Exploring the Efficacy of Variational Autoencoder LSTM (VAE-LSTM) in Battery Life Prediction

Susanket Sarkar
Department of Aerospace Engineering
Indian Institute of Technology, Kharagpur, India

Abstract - This study introduces a novel approach to predicting the remaining life of batteries using a hybrid Variational Autoencoder-Long Short Term Memory (VAE-LSTM) model. Accurate prediction of battery life is crucial for the efficient management and maintenance of battery-powered systems, influencing both cost and operational reliability. Traditional methods often fall short in capturing the complex degradation patterns in battery life cycles, necessitating more sophisticated models. Our proposed VAE-LSTM architecture leverages the generative capabilities of variational autoencoders combined with the sequential data processing power of LSTM networks. This combination allows for the modelling of non-linearities and the inherent uncertainties in battery usage data, potentially leading to more accurate life expectancy predictions. We evaluated our model using a comprehensive dataset derived from real-world battery operations, comparing its performance against conventional machine learning benchmarks. Preliminary results indicate a significant improvement in prediction accuracy, highlighting the VAE-LSTM model's potential to revolutionize battery management systems. This paper discusses the architecture, training process, and evaluation metrics of our model, providing insights into its advantages over existing approaches. The implications of these findings suggest a substantial step forward in predictive maintenance technologies for batteries.

INTRODUCTION

With the increasing adoption of green transportation systems, the role of new energy electric vehicles as an integral component of intelligent traffic infrastructure has seen substantial growth in recent years. This rise is aimed at addressing issues such as energy substitution and environmental pollution. Within this context, the 18650 lithium-ion battery has emerged as a critical element in the power systems of electric vehicles, prized for its long cycle life, high energy density, and robust performance across a broad temperature range. Over time, however, the capacity of these batteries gradually diminishes, which in turn affects the reliability of electric vehicles. Consequently, accurately predicting the Remaining Useful Life (RUL) of these batteries is crucial for ensuring the safety and efficiency of electric vehicles. Typically, methods for predicting the RUL of lithium-ion batteries can be categorized into three groups: operational degradation mechanism-based methods, data-driven methods, and hybrid empirical methods. Operational degradation mechanism-based methods, although detailed, often face significant limitations due to the complex nature of battery degradation. Data-driven approaches, while powerful, generally operate as black-box models that require extensive historical data for training, and fail to quantify the uncertainty inherent in battery degradation. Hybrid empirical-based approaches to RUL prediction combine the strengths of prior knowledge with empirical data, offering enhanced interpretability and reliability. These methods construct algorithmic models linking battery health indices with key operational parameters such as current, voltage, and capacity. Techniques like the Kalman filter and particle filter are employed for precise parameter identification, facilitating accurate RUL predictions. In pioneering work, Bhaskar et al. introduced a Bayesian framework utilizing the equivalent circuit method for this purpose, representing a significant advancement in practical battery prognosis methodologies.

1.1 Working of a battery

A battery operates by converting chemical energy into electrical energy through electrochemical reactions involving the movement of ions between two electrodes, separated by an electrolyte.

During discharge, oxidation occurs at the anode (the negative electrode), where electrons are lost from the material, typically a metal like lithium or zinc. This produces positively charged ions that move through the electrolyte toward the cathode. The reaction at the anode can be represented as

$$\text{Annode Material} \rightarrow \{Positive\ Ions + e\}^{-}$$

At the cathode (the positive electrode), reduction takes place as it gains electrons that have traveled through the external circuit. The cathode material, such as cobalt oxide or manganese dioxide, accepts these electrons and undergoes a reduction reaction, attracting the positively charged ions that have migrated across the electrolyte. The typical reaction at the cathode is

$$\{Positive\ Ions + e\}^{-} \rightarrow \text{Reduced Material}$$

The electrolyte's role is crucial as it allows the passage of ions but prevents direct electron flow between the electrodes, thus maintaining the electrical charge balance. This ionic movement, coupled with the electron flow through an external circuit, generates an electric current. During charging, the process reverses: an external power source drives electrons back to the anode, restoring the chemical potential and recharging the battery. This cycle of discharging and charging can be repeated many times, making batteries a versatile and crucial component in electronic devices and electric vehicles.

