CONTROLLER AREA NETWORK (CAN) DEEP PACKET INSPECTION

NVIDIA

Görkem Batmaz, Systems Engineer
Ildikó Pete, Systems Engineer

28th March, 2018

Car Hacking

“Immediately my accelerator stopped working. As I frantically pressed the pedal and watched the RPMs climb, the Jeep lost half its speed, then slowed to a crawl.” (Andy Greenberg, Wired)

2014 Jeep Cherokee (remote attack)

Engage brakes, Take control of steering

Agenda

AUTOMOTIVE SECURITY

- Connectivity in Modern Vehicles
- Controller Area Network (CAN) Vulnerabilities

CAN ATTACKS

- Attack Types
- Detection & Prevention

CAN ANOMALY DETECTOR

- Data
- Approach

RESULTS & CONCLUSIONS

- Discussion of Results

1

Increasing Complexity & functionality

2

Interconnectedness

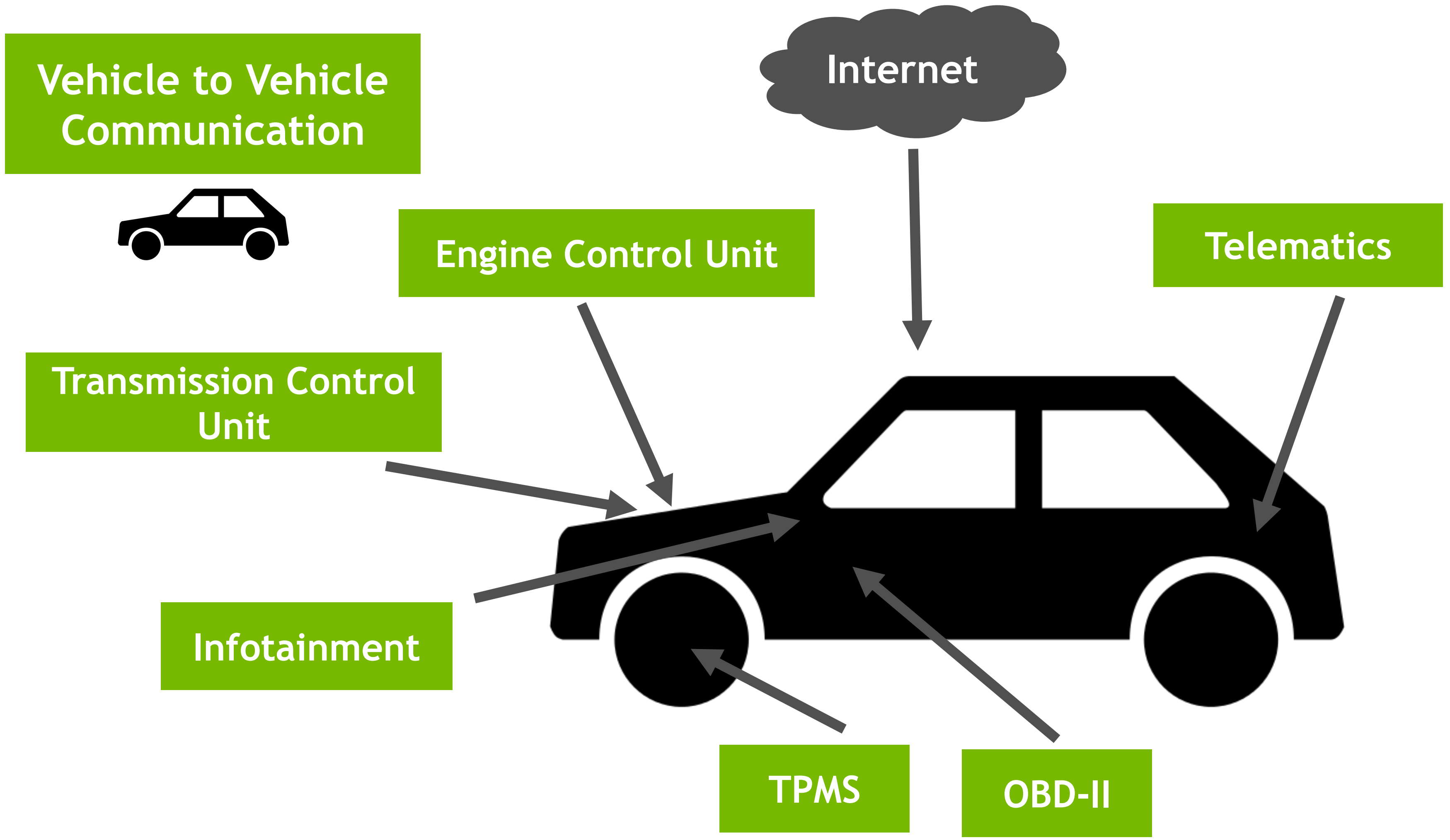


Figure1. Some connections of a modern car

Controller Area Network (CAN) Security

CAN Characteristics

Message types: Information, Diagnostic

Message exchange: Broadcast

Message-based protocol, no addressing

Arbitration method to resolve priorities

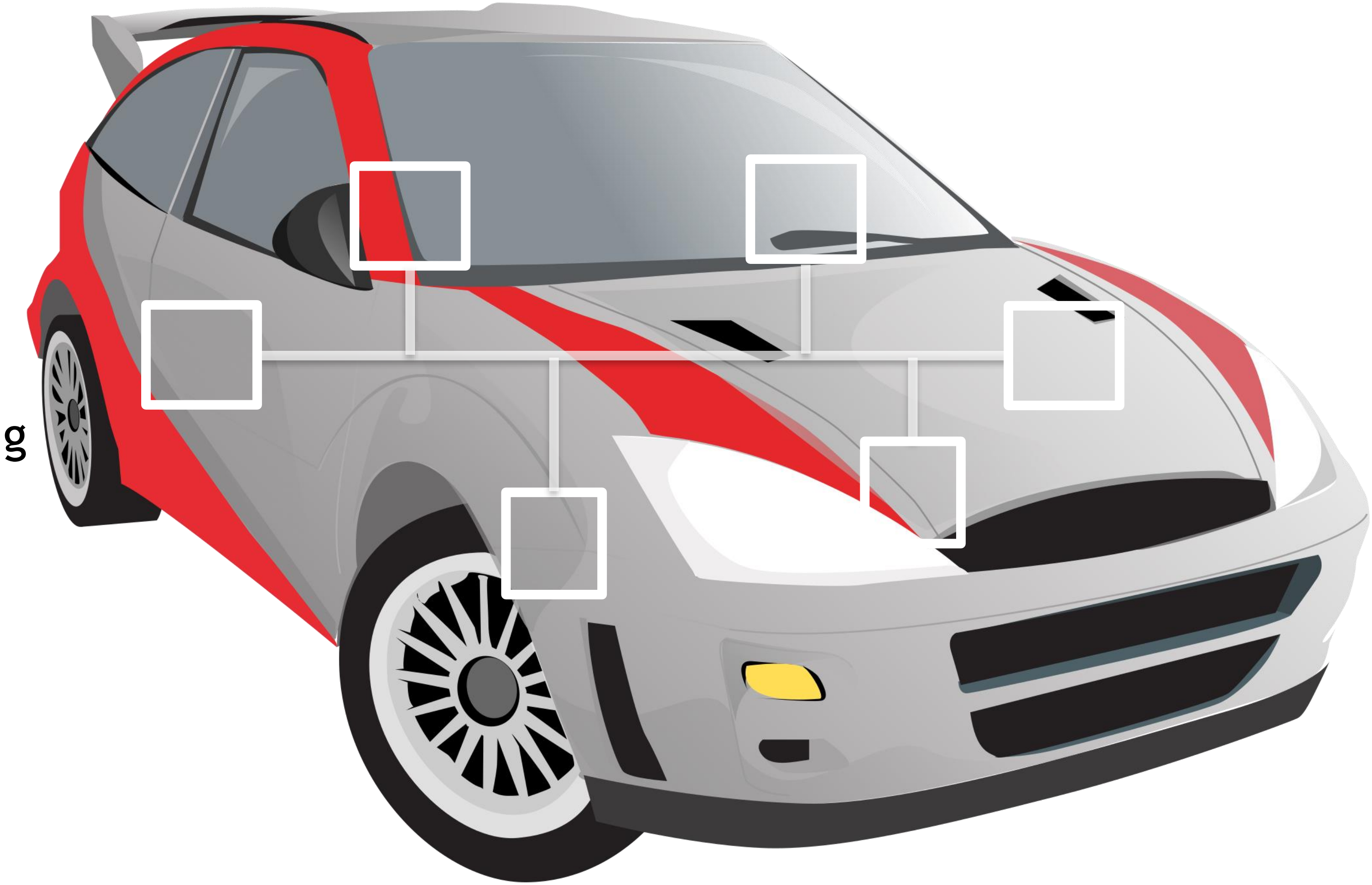


Figure2. The CAN network

CAN Vulnerabilities

Confidentiality	Every message sent on CAN is broadcast to every node → Eavesdropping
