Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

GENERATIVE LEARNING FOR SIMULATION OF VEHICLE FAULTS

Patrick Kuiper[1], Sirui Lin[2], Jose Blanchet[2], and Vahid Tarokh[1]

[1]Dept. of Electrical and Computer Eng., Duke University, Durham, NC, USA
[2]Dept. of Management Science and Eng., Stanford University, Stanford, CA, USA

ABSTRACT

We develop a novel generative model to simulate vehicle health and forecast faults, conditioned on practical operational considerations. The model, trained on data from the US Army's Predictive Logistics program, aims to support predictive maintenance. It forecasts faults far enough in advance to execute a maintenance intervention before a breakdown occurs. The model incorporates real-world factors that affect vehicle health. It also allows us to understand the vehicle's condition by analyzing operating data, and characterizing each vehicle into discrete states. Importantly, the model predicts the time to first fault with high accuracy. We compare its performance to other models and demonstrate its successful training.

1 INTRODUCTION

Many organizations have realized the utility of using machine learning and a data driven approach to understand and improve the health of their vehicle fleets; these organizations invest enormous amounts of resources in transportation systems as they are critical to operations. We focus this analysis on the United States' Department of Defense (DoD), where the US Army alone is projected to spend an estimated $5 billion per year (in 2020 dollar terms through 2050), developing and acquiring ground vehicles, where ground vehicles are any vehicles other than aircraft and ships (CBO 2021). Maintaining this enormous investment is critical to ensuring combat readiness across the DoD, where the department spent $90 billion in 2022 on maintaining vehicles across domains: ground, air, and sea (GAO 2022).

Predicting requirements is critical to an effective maintenance program. The application of statistics towards vehicle maintenance prediction is often referred to as predictive maintenance. Recognizing the importance of predictive maintenance, in the 2022 National Defense Authorization Act (NDAA) Congress required the DoD Inspector General Office to review predictive maintenance practices, originally established by DoD directives in 2002 and 2007 (DoDIG 2023). Given the importance of predictive maintenance in promoting readiness and conserving budgetary resources, the US Army is employing hardware and software solutions to address Congressional and Departmental guidance.

We propose a model designed to achieve *predictive maintenance*, with the goal of completing corrective actions on a vehicle in order to avoid more costly or catastrophic damage in the future (Theissler et al. 2021). This predictive maintenance model is *data driven*, where conclusions are based on the outcomes observed in a dataset of sensor and fault signals recorded from an onboard vehicle computer. This data was collected by the US Army under an effort designated as Condition Based Maintenance (CBM). In this effort millions of time indexed data points were collected from a number of military vehicle types, with the goal of informing the development of maintenance processes, hardware, and analysis software to support predictive maintenance. We have developed a generative learning approach trained on a publicly releasable subset of this CBM data to predict future vehicle faults. Additionally, we simulate outcomes conditioned on practical vehicle considerations, including vehicle age, location, and mode sub-type.

We begin the discussion of our model with a brief literature review, introducing general time-series modeling, then focusing on prediction models for vehicle faults, and relevant technical reports related to the maintenance of military vehicles. Next, we review an analysis related to the DoD's efforts with predictive

979-8-3315-3420-2/24/$31.00 ©2024

maintenance and the CBM dataset used to develop our model. We then introduce the proposed generative fault prediction model. We conclude with results, providing prediction performance analysis, comparison to alternate models, and interpretation of the analysis provided by the proposed model. Additional information regarding model training convergence and parameter interpretation are provided in referenced appendices, including a link to a github repository with all modeling code and processor information in Appendix B of the full version of this paper (Kuiper et al. 2024). In summary, our contributions include:

1. We develop and train a practical, generative learning model to accurately predict vehicle faults and time to first fault on a rolling basis in real time.
2. We generate future sensor feature covariates based on the values of previous covariates, using an attention based transformer.
3. We also generate future sensor feature covariates based on the values of previous data and a selected vehicle use profile, where the profile is based upon practical considerations, using a Variational Auto-Encoder and K-means clustering.
4. We provide results and conclusions which are interpretable and understandable among professional practitioners in the application field.

2 RELATED WORK

Time series analysis, which leverages time-indexed data to extract meaningful statistical conclusions, remains an important field in statistics with applications in finance, medicine, and manufacturing. The wide adoption of machine learning techniques, specifically generative learning, has led to the development of a number of innovative time series methods. We discuss these techniques, along with their application to reliability engineering in automobiles.

2.1 General Time Series Prediction

A number of effective, well established methods exist to perform time series analysis; these include Auto-Regressive (AR) and Moving Average (MA) models, in addition to more advanced integration of both AR and MA, with Auto-Regressive Moving Average (ARMA) and Auto-Regressive Integrated Moving Average (ARIMA) (Box and Jenkins 1976). Approaches employing neural networks specifically pertaining to the forecasting of time-indexed sequence data have included Recurrent Neural Networks (RNNs) and Long-Short Term Memory networks (LSTMs). Of particular interest to this analysis is the Deep-Auto Regressive (DeepAR) method developed by Flunkert et al. (2017). The DeepAR method employs a LSTM-based neural network design, with an AR component, that learns from multiple time series data points. In the proposed model, multiple time series data includes the fault target and sensor feature covariates.