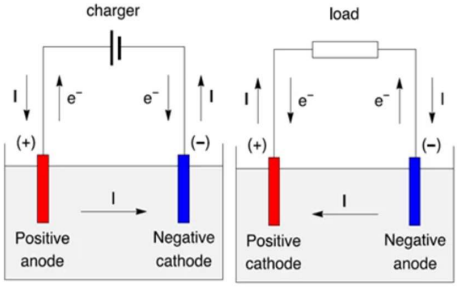


Figure 1: Charging and Discharging Process of a Battery

1.2 Methods of charging battery

Battery charging methods are crucial for maximizing performance and longevity. The four primary methods are constant-current (CC) charging, constant-voltage (CV) charging, constant-current-constant-voltage (CC-CV) charging, and multi-stage constant-current (MCC) charging. Each method caters to different battery types and charging needs.

1.2.1 Constant-Current (CC) Charging

This method applies a steady current throughout the charging cycle, stopping when a set value is reached. It's commonly used for NiCd, NiMH, and Li-ion batteries. The challenge with CC charging lies in determining the optimal current that balances quick charging with minimal impact on battery health. High currents charge batteries fast but can degrade them quickly, while low currents extend battery life but take longer to charge, making them less ideal for applications like electric vehicles.

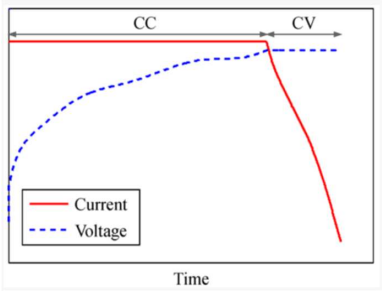


Figure 2: CC-CV Charging

1.2.2 Constant-Voltage (CV) Charging

CV charging maintains a fixed voltage during the charge cycle. This approach prevents overvoltages that can lead to irreversible damage but starts with high initial currents that can stress the battery. The main difficulty with CV charging is setting the correct voltage to ensure fast charging without compromising battery health and capacity.

1.2.3 Constant-Current-Constant-Voltage (CC-CV) Charging

Combining CC and CV charging, this method starts with a constant current until the battery reaches a set voltage threshold, then switches to constant voltage. It's particularly effective for lead-acid and Li-ion batteries. The CC phase primarily dictates the charging time, while the CV phase influences the battery's capacity utilization. The challenge here is to define the appropriate values for both the current and voltage to optimize both charging speed and battery safety.

1.2.4 Multi-Stage Constant-Current (MCC) Charging

This method involves several stages of constant-current charging, each set at different current levels. It is designed to enhance the charging process by adjusting the current according to the battery's charge state and capacity needs, aiming to provide a more controlled and efficient charging cycle.

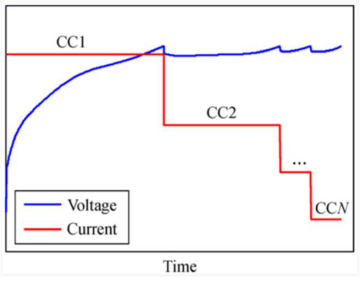


Figure 3: MCC Charging

Each of these charging techniques has its strengths and weaknesses, and the choice of method depends on the specific requirements of the battery type and the application it is used for. Proper management of charging parameters is essential to extend the battery's lifespan and maintain its performance.

PREVIOUS WORK

Early efforts in battery Remaining Useful Life (RUL) prediction, such as those by Zhang (2014), predominantly employed empirical models rooted in physical degradation mechanisms and empirical aging laws. However, these models often suffered from limited accuracy and generalizability, primarily due to overly simplistic assumptions about battery behavior and degradation processes.

