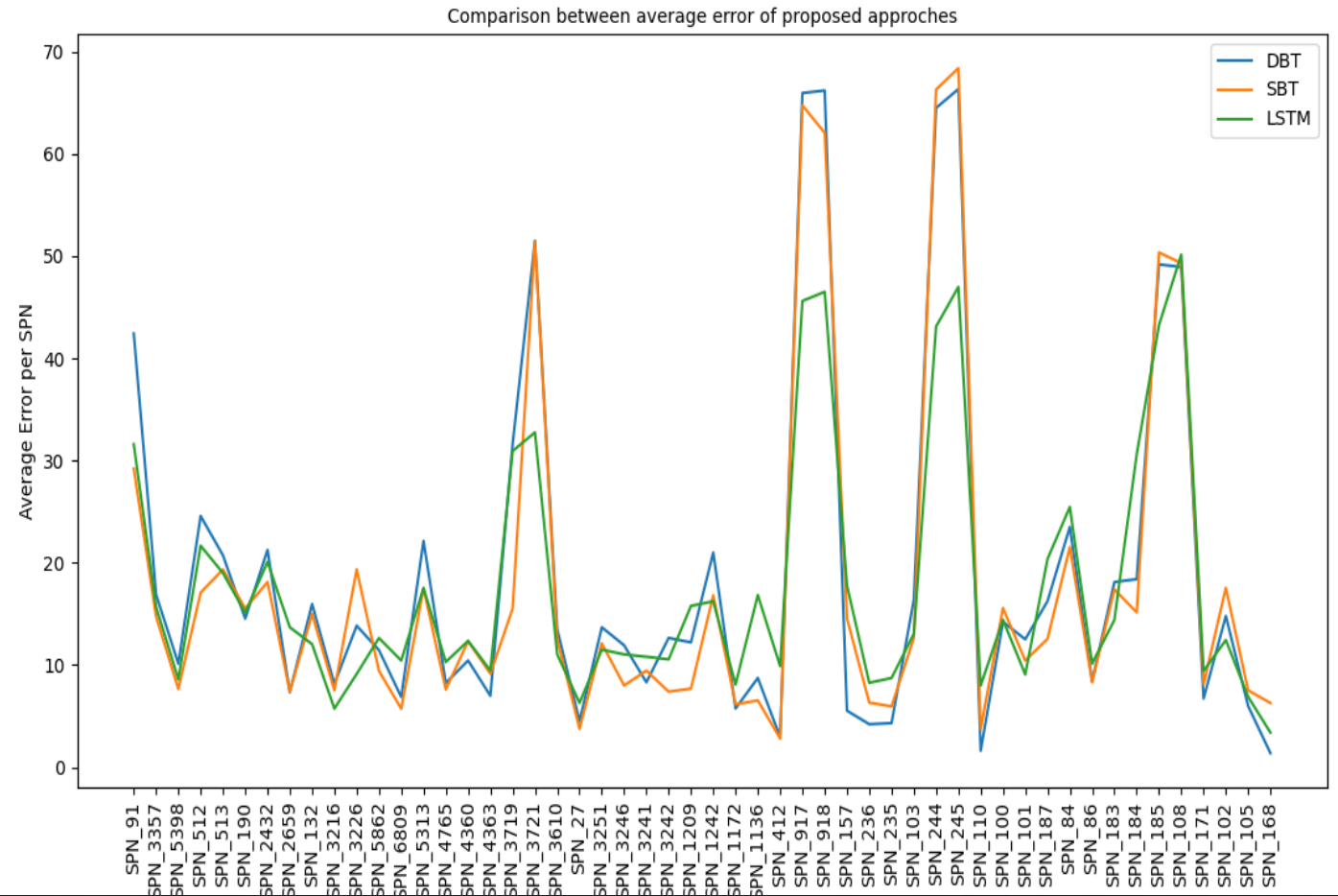
Error comparison at 4 different values of k, where k represents the number of missing SPNs



Test Case 2: Missing SPN as well as its correlated SPNs

- ◇ Correlation Identification:
 - ◇ Correlated SPNs determined using Pearson Correlation (>0.5).
- ◇ Comparison of Model Performances:
 - ◇ All models tested with a window size of 10.
 - ◇ Highest error when target SPN's correlated SPNs are missing.
- ◇ Observations:
 - ◇ LSTM performs best, focusing on past temporal patterns.
 - ◇ Transformer models learn inter-sensor dependencies.
 - ◇ Each model type has distinct advantages.
- DBT Specific Findings:
 - Examined in two scenarios: missing random SPNs and missing correlated SPNs.
 - Average prediction error increases with the number of missing SPNs, but not significantly.
 - Error peaks when correlated SPNs are missing, highlighting transformer models' learning capabilities.

Error comparison
between proposed
approaches when
correlated SPNs
are missing



Single Model to Predict Multiple Future Steps for One SPN Missing at a Time



Test Case 1: One-shot Method

1. Methodology:

- ◇ Predicts missing sensor value at a specific future time step n directly.
- ◇ Uses remaining 50 sensors as input for the prediction.

2. Advantages:

- ◇ Computational efficiency and faster prediction times.
- ◇ Minimizes error propagation risk.

3. Results:

- ◇ DBT shows lowest prediction errors across most future steps.
- ◇ LSTM and SBT errors higher, especially for long-term predictions.
- ◇ LSTM exhibits highest errors at steps 50 and 100.