The DeepAR method produces a probabilistic function via Monte Carlo simulation, allowing for the model to be queried on quantiles, which then generates forecasted future data. This enables the generation of multiple future predictions via the quantile function, improving the utility of the analysis when compared to point estimates. A critical consideration is that the DeepAR model requires the incorporation of *future* covariates when forecasting a future target variable. A novel approach for generating multivariate time series predictions is proposed by Gangopadhyay et al. (2020), with the use of a spatio-temporal attention mechanism (STAM). This model isolates critical time steps (temporal) and variables (spatial) when forecasting future target predictions. It does not depend on future covariates during the prediction step as with DeepAR; however, we found STAM alone does not perform as well on forecasting vehicle faults when compared to the DeepAR model (see Section 5 for comparison). We propose a hybrid model using both STAM and DeepAR, where the requirement of the DeepAR model to incorporate future covariates is addressed by generating these sensor values using STAM. We develop this approach further in the Section 4 with additional discussion in Section 5.4.

2.2 Vehicle Fault Prediction

Using generative learning to anticipate faults in vehicles remains a relatively new topic. Recently, Hojjati et al. (2023) completed an analysis similar to our proposed application, using self-supervised learning and a graph based technique to forecast vehicle faults. The formulation of the technique developed by Hojjati et al. (2023) differ significantly to our proposed method, and the dataset is much smaller, with the faults being labeled by experts visually reviewing sensor signals. This is in contrast to our analysis which leverages a much larger dataset collected in non-experimental, field conditions, where the faults are labeled by an automated vehicle onboard computer.

Another similar analysis is provided by Shafi et al. (2018), where several discriminative classification methods are evaluated for determining the occurrence of vehicle faults in several different subsystems. The evaluated classification model takes real-time vehicle data in the form of sensor data and Diagnostic Trouble Codes (DTC), with dimensional reduction using Principal Component Analysis (PCA). The experiments proposed in this analysis are similar to our proposal in that data was collected from dozens of vehicles of a single vehicle family (make / model), with the goal of forecasting faults in real time using machine learning. However, the model only produces a discriminative classification, not a generative prediction, and no interpretable information is provided about the causes of vehicle failure from this analysis.

A general review of machine learning applications for vehicle predictive maintenance is proposed by Theissler et al. (2021). Of note from this rigorous review and categorization of related work is the proposal of several research challenges in the field. The authors highlight the need for *public real-world datasets*, which are *labeled*. An additional relevant challenge referenced in this work is the use of more complex, *truly deep neural networks* for modeling, while maintaining *interpretability* and *explainability* given the need for understanding when addressing maintenance concerns. We work to address all of these research challenges in this proposal and provide a practical model for a large family of vehicles in the US Army, which may be readily applied to additional vehicle types.

2.3 Military Maintenance Background

An detailed report of the CBM dataset, along with a number of other predictive maintenance efforts established by the US Navy and Marine Corps, is provided by Thurston et al. (2019). This report provides a wealth of technical detail concerning the vehicle population covered in the CBM dataset, along with the entire DoD predictive maintenance effort across the uniformed services.

As a check on the CBM dataset, we reference Kays et al. (1998) who developed an insightful analysis into historical operational readiness rates of several common US Army vehicles. This analysis is relevant as it describes with precision common operational readiness rates observed among organizations in the US Army. These published readiness rates serve as a validation for observed rates recorded in the CBM dataset used in the proposed analysis.

3 CBM DATASET OVERVIEW

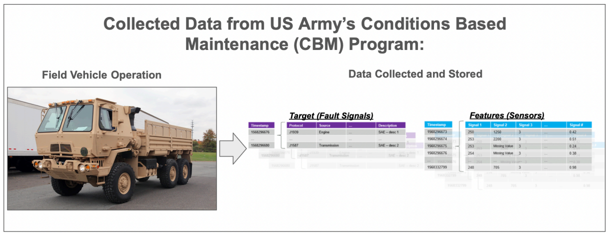
We have identified a research demand for a real-world dataset composed of vehicle sensor readings, with associated vehicle faults. Such a dataset would allow for the development of a data driven fault prediction model. This dataset would be of particular use in military and civilian automotive applications. We now provide further details about the US Army CBM dataset which we propose meets the challenges presented by the predictive maintenance research community, and further develop our assumptions that vehicle electronic fault signals may be used as a proxy for vehicle condition.

3.1 US Military Predictive Maintenance Data Collection Efforts

According to the Government Accountability Office (GAO) report titled "Military Readiness: Actions Needed to Further Predictive Maintenance on Weapon Sytems", published in December 2022 (GAO 2022), the US DOD's predictive maintenance efforts may be summarized as

> ...any effort that uses condition-monitoring technology or analysis of historical data to anticipate maintenance needs in a manner that reduces unscheduled reactive maintenance or overly prescriptive preventive maintenance. The military services use multiple terms for predictive maintenance or predictive maintenance enablers, including Condition-Based Maintenance (CBM), Condition-Based Maintenance Plus (CBM+), Prognostic and Predictive Maintenance (PPMx), Enhanced Reliability-Centered Maintenance (eRCM), and Predictive Maintenance.

The dataset used in this analysis is from the US Army's CBM effort. As illustrated in Figure 1, sensor and fault data was recorded over a period of months from a number of vehicle types. As referenced in this GAO report, billions of dollars are spent and military readiness is degraded due to unplanned maintenance associated with US weapons ("weapons" includes vehicles). This degradation of resources and readiness may be significantly reduced through the use of effective predictive maintenance, as dictated in the 2002 DOD predictive maintenance policy.



Collected Data from US Army's Conditions Based Maintenance (CBM) Program:

Target (Fault Signals)

Timestamp	Protocol	Source	...	Description
1568296676	J1939	Engine	...	SAE – desc 1
...	...	...	...	...
1568296680	J1587	Transmission	...	SAE – desc 2

Features (Sensors)

Timestamp	Signal 1	Signal 2	Signal 3	...	Signal #
1568296673	250	1250	3	...	0.42
1568296674	253	2200	3	...	0.51
1568296675	253	Missing Value	3	...	0.24
1568296676	254	Missing Value	3	...	0.38
...	...	...	...	...	...
1568332799	248	705	3	...	0.98

Figure 1: Description of Condition Based Maintenance dataset. Data from field vehicle operations, collected over several months from various vehicle types, are presented. This dataset includes time series of various fault signals, each with associated recording timestamps, as well as time series from multiple sensors also timestamped. Our analysis aims to utilize the sensor data as feature covariates to predict fault occurrences.