Authenticity	Lack of sender authentication → Masquerading
Availability	Arbitration rules (high priority messages) → Denial of Service
Non Repudiation	No mechanisms to prove an ECU sent or received a message

Most Critical Attack Types on CAN

REPLAY	INJECTION	DOS
Replace message contents with some pre-recorded values	Inject false messages appearing to be legitimate	Flood the network

Detection & Prevention

Device identification

CRYPTOGRAPHIC SERVICES

ANOMALY DETECTION

Over-the-air updates

ANTI-MALWARE

ECU software integrity

Tamper detection

Secure boot

Anomaly Detection

Finding unusual patterns in data that do not conform to expected behavior

E.g. fraud detection

Types of Anomalies

Point Anomaly	Collective Anomaly	Contextual (Conditional) Anomaly
E.g. vehicle speed is **500 miles/hour**	E.g. vehicle speed is **80 miles/hour** & steering wheel angle is **90 degrees**	E.g. vehicle speed changes from **50 miles/hour** to **80 miles/hour** in less than **X seconds**

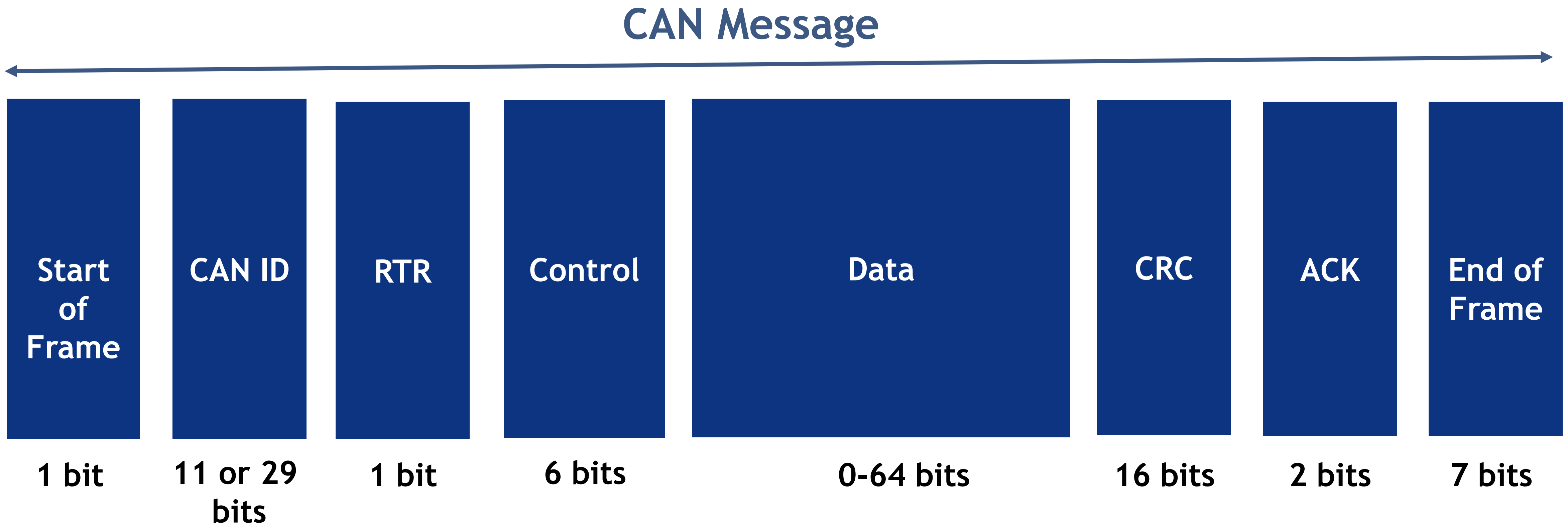
Controller Area Network (CAN) Anomaly Detector

Detect security-related CAN network anomalies resulting from malicious activities