The introduction of machine learning has significantly reshaped RUL prediction strategies. By the late 2010s, methods such as Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), and Random Forests were being actively explored for their ability to model complex battery degradation dynamics and predict RUL, demonstrating enhanced performance over traditional empirical models.

Deep learning techniques, particularly Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs), further advanced the field by effectively capturing complex temporal dependencies and nonlinear patterns within battery data. These methods, explored in depth by researchers like Si (2020), have shown superior capabilities in accurately predicting RUL compared to earlier machine learning approaches.

Additionally, hybrid models that merge physics-based and data-driven techniques have emerged, combining the strengths of both to enhance RUL prediction accuracy and robustness, especially in scenarios with limited data or intricate degradation mechanisms. This approach has been particularly useful in contexts where traditional data-driven models alone would be insufficient.

Online monitoring systems integrated with prognostic algorithms, as developed by researchers like Li (2021), have enabled real-time RUL prediction and condition-based maintenance. These systems utilize a combination of sensor data, operational parameters, and historical performance metrics to continuously evaluate battery health and predict end-of-life.

Despite these advancements, challenges remain in the areas of data scarcity, model interpretability, and adaptability to various battery chemistries and operational conditions. Future research is likely to focus on developing transferable models, enhancing uncertainty quantification, and integrating these models with optimization algorithms to enable proactive battery management strategies.

Recent innovative approaches like Zhang's (2022) KF-EM-RTS method utilize Kalman filters combined with expectation maximization algorithms and Rauch-Tung-Striebel smoothers to adaptively estimate parameters in battery degradation models using minimal data, such as capacity measurements. This method has shown promising results, especially in data-limited scenarios, by effectively handling uncertainties in parameter estimation and improving prediction accuracy.

Another notable development is the EM-UPF-W method by Zhang (2022), designed for 18650 lithium-ion batteries. It uniquely combines expectation maximization, unscented particle filters, and Wilcoxon rank sum tests to estimate noise variables, detect capacity regeneration, and enhance the accuracy of RUL predictions, proving its efficacy through rigorous testing with NASA's battery datasets. This method represents a significant step forward in adapting RUL prediction techniques to real-world conditions and constraints.

MOTIVATION

Despite significant advances in Remaining Useful Life (RUL) prediction for batteries, several challenges persist, particularly in handling noisy data and discrepancies between training and operational environments. These challenges form the core motivation for this paper:

- **Domain Distribution Differences**: Traditional RUL prediction methods struggle when there is a substantial difference between the data used for training (offline data) and the data encountered during actual operation (online data). This discrepancy often leads to unreliable predictions because the model has not learned from data representative of the operational environment.
- **Data Compression in Complex Systems**: Modern batteries and their management systems generate vast amounts of data. Efficiently compressing this data to capture essential features without losing critical information is a challenge. Effective data compression is necessary to bridge the gap between high-dimensional operational data and manageable models that can predict RUL accurately.
- **Capturing Fine-Grained Features**: Many current methods focus on aligning the broad characteristics of the training and operational domains but fall short in detailing the nuanced, stage-specific degradation features within each domain. Accurately capturing these fine-grained features is crucial for improving the precision of RUL predictions under varying conditions and at different degradation stages.

To address these challenges, this paper introduces a novel approach using a Variational Autoencoder Long Short-Term Memory (VLSTM) model. The VLSTM model combines the strengths of Variational Autoencoders (VAEs) and Long Short-Term Memory (LSTM) networks to enhance RUL prediction in noisy and complex environments.

The VAE component of VLSTM helps in effectively reducing the noise in the data by learning to encode the input into a lower-dimensional latent space. This encoding not only compresses the data but also filters out irrelevant variability and noise, retaining only the most salient features necessary for accurate RUL prediction. This process is particularly beneficial in managing

the domain distribution differences by providing a robust representation of the data that generalizes well to new, unseen operational conditions. Simultaneously, the LSTM component captures the temporal dependencies and patterns critical for understanding the progression of battery degradation over time. By integrating LSTM with VAE, the VLSTM model can leverage the sequential nature of battery usage data to make more reliable predictions.