3.2 Fault Signals and US Army Vehicles

Vehicle fault targets in this analysis are recorded in accordance with the Society of Automotive Engineers (SAE) J1939 and J1708 data specifications (Thurston et al. 2019). It is important to note that the analysis we propose relies on these *fault signals*, also known as Diagnostic Trouble Codes (DTC), as a target and not an observation of vehicle failure. This is because failure data is not available in the analyzed set. The assumption that vehicle fault signals or DTC data is highly correlated with actual failure is supported by the detailed work regarding the CBM dataset completed by Thurston et al. (2019) as described directly below:

> DTCs can be very useful for development and enhancement of PHM (Predictive Health Monitoring) capabilities, because they contain the ground truth information for impending failures. In fact, they represent a level of condition monitoring that is already implemented by electronic control units (ECUs), employing traditionally conservative thresholds to reduce false alarm problems.

While true vehicle failure would be an informative additional covariate, the goal of predictive maintenance is to take action based on vehicle signals before failure, where failures have high correlation with fault signals, as these signals measure the true status of critical vehicle systems (i.e. engine, transmission, and brakes).

4 GENERATIVE FAULT PREDICTION MODEL DEFINITION

In this section we propose a target distribution of vehicle faults conditioned on vehicle sensor values, which would provide significant utility towards predictive maintenance. We formulate a model for this target distribution employing generative learning. We fully specify this generative learning model and provide details concerning training and experimentation. The proposed model is designed to predict if a vehicle will experience a fault and time to first fault, given a fault is predicted to occur.

4.1 Target Distribution Definition

The ultimate goal of this effort is to define the most accurate probability distribution, shown in Equation (1), where $(z_t, t \geq 0)$ is the fault *target*, or fault indicator time series, and $(x_{i,t}, 1 \leq i \leq 38, t \geq 0)$ are the sensor *features* or covariate time series. Here t is defined as a time point and i is the label of sensor feature covariates, where there are 38 total sensor covariates in the CBM dataset considered. With this target distribution, a vehicle operator or maintainer would know precisely the probability of *if* a vehicle is to experience a fault from time point t_0 to t_0+T, or $t = [t_0 : t_0+T]$, and *when* the fault would occur. We propose this information will allow for vehicle operators and maintainers to significantly improve predictive maintenance. Below we define this conditional probability distribution, which we name the target distribution:

$$P(z^k_{t_0:t_0+T} | z^k_{0:t_0-1}, (x^k_{i,0:t_0-1}, 1 \leq i \leq 38)) \; \forall k \in K \; \forall t_0 \in S, \tag{1}$$

where, $z^k_{t_0:t_0+T}$ represents the time series of fault signals collected from vehicle k from time point t_0 to t_0+T, $x^k_{i,t_0:t_0-1}$ represents the time series of sensor feature i collected from vehicle k from time point 0 to t_0-1, and for model training and experimental purposes, the set S comprises randomly selected start times for each vehicle k in K, where K is the set of selected vehicles for analysis.

4.2 Generative Model Definition

While ideally we could formulate the probability distribution in Equation (1) explicitly, in practice this is not feasible so we employ a generative model to forecast outcomes given the data collected, and subsequently employ these results to model the target distribution. To accomplish this, we employ a hybrid of generative models, using DeepAR in conjunction with STAM (see Section 2.1) and a VAE. The complete prediction process is described below:

- **Step 1:** We train a DeepAR model (Q_θ) on a large (relative to forecast length) block of vehicle sensor feature and fault target data.
- **Step 2:** Given time point t_0 to begin our forecasting period, we train the STAM model (g_θ) and a VAE (v_θ) to generate estimates of future sensor feature covariates.
- **Step 3:** We generate future targets from times t_0 to t_0+T using the previously trained DeepAR model (Q_θ), and covariates generated from both the STAM model (g_θ) and the VAE model (v_θ).
- **Step 4:** We use classifier (C - see Equation (3)) and regression model (R - see Equation (4)) to determine if there is a fault during the forecasted time period $[t_0 : t_0+T]$, and what the time to first fault is, given a fault is expected to occur.

The complete process of model training and experimentation on a single vehicle dataset is shown in Figure 2, with each step identified. We note that each vehicle $k \in K$ requires the training of a DeepAR model, Q_θ, and each forecast $t_0 \in S$ requires the training of a STAM model, g_θ, potentially with the evaluation of the VAE model, v_θ. We will further explain this process in the proceeding subsections.