Attacks: Injection, Replay

Anomalies: Contextual

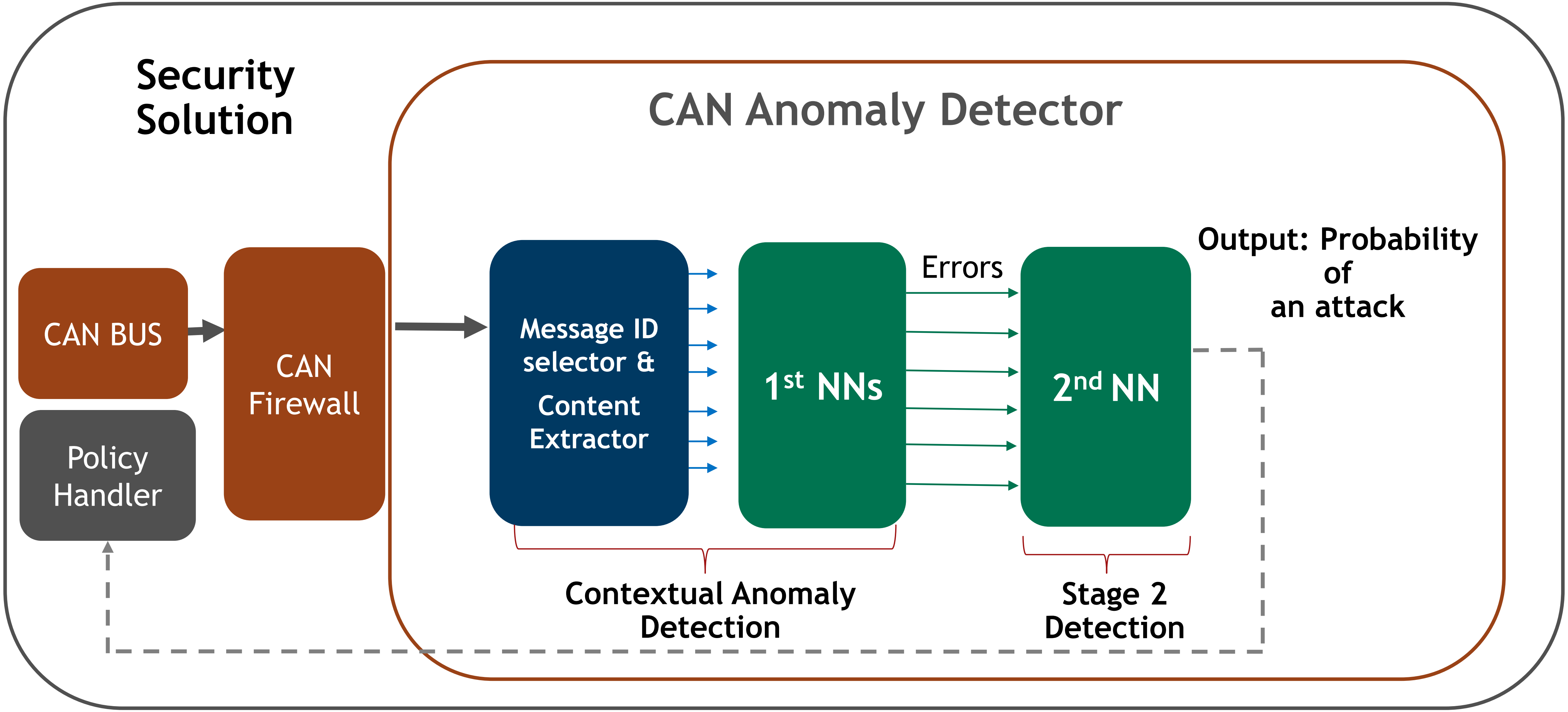
CAN Frame


CAN Message
Start of Frame
1 bit
CAN ID
11 or 29 bits
RTR
1 bit
Control
6 bits
Data
0-64 bits
CRC
16 bits
ACK
2 bits
End of Frame
7 bits

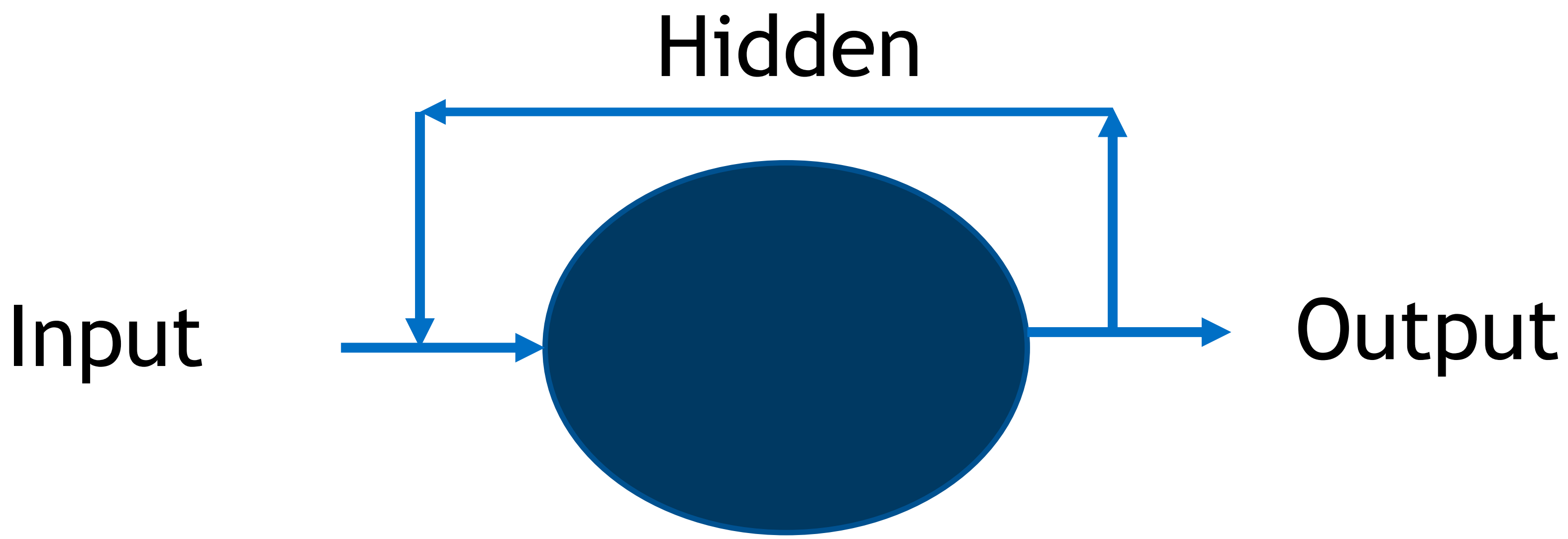
The Dataset: BB8 CAN flow

Timestamp	MessageID	Length	PAYLOAD								
			BYTE 0	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5	BYTE 6	BYTE 7	
574165791302335	101	8	143	4	140	4	160	4	155	4	W-Speed
574165791302421	102	8	3	254	55	254	15	254	15	254	SUSPENSION
574165791302432	103	4	1	0	252	255	0	0	0	0	ROLL&YAW
574165791302441	104	6	223	255	247	255	223	3	0	0	ACCELERATION