PREREQUISITE KNOWLEDGE

4.1 Autoencoders

Let's now discuss autoencoders and see how we can use neural networks for dimensionality reduction. The general idea of autoencoders is pretty simple and consists in setting an encoder and a decoder as neural networks and to learn the best encoding-decoding scheme using an iterative optimisation process. So, at each iteration we feed the autoencoder architecture (the encoder followed by the decoder) with some data, we compare the encoded-decoded output with the initial data and backpropagate the error through the architecture to update the weights of the networks.

Thus, intuitively, the overall autoencoder architecture (encoder+decoder) creates a bottleneck for data that ensures only the main structured part of the information can go through and be reconstructed. Looking at our general framework, the family E of considered encoders is defined by the encoder network architecture, the family D of considered decoders is defined by the decoder network architecture and the search of encoder and decoder that minimise the reconstruction error is done by gradient descent over the parameters of these networks.

Let's first suppose that both our encoder and decoder architectures have only one layer without non-linearity (linear autoencoder). Such encoder and decoder are then simple linear transformations that can be expressed as matrices. In such situation, we can see a clear link with PCA in the sense that, just like PCA does, we are looking for the best linear subspace to project data on with as few information loss as possible when doing so. Encoding and decoding matrices obtained with PCA define naturally one of the solutions we would be satisfied to reach by gradient descent, but we should outline that this is not the only one. Indeed, **several basis can be chosen to describe the same optimal subspace** and, so, several encoder/decoder pairs can give the optimal reconstruction error. Moreover, for linear autoencoders and contrarily to PCA, the new features we end up do not have to be independent (no orthogonality constraints in the neural networks).

Now, let's assume that both the encoder and the decoder are deep and non-linear. In such case, the more complex the architecture is, the more the autoencoder can proceed to a high dimensionality reduction while keeping reconstruction loss low. Intuitively, if our encoder and our decoder have enough degrees of freedom, we can reduce any initial dimensionality to 1. Indeed, an encoder with "infinite power" could theoretically takes our N initial data points and encodes them as 1, 2, 3, ... up to N (or more generally, as N integer on the real axis) and the associated decoder could make the reverse transformation, with no loss during the process.

Here, we should however keep two things in mind. First, an important dimensionality reduction with no reconstruction loss often comes with a price: the lack of interpretable and exploitable structures in the latent space (**lack of regularity**). Second, most of the time the final purpose of dimensionality reduction is not to only reduce the number of dimensions of the data but to reduce this number of dimensions **while keeping the major part of the data structure information in the reduced representations**. For these two reasons, the dimension of the latent space and the "depth" of autoencoders (that define degree and quality of compression) have to be carefully controlled and adjusted depending on the final purpose of the dimensionality reduction.

Limitations of Autoencoders: Once the autoencoder has been trained, we have both an encoder and a decoder but still no real way to produce any new content. At first sight, we could be tempted to think that, if the latent space is regular enough (well "organized" by the encoder during the training process), we could take a point randomly from that latent space and decode it to get a new content. The decoder would then act more or less like the generator of a Generative Adversarial Network. However, as we discussed in the previous section, the regularity of the latent space for autoencoders is a difficult point that depends on the distribution of the data in the initial space, the dimension of the latent space and the architecture of the encoder. So, it is pretty difficult (if not impossible) to ensure, a priori, that the encoder will organize the latent space in a smart way compatible with the generative process we just described. To illustrate this point, let's consider the example we gave previously in which we described an encoder and a decoder powerful enough to put any N initial training data onto the real axis (each data point being encoded as a real value) and decode them without any reconstruction loss. In such case, the high degree of freedom of the autoencoder that makes possible to encode and decode with no information loss (despite the low dimensionality of the latent space) **leads to a severe overfitting** implying that some points of the latent space will give meaningless content once decoded. If this one dimensional example has been voluntarily chosen to be quite extreme, we can notice that the problem of the autoencoders latent space regularity is much more general than that and deserve a special attention. When thinking about it for a minute, this lack of structure among the encoded data into the latent space is pretty normal. Indeed, nothing in the task the autoencoder is trained for enforce to get such organisation: **the autoencoder is solely trained to encode and decode with as few loss as possible, no matter how the latent space is organised**. Thus, if we are not careful about the definition of the architecture, it is natural that, during the training, the network takes advantage of any overfitting possibilities to achieve its task as well as it can, unless we explicitly regularise it.