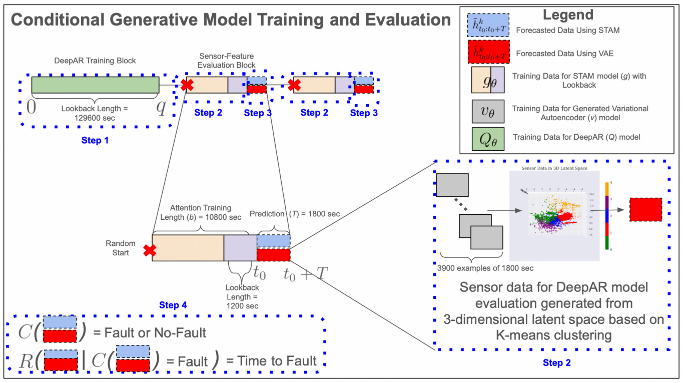

Conditional Generative Model Training and Evaluation
Legend
$\tilde{h}^k_{t_0:t_0+T}$ Forecasted Data Using STAM
$\hat{h}^k_{t_0:t_0+T}$ Forecasted Data Using VAE
g_θ Training Data for STAM model (g) with Lookback
v_θ Training Data for Generated Variational Autoencoder (v) model
Q_θ Training Data for DeepAR (Q) model
DeepAR Training Block
Sensor-Feature Evaluation Block
0
Lookback Length = 129600 sec
q
Step 1
Step 2
Step 3
Step 2
Step 3
Attention Training Length (b) = 10800 sec
Prediction (T) = 1800 sec
Random Start
t_0
$t_0 + T$
Lookback Length = 1200 sec
Step 4
$C(\;)$ = Fault or No-Fault
$R(\;|\;C(\;)$ = Fault $)$ = Time to Fault
Sensor Data in 3D Latent Space
3900 examples of 1800 sec
Sensor data for DeepAR model evaluation generated from 3-dimensional latent space based on K-means clustering
Step 2

Figure 2: Overview of generative model for conditional vehicle fault prediction on vehicle $k \in K$. At Step 1, a DeepAR model (Q_θ) is trained using time series of fault target and sensor feature values from time point 0 to q, and is subsequently evaluated at randomly selected time periods, with $t_0 \in S$. At Step 2, a STAM model (g_θ) is trained over a period of $b = 3$ hours and a VAE model (v_θ) is trained on out of sample data. At Step 3, the DeepAR model (Q_θ) generates representations of the fault indicator $\tilde{h}^k_{t_0:t_0+T}$ (resp. $\hat{h}^k_{t_0:t_0+T}$) of $T = 30$ minutes using the sensor feature covariates generated from the STAM model (resp. the VAE model), where STAM utilizes a lookback length of 20 minutes. At Step 4, the classifier (C) and the regression model (R) use the predictions from both the STAM and VAE model to forecast if there is a fault and what time the first fault is (given a fault is predicted) from time point t_0 to $t_0 + T$.

4.3 Spatio-Temporal Attention Mechanism (STAM) and Variational Auto Encoder (VAE) for Feature Generation

As discussed in Step 1, in order to develop Equation (1), we reference the DeepAR model, Q_θ and define a lookback parameter of length ℓ and Spatio-Temporal Attention Mechanism (STAM) training length b. It is critical to note that to achieve reasonable performance with the DeepAR model, *features* must be available during the prediction time period $t = [t_0 : t_0 + T]$, where t_0 is the prediction start time ($t_0 \in S$ for each vehicle k in K) and $t_0 + T$ the end time for the selected forecast. The requirement of features during the prediction time poses a problem during practical forecasting because this information is *future sensor* data which is not available at prediction time. To meet this requirement, we produce sensor feature data during the prediction time period using two generative methods:

1. $g_\theta \equiv$ *spatio-temporal attention mechanism (STAM)*, where the STAM is trained on recent data: $g_\theta(x^k_{i,t_0-b:t_0-1}) \Rightarrow \tilde{x}^k_{i,t_0:t_0+T}$. This model captures the vehicle's most recent operational history.
2. $v_\theta \equiv$ *variational auto-encoder (VAE)*, where the VAE is trained on out of sample data: $v_\theta(\hat{x}^{k'}_{i,t'_0:t'_0+T}) \Rightarrow \hat{x}^k_{i,t_0:t_0+T}$ such that $t'_0 \in S'$ and $k' \in K'$, and $k' \notin K$. This model captures how the vehicle will be operated in the future, which is either based upon the true data, $v_\theta(x^k_{i,t_0:t_0+T})$, or conditionally selected based on latent space cluster state, $v_\theta(\hat{x}^{k'}_{i,t'_0:t'_0+T})$, to *simulate* actual vehicle operating conditions.

In the prediction process, with the generated sensor data from both the STAM and VAE generators, we propose calculating two *representations* of the fault indicator future time series:

$$Q_\theta(z^k_{t_0:t_0+T} | z^k_{0:q}, x^k_{i,0:q}, \tilde{x}^k_{i,t_0:t_0+T}) \Rightarrow \tilde{h}^k_{t_0:t_0+T} \tag{2a}$$

$$Q_\theta(z^k_{t_0:t_0+T} | z^k_{0:q}, x^k_{i,0:q}, \hat{x}^k_{i,t_0:t_0+T}) \Rightarrow \hat{h}^k_{t_0:t_0+T} \tag{2b}$$

where $\tilde{h}^k_{t_0:t_0+T}$ and $\hat{h}^k_{t_0:t_0+T}$, are hidden representations of the target distribution of $z^k_{t_0:t_0+T}$. We call these values hidden representations because they are used in the classification and regression models, $C(\tilde{h}^k_{t_0:t_0+T}, \hat{h}^k_{t_0:t_0+T}) = c \in \{0,1\}$ and $R(\tilde{h}^k_{t_0:t_0+T} | C(\tilde{h}^k_{t_0:T}, \hat{h}^k_{t_0:t_0+T}) = 1) = f \in \mathbb{R}^+$, shown in Equations (3) and (4), where the regression model R is not trained with the VAE generated hidden representation, $\hat{h}^k_{t_0:t_0+T}$. Here the predicted value from the regression, f, is defined as the time to *first* fault observed over the forecasted time period. We focus on predicting the time to first fault, as it has the most practical implications with respect to vehicle condition and function: the first fault is most likely to cause a maintenance event affecting operation. The classification and regression models take in the hidden representation data given each type of input and predicts fault or no-fault, and time to first fault. We employ the random forest classification and random forest regression models for C and R, respectively. These models were selected based upon their performance and potential for interpretability (Ho 1995). Additionally, the parameters of the DeepAR model, θ, are identical for the evaluation of Q when used with generated covariates $\tilde{x}^k_{i,t_0:t_0+T}$ and $\hat{x}^k_{i,t_0:t_0+T}$.