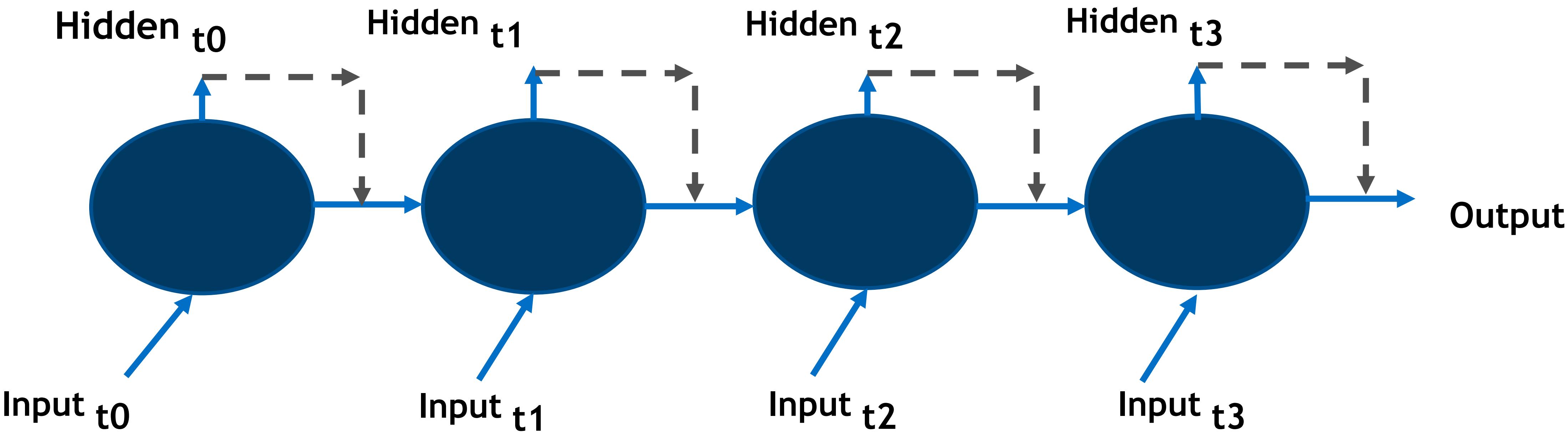
Constraints	Solutions
Multiple ECUs on the CAN BUS	Message ID Selection
Unstructured Data	Content Extraction
Power/Performance	Recurrent Neural Networks (RNNs)


Security
Solution
CAN Anomaly Detector
CAN BUS
CAN
Firewall
Policy
Handler
Message ID
selector &
Content
Extractor
1st NNs
Errors
2nd NN
Output: Probability
of
an attack
Contextual Anomaly
Detection
Stage 2
Detection

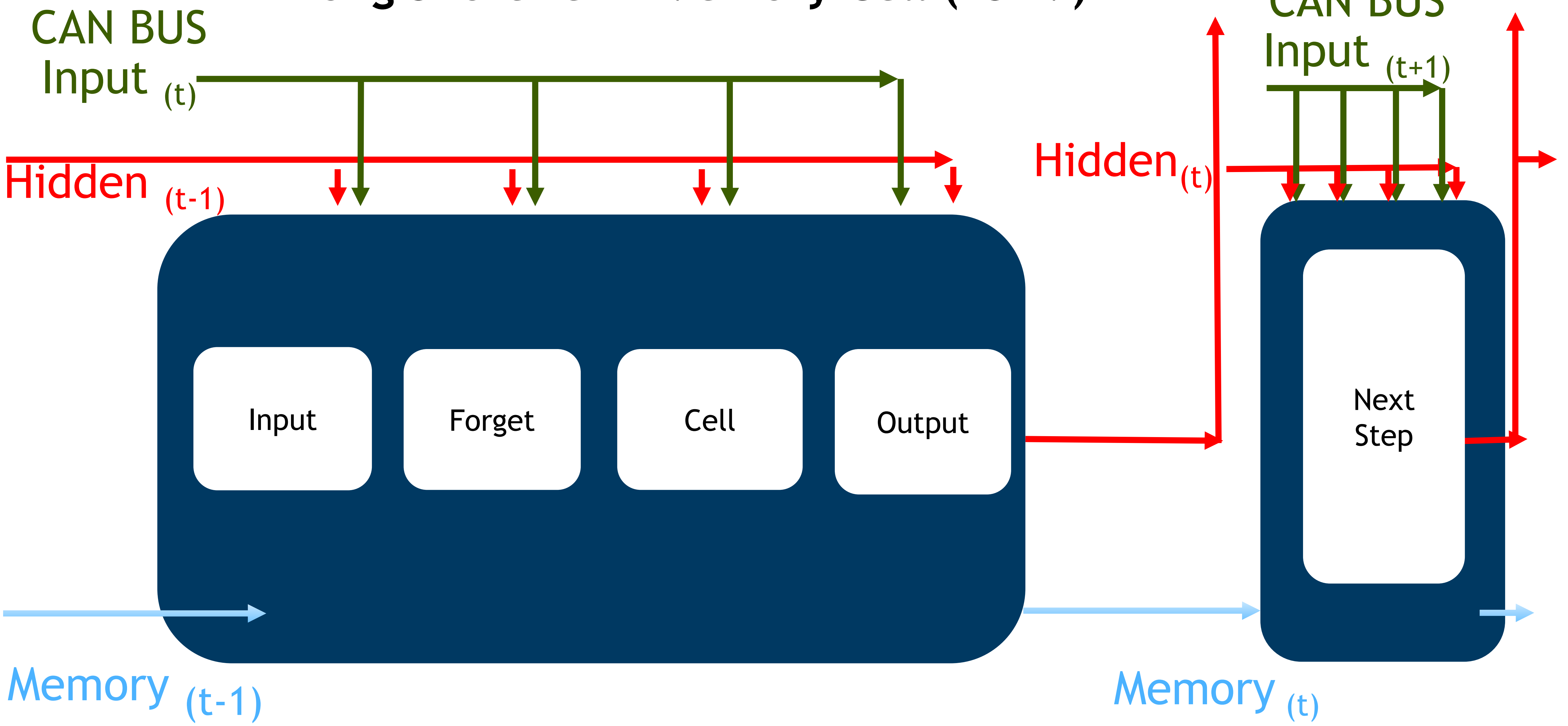
Recurrent Neural Network (RNN)