Test Case 2: Recursive Feeding Method

1. Methodology:

- ◇ Predicts one step ahead, then feeds this prediction back to forecast the next step.
- ◇ Repeats the process recursively for desired n-th step.

2. Advantages:

- ◇ Simplicity and flexibility.
- ◇ Provides intermediate forecasts for all steps up to the target.

3. Results:

- ◇ DBT again has lower prediction errors for almost all future steps.
- ◇ LSTM shows improvement over One-shot Method, especially in short-to-medium steps.
- ◇ SBT remains higher in error but improves in long-term predictions.

Results

- Conducted two types of experiments: One-shot Method and Recursive Feeding Method.

Model	Average Errors for n-steps in future prediction (in %)								
	1	3	5	10	15	20	25	50	100
DBT	4.90	4.45	4.86	5.20	5.41	5.36	7.35	9.43	11.88
SBT	6.90	6.78	6.81	7.31	7.19	7.67	8.62	10.93	14.22
LSTM	5.66	5.45	5.56	5.86	5.93	7.62	9.32	12.93	15.86

Table 2: Average Errors for n-steps in future prediction (in %) for One-shot Method

Model	Average Errors for n-steps in future prediction (in %)								
	1	3	5	10	15	20	25	50	100
DBT	4.90	3.73	3.55	4.01	4.66	4.70	6.04	8.18	10.33
SBT	6.90	5.97	5.40	6.05	5.91	6.27	7.04	9.55	13.17
LSTM	5.66	4.21	4.79	4.34	4.57	6.11	8.13	11.50	14.31

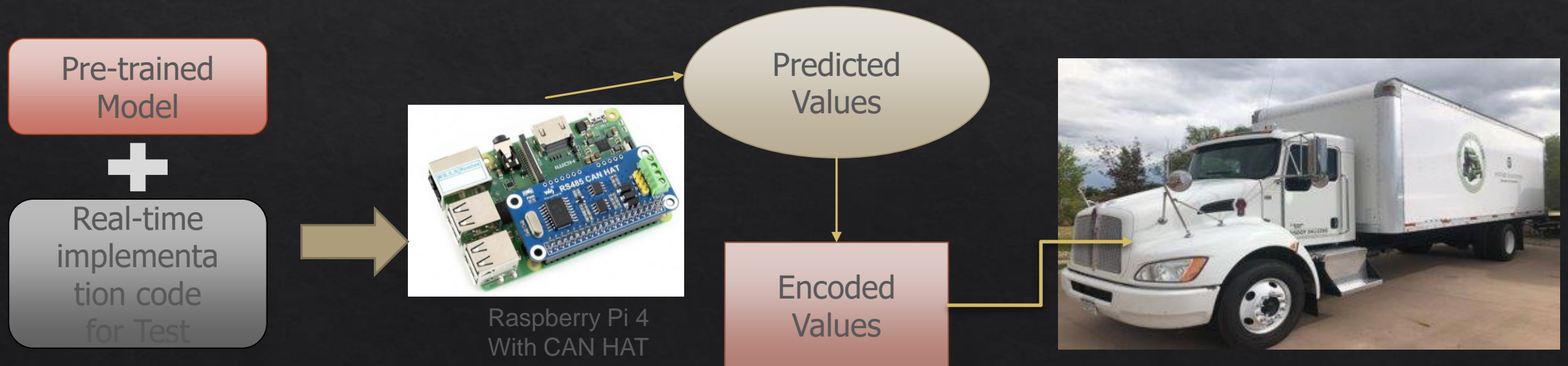
Table 3: Average Errors for n-steps in future prediction (in %) for Recursive Feeding Method



Future Work

Future Work

- ◇ Real-world Integration: Implement models in operational MHDs, e.g., 2014 Kenworth T270.
- ◇ Malfunction Simulations: Real-time on-road experiments to simulate sensor malfunctions.
- ◇ Resilience Evaluation: Reintroduce predicted values into vehicle's control for holistic resilience assessment.



Thank you



Colorado State University



Questions ?

References

1. Hossein Shirazi, William Pickard, Indrakshi Ray, and Haonan Wang. 2022. Towards Resiliency of Heavy Vehicles through Compromised Sensor Data Reconstruction. In Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy (Baltimore, MD, USA) (CODASPY '22). Association for Computing Machinery, New York, NY, USA, 276–287. Kyong-Tak Cho and Kang G. Shin. 2016. Fingerprinting Electronic Control Units for Vehicle Intrusion Detection. In 25th USENIX Security Symposium, USENIX Security 16. 911–927
2. Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X. Liu, and Schahram Dustdar. 2022. Pyraformer: Low-Complexity Pyramidal Attention for Long-Range Time Series Modeling and Forecasting. In International Conference on Learning Representations.
3. Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. Proc. of AAAI Conf. on AI (2021)
4. Matt Gorbett, Hossein Shirazi, and Indrakshi Ray. 2023. Sparse Binary Transformers for Multivariate Time Series Modeling. Proc. of ACM SIGKDD '23
5. Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. Neural computation 9, 8 (1997), 1735–1780.
6. Hossein Shirazi, Indrakshi Ray, and Charles Anderson. 2020. Using Machine Learning to Detect Anomalies in Embedded Networks in Heavy Vehicles. In Foundations and Practice of Security, Vol. 12056. Springer International Publishing, 39–55.
7. Jeremy Daily, SystemsCyber, Colorado State University, URL: <https://www.engr.colostate.edu/~jdaily/>