$$C(\tilde{h}^k_{t_0:t_0+T}, \hat{h}^k_{t_0:t_0+T}) = c \in \{0,1\}. \tag{3}$$

$$R(\tilde{h}^k_{t_0:t_0+T} | C(\tilde{h}^k_{t_0:t_0+T}, \hat{h}^k_{t_0:t_0+T}) = 1) = f \in \mathbb{R}^+. \tag{4}$$

We note that when training the classification model C, v_θ is given the *true* future covariates for encoding and decoding, using the trained VAE: $v_\theta(x^k_{i,t_0:t_0+T}) \Rightarrow \hat{x}^k_{i,t_0:t_0+T}$. This is to ensure an accurate fault classification is given when conditioning on selected latent cluster states. While in practice the *true* future sensor covariate timer series, $x^k_{i,t_0:t_0+T}$, would not be available for the forecast, the information which characterizes the vehicle latent cluster *state* would be available. Using the VAE model, we would be able to generate future fault covariates from the true vehicle state, where covariates from the common states are known to be close in the latent space.

Next, in Section 4.4, we will discuss the random selection and generation of sensor data from the trained VAE latent space, and subsequent simulation of fault probabilities conditioned on this state. Finally, we observe that with the output of Equations (3) and (4), we may produce a prediction for $z^{\star k}_{t_0:t_0+T}$, or a final time series forecast indicating exactly when an *initial* or first fault will occur. We propose that this is critically important information for predictive maintenance.

4.4 Simulation using Generative Modeling Conditioned on Vehicle State

As illustrated in Figure 3, the training of a VAE, v_θ, on *out of sample* vehicle data $v_\theta(\hat{x}^{k'}_{i,t'_0:t'_0+T}) \Rightarrow \hat{x}^k_{i,t_0:t_0+T}$ such that $t'_0 \in S'$ and $k' \in K'$, and $k' \notin K$ allows us to define latent states, in a reduced dimension, which represent practical considerations associated with vehicles. These practical considerations could include: vehicle age, maintenance condition, location, etc. We enumerate the practical considerations present in in the CBM dataset and discuss their impact on the vehicle latent states in Section 5.1. These states are designated by clustering on the reduced latent dimension produced by the encoding step of the VAE.

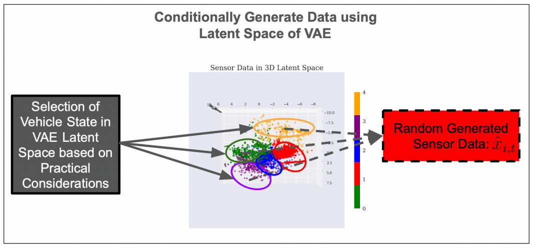

Conditionally Generate Data using
Latent Space of VAE
Sensor Data in 3D Latent Space
Selection of Vehicle State in VAE Latent Space based on Practical Considerations
Random Generated Sensor Data: $\hat{x}_{i,t}$
0
1
2
3
4

Figure 3: Illustration of mapping vehicle operational condition to generate future sensor readings.

In order to simulate vehicle conditions, which were not observed in the actual data, we may randomly sample data points from the selected state, decode the random point to generate sensor data, $\hat{x}^{k'}_{i,t'_0:t'_0+T}$. We may then use this data in the DeepAR model and generate the hidden representation $\hat{h}^{k'}_{t'_0:t'_0+T}$, with the simulated sensor values. This will produce alternate values for the classification of vehicle faults and time to first fault, using models C and R respectively, based on practical considerations.

5 RESULTS

We develop our results addressing two topics. First, what does the clustering of vehicle sensor data, discussed in Section 4.4, practically indicate with respect to vehicle conditions. Second, how accurate is the discriminative model described in Section 4.1 at predicting if there will be a vehicle fault observed during the generated time period, and how accurately can we predict when the first fault will occur. We provide results allowing for model interpretation, and comparison to alternate generative model formulations. We discuss the interpretation of model parameters and convergence analysis regarding the training of the DeepAR model in Appendices A and C (resp.) of the full version of this paper (Kuiper et al. 2024).

5.1 Simulation using Latent Space Representation and K-means Clustering

In Figure 4, we provide a representation of vehicle out of sample sensor data in a three dimensional latent space. Here the trained VAE, v_θ, maps historical sensor readings $\hat{x}^{k'}_{i,t'_0:t'_0+T}$, such that $t'_0 \in S'$ and $k' \in K'$, into the latent representation shown. See Section 4 for definitions of these variables. Subsequently, the K-means clustering algorithm is applied to the latent representation, producing five clusters.

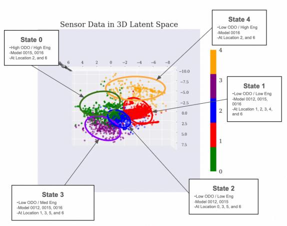


Figure 4: K-means clusters with state interpretation.

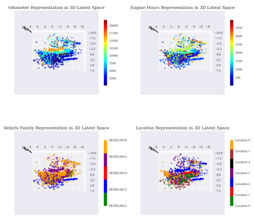


Figure 5: VAE latent representation with metadata.

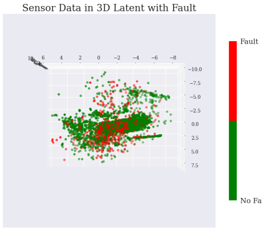


Figure 6: Latent space fault visualization.