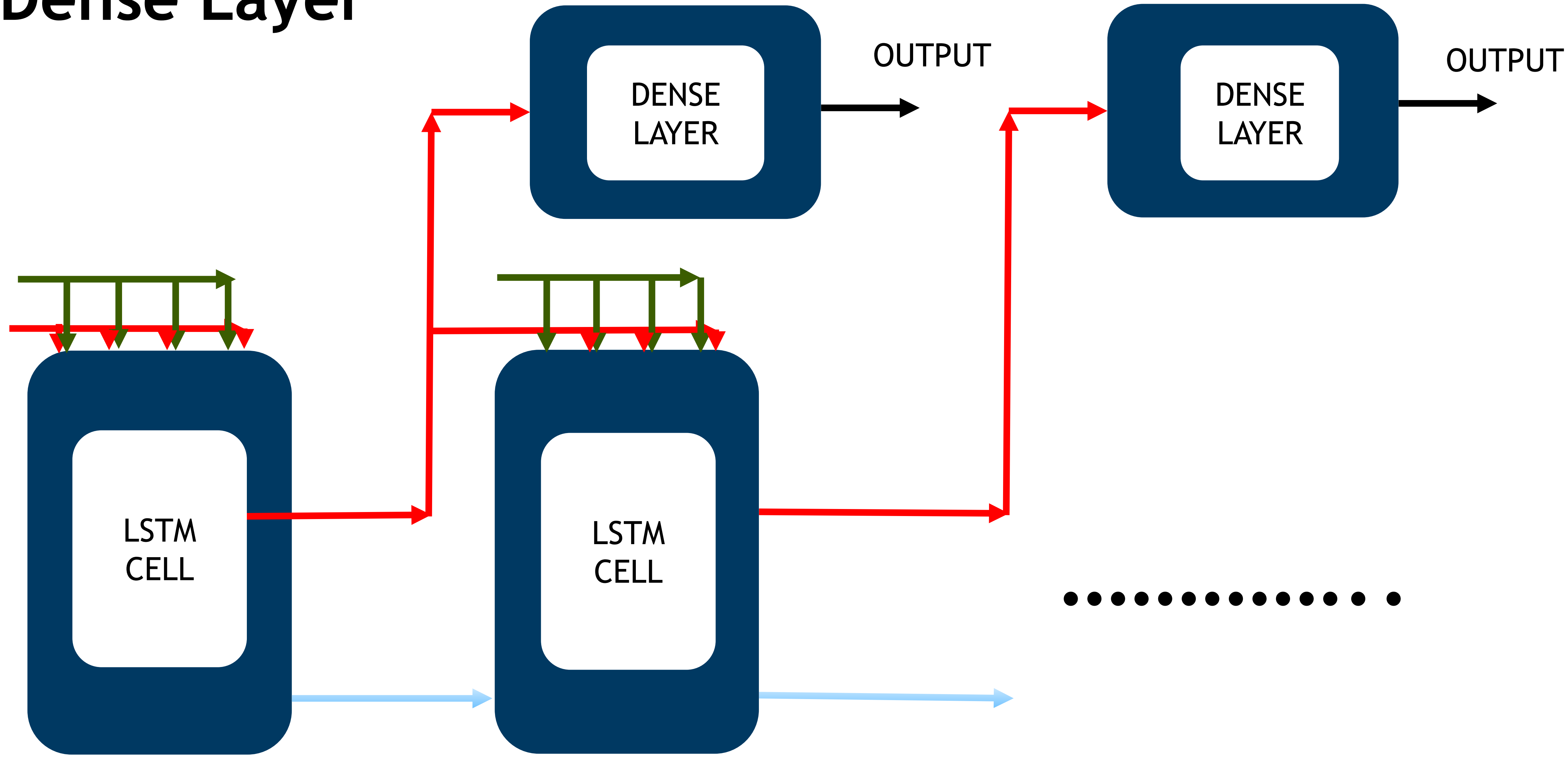
Recurrent Neural Network (RNN)



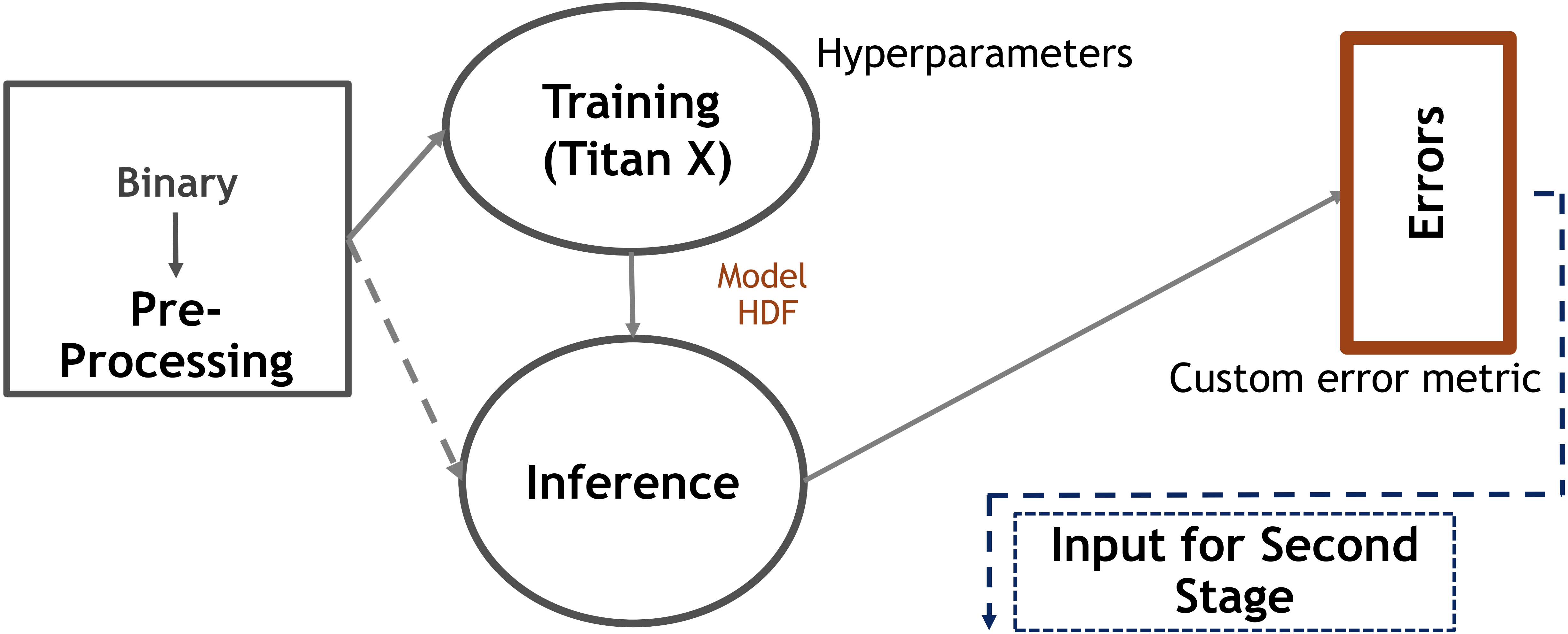
Long Short Term Memory Cell (LSTM)