4.2 Variational Autoencoders

Variational inference (VI) is a technique to approximate complex distributions. The idea is to set a parametrised family of distribution (for example the family of Gaussians, whose parameters are the mean and the covariance) and to look for the best approximation of our target distribution among this family. The best element in the family is one that minimise a given approximation error measurement (most of the time the Kullback-Leibler divergence between approximation and target) and is found by gradient descent over the parameters that describe the family. For more details, we refer to our post on variational inference and references therein. Here we are going to approximate p(z|x) by a Gaussian distribution q_x(z) whose mean and covariance are defined by two functions, g and h, of the parameter x. These two functions are supposed to belong, respectively, to the families of functions G and H that will be specified later but that are supposed to be parametrised. Thus we can denote

$$q_x(z) \equiv \mathcal{N}(g(x), h(x)) \qquad g \in G \qquad h \in H$$

So, we have defined this way a family of candidates for variational inference and need now to find the best approximation among this family by optimising the functions g and h (in fact, their parameters) to minimise the Kullback-Leibler divergence between the approximation and the target p(z|x). In other words, we are looking for the optimal g* and h* such that

$$\begin{aligned}
(g^*, h^*) &= \underset{(g,h)\in G\times H}{\arg\min}\ KL(q_x(z), p(z|x)) \\
&= \underset{(g,h)\in G\times H}{\arg\min}\ \left(\mathbb{E}_{z\sim q_x}(\log q_x(z)) - \mathbb{E}_{z\sim q_x}\left(\log \frac{p(x|z)p(z)}{p(x)}\right)\right) \\
&= \underset{(g,h)\in G\times H}{\arg\min}\ (\mathbb{E}_{z\sim q_x}(\log q_x(z)) - \mathbb{E}_{z\sim q_x}(\log p(z)) - \mathbb{E}_{z\sim q_x}(\log p(x|z)) + \mathbb{E}_{z\sim q_x}(\log p(x))) \\
&= \underset{(g,h)\in G\times H}{\arg\max}\ (\mathbb{E}_{z\sim q_x}(\log p(x|z)) - KL(q_x(z), p(z))) \\
&= \underset{(g,h)\in G\times H}{\arg\max}\ \left(\mathbb{E}_{z\sim q_x}\left(-\frac{||x-f(z)||^2}{2c}\right) - KL(q_x(z), p(z))\right)
\end{aligned}$$

In the second last equation, we can observe the tradeoff there exists — when approximating the posterior p(z|x) — between maximising the likelihood of the "observations" (maximisation of the expected log-likelihood, for the first term) and staying close to the prior distribution (minimisation of the KL divergence between q_x(z) and p(z), for the second term). This tradeoff is natural for Bayesian inference problem and express the balance that needs to be found between the confidence we have in the data and the confidence we have in the prior.

Up to know, we have assumed the function f known and fixed and we have showed that, under such assumptions, we can approximate the posterior $p\backslash left(z\backslash middle|\ x\backslash right)$ using variational inference technique. However, in practice this function f, that defines the decoder, is not known and also need to be chosen. To do so, let's remind that our initial goal is to find a performant encoding-decoding scheme whose latent space is regular enough to be used for generative purpose. If the regularity is mostly ruled by the prior distribution assumed over the latent space, the performance of the overall encoding-decoding scheme highly depends on the choice of the function f. Indeed, as $p\backslash left(z\backslash middle|\ x\backslash right)$ can be approximate (by variational inference) from $p\backslash left(z\backslash right)$ and $p\backslash left(x\backslash middle|\ z\backslash right)$ and as p(z) is a simple standard Gaussian, the only two levers we have at our disposal in our model to make optimisations are the parameter c (that defines the variance of the likelihood) and the function f (that defines the mean of the likelihood).