State	Fault Rate
State 0	0.080
State 1	0.078
State 2	0.069
State 3	0.111
State 4	0.098

Table 1: Simulation of fault rates of vehicle states.

We referenced metadata with respect to vehicle location, odometer mileage, engine hours, and subfamily and found that the VAE latent state mapping and K-means application identified practical clusters or *states*, numbered 0-4, in which each vehicle operates. Figure 5 provides the latent representation of each of the metadata categories (location, vehicle odometer mileage, engine hours, and vehicle sub-family). This figure shows how the logical clusters or states were interpreted in Figure 4. In Figure 6, we provide another view of the three dimensional latent space, where each point is labeled according to whether the vehicle truly experienced a fault, or did not experience a fault, during the generated time period $t = [t_0 : t_0 + T]$.

As proposed in Section 4, we complete a simulation generating sensor data using the decoding function of the VAE to condition the trained DeepAR model Q_θ when producing $\hat{h}^k_{t_0:t_0+T}$, the VAE generated hidden representation of the true fault indicator function $z^k_{t_0:t_0+T}$. We employ the trained classifier, C, proposed in Equation (3) to determine if a vehicle experiences a fault during a predicted time period. This allows practitioners to generate simulated outcomes conditioned on practical considerations, or "what if" scenarios associated with their vehicles: i.e. what would happen if they changed the location the vehicle operates out of, what happens as the vehicle mileage, or engine age increase, etc. In Table 1 we provide these simulated probabilities. In this simulation, for the same test vehicles and randomly selected time periods used throughout this analysis, we have generated sensor covariates, $\hat{x}^k_{t_0:t_0+T}$, by randomly sampling data points in the VAE latent space from a fixed state, where the members of the state are determined by the K-means model. For example, if all the vehicles ($k \in K$) are assumed to be operating in State 0, we may sample hundreds of points in the latent space only from this fixed state, generate $\hat{x}^k_{t_0:t_0+T}$ using the trained VAE v_θ, and subsequently use this generated data in the DeepAR model Q_θ to produce the hidden representation $\hat{h}^k_{t_0:t_0+T}$, which is then evaluated using the trained classifier C. In this example for State 0, we expect a fault rate of 0.080.

The conditioned fault rates observed in this simulation follow logically given the practical interpretation of vehicle states given in Figure 4. We observe, relative to other states, that the fault rate of States 3 and 4 are the highest, where these are vehicles with medium to high engine hours and lower odometer readings. This may be an indicator of scheduled service requirements for newer vehicles. Additionally, considerations with locations 1 and 3 may weigh on the fault rates as different locations often have varying maintenance programs. As noted in the introduction and formulation of this generative learning model, a great advantage is given by the ability to visualize and interpret results in order to understand why the model is generating specific fault forecasts. Furthermore, the values provided in Table 1 are validated by US Army published standards and observed operational readiness rates of vehicles, cited to be between $80\% - 100\%$ (Kays et al. 1998).

5.2 Fault Prediction

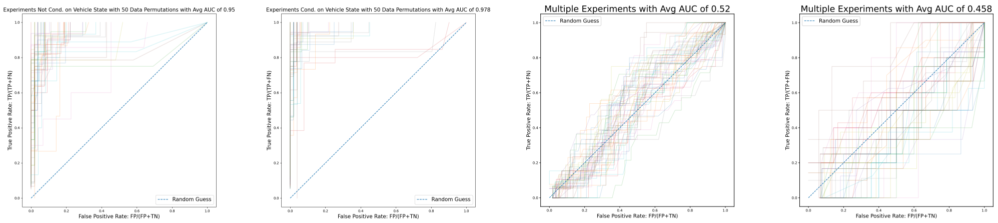
In this section, we discuss the performance of the discriminative classifier proposed in Equation (3), which determines if a fault is going to be forecasted during the generated time period $t = [t_0 : t_0 + T]$. In the models trained and experiments visualized, the time period of prediction is 30 minutes, where $T = 1800\,secs$. To be specific, we consider two cases: 1. using only the STAM generated covariate data, and 2. both the STAM and the VAE generated covariate data. In the latter case, for the VAE we use the actual latent state of the vehicle, by referencing the true future feature covariates. Using the actual latent state facilitates the development of the simulation by mapping the correct latent state to the hidden representation. The latent state is available in practice as the practitioner knows current attributes of the vehicle (vehicle mileage, engine age, etc) and may choose the appropriate state.

For each case, we choose 50 pairs of training and test sets through random permutation of the data. We note that the training and test sets were not split between distinct vehicles, because in practice the regression model will be updated as the vehicle is operated. As shown in Figure 7, for the test set in each of the 50 pairs, we plot the Receiver Operating Characteristics (ROC) curve using a unique color. The average Area Under the Curve (AUC) for the 50 tests are also computed.

These results articulate the highly accurate and steady performance achieved when using: 1. only the STAM generated covariate data, resulting in a 0.950 AUC, and 2. both the STAM and the VAE generated covariate data, resulting in a 0.978 AUC. More specifically, Figure 7a displays the ROC curves when the hidden representation ($\tilde{h}^k_{t_0:t_0+T}$) is generated by the DeepAR model taking in only the STAM generated covariates (see Equation (2a)) and this hidden representation is used in the training and testing of the classifier: $C(\tilde{h}^k_{t_0:t_0+T})$. In Figure 7b, the hidden representations ($\tilde{h}^k_{t_0:t_0+T}$ and $\hat{h}^k_{t_0:t_0+T}$) generated by the DeepAR models using the covariates generated from the STAM and the VAE (see both Equations (2a) and (2b)) are both used in the training and testing of the classifier: $C(\tilde{h}^k_{t_0:t_0+T}, \hat{h}^k_{t_0:t_0+T})$. We observe that the inclusion of both the STAM and VAE generated covariates improves the average AUC performance by 2.8%. This improvement is expected as additional information is provided to the DeepAR (Q_θ) model via the VAE (v_θ), with the actual latent state information. Even without the latent state information, the model performs with satisfactory accuracy (0.950). To further illustrate the advantages of our method in fault prediction, we compare the AUC performance of our method to that of other existing methods in Section 5.4.