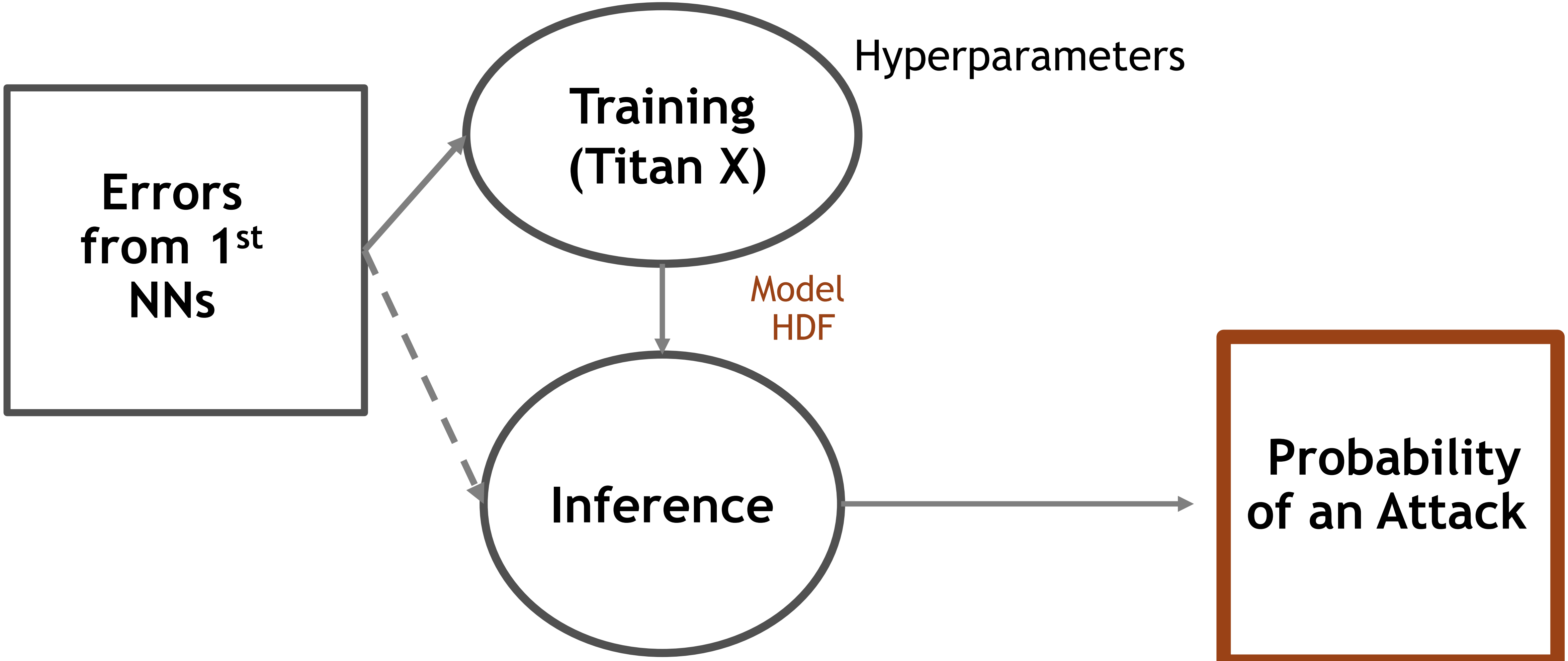
Dense Layer



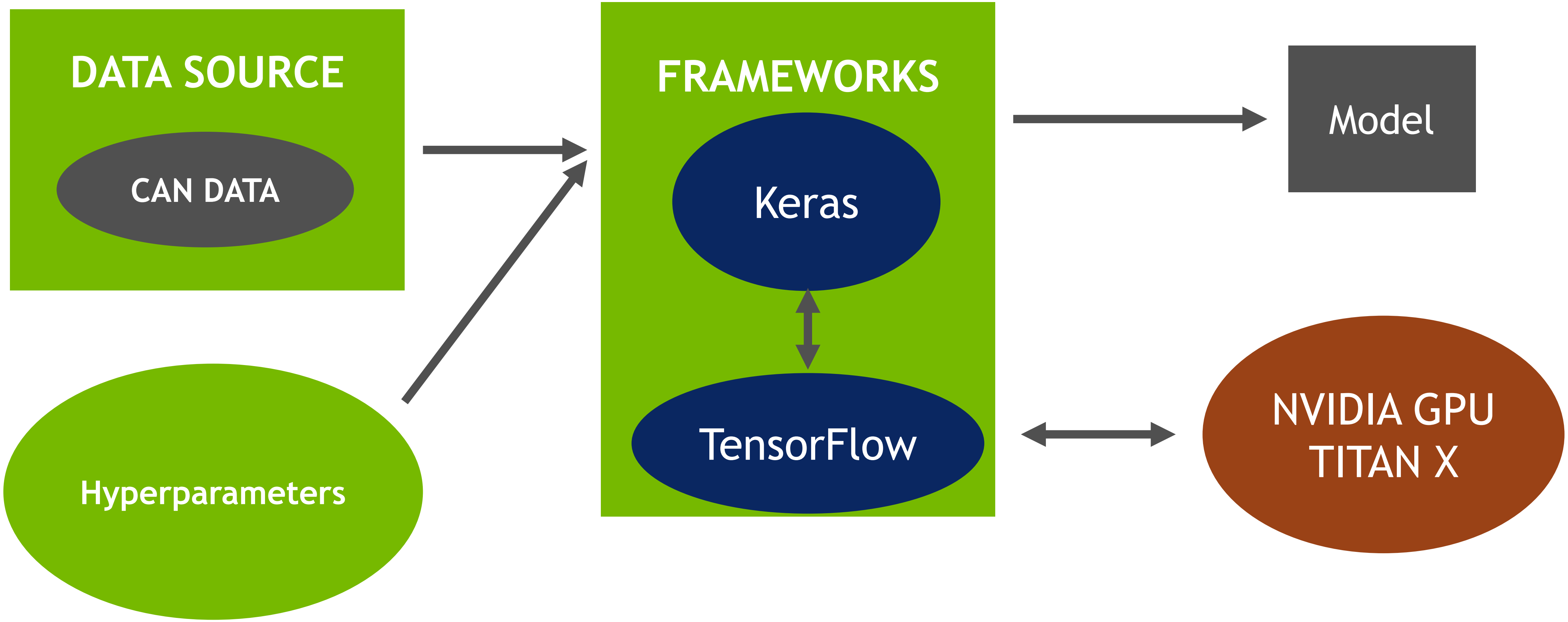
Contextual Anomaly Detection Work Flow



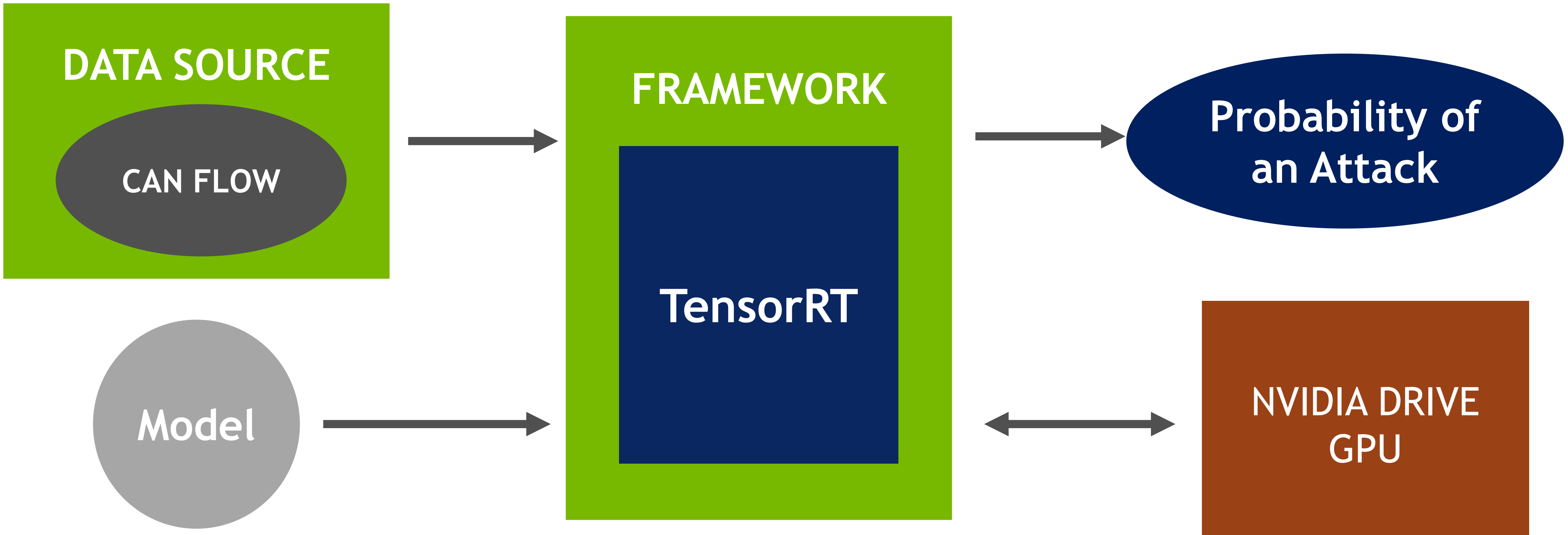
Contextual Anomaly Detection Work Flow-2nd Stage



Training Architecture



Production Architecture



Model Evaluation

Using Sensitivity & Specificity

True Positives (Anomalies) caught

True Negatives allowed

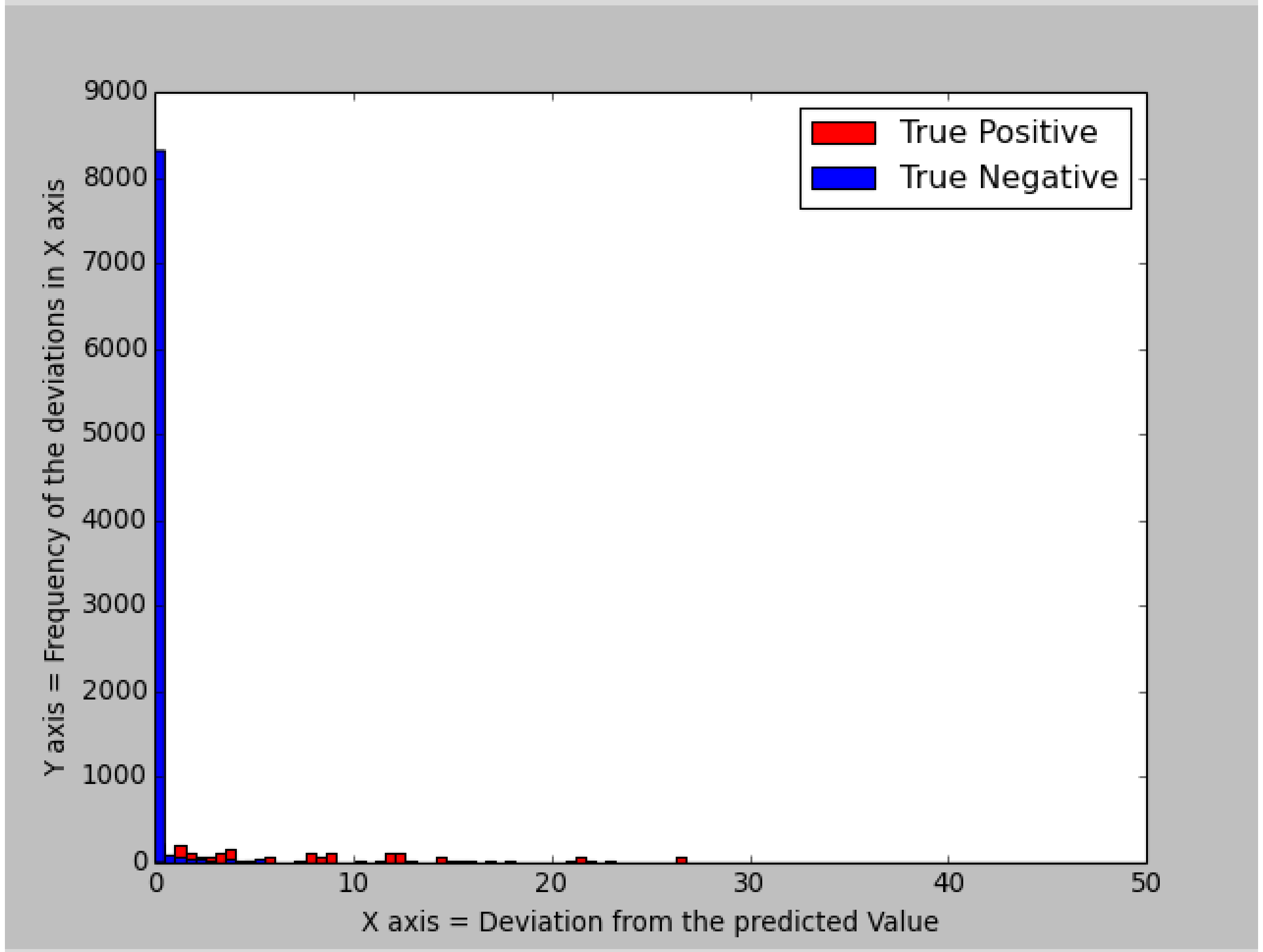


Figure 3. Histogram - Error values output by the 2nd NN

X axis: Deviation

Y axis: Frequency of errors

Median of Positives: 7.82

Median of Negatives: 0.04

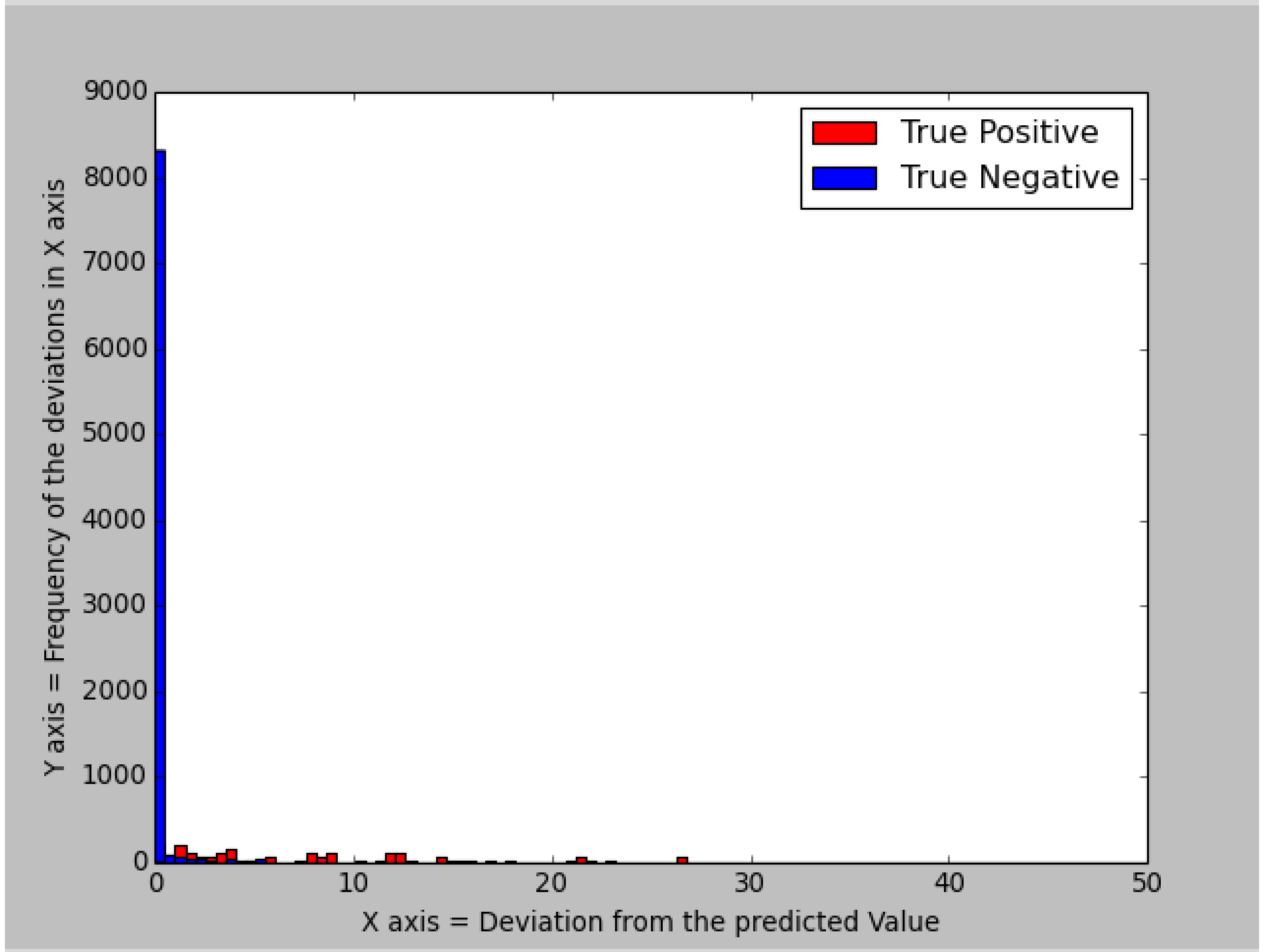


Figure 4. Histogram - Error values output by the 2nd NN

X axis: Deviation

Y axis: Frequency of errors

- Sensitivity: 0.87
- Specificity: 0.94

Results Per Attack Type

Injection attacks
Total: 37
Detected: 32

Replay attacks
Total: 42
Detected: 37

Conclusion

A wall between Autonomous-Driving Software and the unsecured CAN-BUS

- Low inference computational cost
- Fast response
- Offline training
- Future Work

THANK YOU

QUESTIONS?

References

[1] Ivan Studnia, Vincent Nicomette, Eric Alata, Yves Deswarte, Mohamed Kaâniche, Youssef Laarouchi
Survey on security threats and protection mechanisms in embedded automotive networks
Retrieved: https://hal.archives-ouvertes.fr/hal-01176042/document

[2] Automotive Security Best Practices
Retrieved: http://www.mbedlabs.com/2016/01/automotive-can-bus-system-explained.html