5.3 Time to First Fault Prediction

As discussed in Section 4.2, once a fault is determined to occur during the predicted time period, $C(\tilde{h}^k_{t_0:t_0+T}, \hat{h}^k_{t_0:t_0+T}) = 1$, we employ a random forest regression model, $R(\tilde{h}^k_{t_0:t_0+T} | C(\tilde{h}^k_{t_0:t_0+T}, \hat{h}^k_{t_0:t_0+T}) = 1) = f \in \mathbb{R}^+$, to predict the time to first fault (Ho 1995). The regression model R achieves a coefficient



(a) ROC curves using STAM covariates. (b) ROC curves using STAM+VAE covariates. (c) ROC curves of the LSTM-based classifier. (d) ROC curves of the STAM-based classifier.

Figure 7: Figure 6(a), 6(b): ROC curves of our proposed discriminative classifiers. We choose 50 pairs of training and test data by random permutation. We plot the ROC curve on each test data using different colors after training our classifiers on the corresponding training data. We compute the average AUC for the 50 ROC curves. In Figure 6(a), we use the representations of the fault indicator generated by the DeepAR model that takes in only the STAM generated sensor covariates, while in Figure 6(b), we utilize the representations of the fault indicator generated by the DeepAR model that takes in both the STAM and VAE generated sensor covariates. Figure 6(c), 6(d): ROC curves of alternate classification models. We choose 50 pairs of training and test data by random permutation. We plot the ROC curve on each test data using different colors after training our classifiers on the corresponding training data. We compute the average AUC for the 50 ROC curves. In Figure 6(c), we we utilize the representations of the fault indicator directly generated by LSTM, while in Figure 6(d), we we utilize the representations of the fault indicator directly generated by STAM.

of determination (r^2) of 0.77 in predicting time to first fault, where R was trained on the 50th quantile generated results. We define the coefficient of determination as $r^2 = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \bar{y})^2}$, where y_i is the true time to first fault, f_i is the forecasted time to first fault for observation i, and $\bar{y}$ is the true mean. We note that all true time to first fault values are no further than 4 minutes ($240\,secs$) from the forecasted time to first fault; however, these predictions could benefit from further refinement on robustness since the present prediction indicates potential under-estimation of the time to first fault. This issue has practical implications for predictive maintenance and we seek to address this shortcoming in future work.

5.4 Comparison to Alternate Models on Fault Prediction

We selected two methods for comparison to the the proposed model. The selected methods include a LSTM and a STAM model, where both are used to directly generate hidden representations of the target fault indicator. By contrast, our proposed method uses the STAM model to generate sensor covariates, and then inputs the generated sensor covariates into the DeepAR model to generate hidden representations of the target fault indicator.

The time period is set to be $t = [t_0 : t_0 + T]$. As with the proposed methodology, the directly generated hidden representation, denoted as $\tilde{h}^k_{t_0:t_0+T}$, serves as the input into a classification model, $C(\tilde{h}^k_{t_0:t_0+T}) = c \in \{0,1\}$. The performance of the comparison models are visualized through plots of ROC curves in Figure 7. In Table 2, we compare the average AUC for these methods and see clearly that the performance for both the LSTM and STAM models is significantly poorer when compared to the proposed model.

5.5 Discussion

A clear question arises from the proposed model: why does the use of the STAM method to generate covariates, and subsequently generating target values using the DeepAR model produce such an accurate fault representation? This is clearly true when compared to other baseline methods as presented in Figure

Table 2: Model comparison table.

Model for Comparison	**AUC**
LSTM Baseline	0.520
STAM Baseline	0.458
DeepAR with STAM Covariates (ours)	0.950
DeepAR with STAM and VAE Covariates (ours)	**0.978**

7(c)(d). We believe this is due to several attributes of both the DeepAR and STAM models, namely: (i) DeepAR, being auto-regressive and recurrent, is particularly designed to learn seasonal behavior well, which is specifically helpful with larger and longer time series datasets (ii) DeepAR incorporates the use of covariates focused towards learning a specific target distribution, (iii) STAM is demonstrated to be particularly scalable and accurate when forecasting time series values with multiple variables.

Specifically referencing point (i), we know that the vehicle sensor feature and fault target data exhibits seasonality or cyclic behavior when explored over significant periods of time. As such, we trained the DeepAR over 36 hours of data (129600 data points). This allowed for the auto-regressive network to learn sufficient detail of recurrent structure over the fault targets and sensor feature covariates, as specified in point (ii). Additionally, while alone not accurate when predicting target values as shown in Section 5.4, the STAM model is excellent at generating the covariates when trained over modest time periods, such as 3 hours of data in the proposed model (10800 data points). STAM provides sufficient accuracy when forecasting the subsequent 30 minutes (1800 data points) of sensor feature data to produce fault target data which is sufficiently distinguishable to support highly accurate binary classification and regression results. Attributes of the STAM and DeepAR models are combined with appropriate training periods to leverage the strengths of both methods in order to facilitate forecasts that are well suited for this vehicle fault prediction problem.

6 CONCLUSION AND FURTHER WORK

We proposed a modeling approach which allows for the highly accurate prediction of future vehicle faults, and the simulation of future faults conditioned on practical considerations. The simulated results are validated using logical interpretation of vehicle metadata and published documentation. We demonstrated that the model is interpretable and provides actionable results for practitioners in the vehicle operation and maintenance fields. Significant further work remains including:

1. Improving robustness of time to first fault forecast (Section 5.3) with mitigation of underestimation.
2. Extending prediction time to the scale of hours.
3. Simulating the evolution of states for multiple vehicles to streamline fleet maintenance scheduling.

We note that a link to a github repository with all modeling code and processor information is available in Appendix B in the full version of this paper (Kuiper et al. 2024). The work presented represents an initial step in this modeling effort, and we will continue to update our procedures to produce accurate and interpretable results for the predictive maintenance effort.

7 ACKNOWLEDGEMENT

The material in this paper is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-20-1-0397.

REFERENCES

Box, G. E. P. and G. M. Jenkins. 1976. *Time Series Analysis: Forecasting and Control*. Holden-Day.

CBO 2021. "Projected Acquisition Costs for the Army's Ground Combat Vehicles".

DoDIG 2023. "Management Advisory: Maintenance Concerns for the Army's Prepositioned Stock–5 Equipment Designated for Ukraine". DODIG Report No. DODIG-2023-076.

Flunkert, V., D. Salinas, and J. Gasthaus. 2017. "DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks". *CoRR* abs/1704.04110.

Gangopadhyay, T., S. Y. Tan, Z. Jiang, R. Meng and S. Sarkar. 2020. "Spatiotemporal Attention for Multivariate Time Series Prediction and Interpretation". *CoRR* abs/2008.04882.

GAO 2022. "Military Readiness: Actions Needed to Further Predictive Maintenance on Weapon Sytems". GAO Report No. GAO-23-105556.

Ho, T. K. 1995. "Random Decision Forests". In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, Volume 1, 278–282 vol.1.

Hojjati, H., M. Sadeghi, and N. Armanfard. 2023. "Multivariate Time-Series Anomaly Detection with Temporal Self-supervision and Graphs: Application to Vehicle Failure Prediction". In *Machine Learning and Knowledge Discovery in Databases: Applied Data Science and Demo Track*, edited by G. De Francisci Morales, C. Perlich, N. Ruchansky, N. Kourtellis, E. Baralis, and F. Bonchi, 242–259. Cham: Springer Nature Switzerland.

Kays, J. L., W. B. Carlton, M. M. Lee, and W. L. Ratliff. 1998. "Analysis of Operational Readiness Rates". Defense Technical Information Center.

Kuiper, P., S. Lin, J. Blanchet, and V. Tarokh. 2024. "Generative Learning for Simulation of Vehicle Faults". *arXiv preprint arXiv:2407.17654*.

Shafi, U., A. Safi, A. R. Shahid, S. Ziauddin and M. Q. Saleem. 2018. "Vehicle Remote Health Monitoring and Prognostic Maintenance System". *Journal of Advanced Transportation* 2018:1–10.

Theissler, A., J. Pérez-Velázquez, M. Kettelgerdes, and G. Elger. 2021. "Predictive Maintenance Enabled by Machine Learning: Use Cases and Challenges in the Automotive Industry". *Reliability Engineering & System Safety* 215:107864.

Thurston, M. G., S. P. McConky, C. J. Valant, and N. G. Nenadic. 2019. "Feasibility of Diagnostics, Prognostics and Hybrid Prognostics across Multiple Platforms". Defense Technical Information Center.

AUTHOR BIOGRAPHIES

PATRICK KUIPER received his B.S. in Operations Research from The United States Military Academy at West Point in 2007, and a M.Eng. in Applied Mathematics from Harvard University in 2014. He is an active duty officer in the United States Army and served as an instructor in the West Point Department of Mathematical Sciences from 2016 to 2019. He is currently pursuing a Ph.D. in Electrical and Computer Engineering at Duke University. His research interests include applications of generative modeling and extreme value theory. His email address is patrick.kuiper@duke.edu.

SIRUI LIN is a Ph.D. student at the Department of Management Science and Engineering (MS&E) at Stanford University, with research interest in applied probability and optimal transport-based statistical inference. His email address is siruilin@stanford.edu.

JOSE BLANCHET is a Professor of Management Science and Engineering (MS&E) at Stanford. Prior to joining Stanford, he was a professor at Columbia (Industrial Engineering and Operations Research, and Statistics, 2008-2017), and before that he taught at Harvard (Statistics, 2004-2008). Jose is a recipient of the 2010 Erlang Prize and several best publication awards in areas such as applied probability, simulation, operations management, and revenue management. He also received a Presidential Early Career Award for Scientists and Engineers in 2010. He is the Area Editor of Stochastic Models in Mathematics of Operations Research and has served on the editorial board of Advances in Applied Probability, Bernoulli, Extremes, Insurance: Mathematics and Economics, Journal of Applied Probability, Queueing Systems: Theory and Applications, and Stochastic Systems, among others. His email address is jose.blanchet@stanford.edu.

VAHID TAROKH worked at the AT&T Labs-Research until 2000. From 2000 to 2002 he was an Associate Professor at Massachusetts Institute of Technology (MIT). In 2002 he joined Harvard University as a Hammond Vinton Hayes Senior Fellow of Electrical Engineering and Perkins Professor in Applied Mathematics. He joined Duke University in 2018 as the Rhodes Family Professor in Electrical and Computer Engineering, Computer Science, and Mathematics and Bass Connections Endowed Professor. He was also a Gordon Moore Distinguished Research Fellow at CALTECH in 2018. Since Jan 2019, he has also been named as a Microsoft Data Science Investigator at Duke University. His email address is vahid.tarokh@duke.edu.