[3] Sasan Jafarnejad, Lara Codeca, Walter Bronzi, Raphael Frank, Thomas Engel
A Car Hacking Experiment: When Connectivity meets Vulnerability

[4] Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and Stefan Savage
Comprehensive Experimental Analyses of Automotive Attack Surfaces
Retrieved: http://www.autosec.org/pubs/cars-usenixsec2011.pdf

[5] Automtive CAN Bus System Explained
Retrieved: http://www.mbedlabs.com/2016/01/automotive-can-bus-system-explained.html

[6] Charlie Miller, Chris Valasek. Adventures in Automotive Networks and Control Units
Retrieved: http://illmatics.com/car_hacking.pdf

[7] Varun Chandola, Arindam Banarjee, Vipin Kumar
Anomaly Detection: A Survey
Retrieved: http://cucis.ece.northwestern.edu/projects/DMS/publications/AnomalyDetection.pdf

[8] Dhruba K. Bhattacharyya, Jugal Kumar Kalita
Network Anomaly Detection – A machine learning perspective

Images

Figure1. Connections of a modern car

Figure 2. CAN network

Figure 3. Histogram – Error values output by the 2nd NN

Figure 4. Histogram – Error values output by the 2nd NN

APPENDICES

NVIDIA

Equations in a LSTM Cell – without the dense layer.

$$
\begin{aligned}
f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
h_t &= o_t \circ \sigma_h(c_t)
\end{aligned}
$$