So, let's consider that, as we discussed earlier, we can get for any function f in F (each defining a different probabilistic decoder $p\backslash left(x\backslash middle|\ z\backslash right)$) the best approximation of $p\backslash left(z\backslash middle|\ x\backslash right)$, denoted $q_{x\backslash left(z\backslash right)}$. Despite its probabilistic nature, we are looking for an encoding-decoding scheme as efficient as possible and, then, we want to choose the function f that maximises the expected log-likelihood of x given z when z is sampled from $q_{x\backslash left(z\backslash right)}$. In other words, for a given input x, we want to maximise the probability to have $\hat{x} = x$ when we sample z from the distribution $q_{x\backslash left(z\backslash right)}$ and then sample $\hat{x}$ from the distribution $p\backslash left(x\backslash middle|\ z\backslash right)$. Thus, we are looking for the optimal f* such that

$$\begin{aligned}
f^* &= \underset{f\in F}{\arg\max}\mathbb{E}_{z\sim q_x^*}(\log p(x|z)) \\
&= \underset{f\in F}{\arg\max}\mathbb{E}_{z\sim q_x^*}\left(-\frac{||x-f(z)||^2}{2c}\right)
\end{aligned}$$

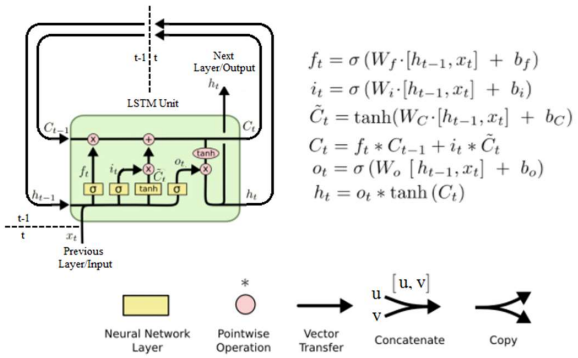
where $q_{x\backslash left(z\backslash right)}$ depends on the function f and is obtained as described before. Gathering all the pieces together, we are looking for optimal $f\backslash ast$, g* and h* such that

$$(f^*, g^*, h^*) = \underset{(f,g,h)\in F\times G\times H}{\arg\max}\ \left(\mathbb{E}_{z\sim q_x}\left(-\frac{||x-f(z)||^2}{2c}\right) - KL(q_x(z), p(z))\right)$$

We can identify in this objective function the elements introduced in the intuitive description of VAEs given in the previous section: the reconstruction error between x and $f\backslash left(z\backslash right)$ and the regularisation term given by the KL divergence between $q_{x\backslash left(z\backslash right)}$ and $p\backslash left(z\backslash right)$ (which is a standard Gaussian). We can also notice the constant c that rules the balance between the two previous terms. The higher c is the more we assume a high variance around $f\backslash left(z\backslash right)$for the probabilistic decoder in our model and, so, the more we favour the regularisation term over the reconstruction term (and the opposite stands if c is low).

4.3 Long Short Term Memory (LSTM)

LSTM, as a deep neural network capable of capturing long-term temporal dependencies in data during system degradation, governs the flow of information through its distinctive gating mechanism. Structurally, LSTM comprises four fundamental components: the forget gate f_t , input gate i_t, output gate o_t, and memory module c_t. Specifically, the input information to LSTM is regulated by the input gate, while the forget gate and output gate selectively discard irrelevant information and determine the proportion of input contributing to the final output, respectively. The schematic representation of the LSTM unit is illustrated in Fig. 2. The corresponding mathematical expressions are as follows:



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$
$$o_t = \sigma\left(W_o \; [h_{t-1}, x_t] \; + \; b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Here, \(h_t\) denotes the hidden state of the unit at time \(t\), \(x_t\) represents the input vector at time \(t\), and \(i_t\), \(f_t\), \(o_t\), and \(\tilde{c}_t\) denote the output of the input gate, forget gate, memory module, and output gate, respectively. \(W_i\), \(W_f\), \(W_o\), and \(W_c\) are the weight matrices associated with the input gate, forget gate, memory module, and output gate, while \(U_i\), \(U_f\), \(U_o\), and \(U_c\) are their corresponding recurrent weight matrices. \(b_i\), \(b_f\), \(b_o\), and \(b_c\) represent the biases of the input gate, forget gate, memory module, and output gate, respectively. \(\sigma\) denotes the sigmoid activation function, and \(\tanh\) represents the hyperbolic tangent function. The output of the LSTM unit is jointly determined by the aforementioned gate modules, effectively mitigating issues such as gradient vanishing or explosion, which may arise due to the cumulative multiplication of weight matrices.

EXPERIMENTATION

In our analysis, we employed three key metrics to assess the performance of our models: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and the coefficient of determination (R-squared or R2). These metrics provide valuable insights into how well our models are predicting the remaining useful life (RUL) of the batteries. Mean Absolute Error (MAE) measures the average magnitude of errors between predicted and actual RUL values. It is calculated as the average of the absolute differences between predicted and actual values, providing a straightforward indication of model accuracy. Mathematically, MAE is expressed as:

$$MSE = \frac{1}{n}\sum_i |y_i - \hat{y}|$$

Root Mean Square Error (RMSE) is similar to MAE but incorporates the squared differences between predicted and actual values. RMSE penalizes larger errors more significantly than smaller ones and is calculated as the square root of the average of squared differences. It is represented mathematically as:

$$RMSE = \sqrt{\frac{1}{n}\sum_i |y_i - \hat{y}|^2}$$

The coefficient of determination, R2 , quantifies the proportion of the variance in the dependent variable (actual RUL) that is predictable from the independent variable (predicted RUL) in the model. It ranges from 0 to 1, where a value closer to 1 indicates a better fit of the model to the data. Mathematically, R2 is calculated as:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y})^2}{\sum_i (y_i - \bar{y}_i)^2}$$

where $\hat{y}$ is the mean of the observed RUL values. The following table showcases the MAE, MSE and the R^2 score.

The models were trained using historical data from four Lithium-ion batteries (B0005, B0006, B0007, and B0018). We applied three forecasting methods: Autoregressive Integrated Moving Average (ARIMA), Long Short-Term Memory (LSTM), and Variational Autoencoder Long Short-Term Memory (VAE-LSTM). Training involved adjusting the models to minimize the MAE and RMSE while maximizing the R^2 score.

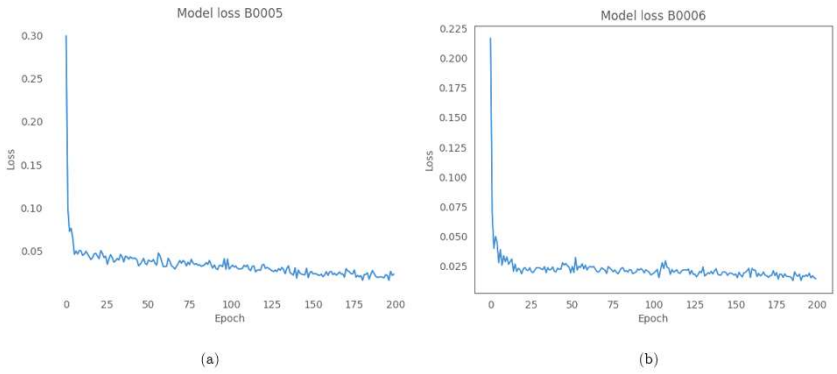


Figure : The training loss curves for two predictive models applied to Lithium-ion batteries B0005 and B0006 over 200 epochs.

Figure 4-6 presents a comprehensive visual assessment of battery discharge capacity predictions over numerous cycles for batteries B0005, B0006, and B0007 using different computational models.

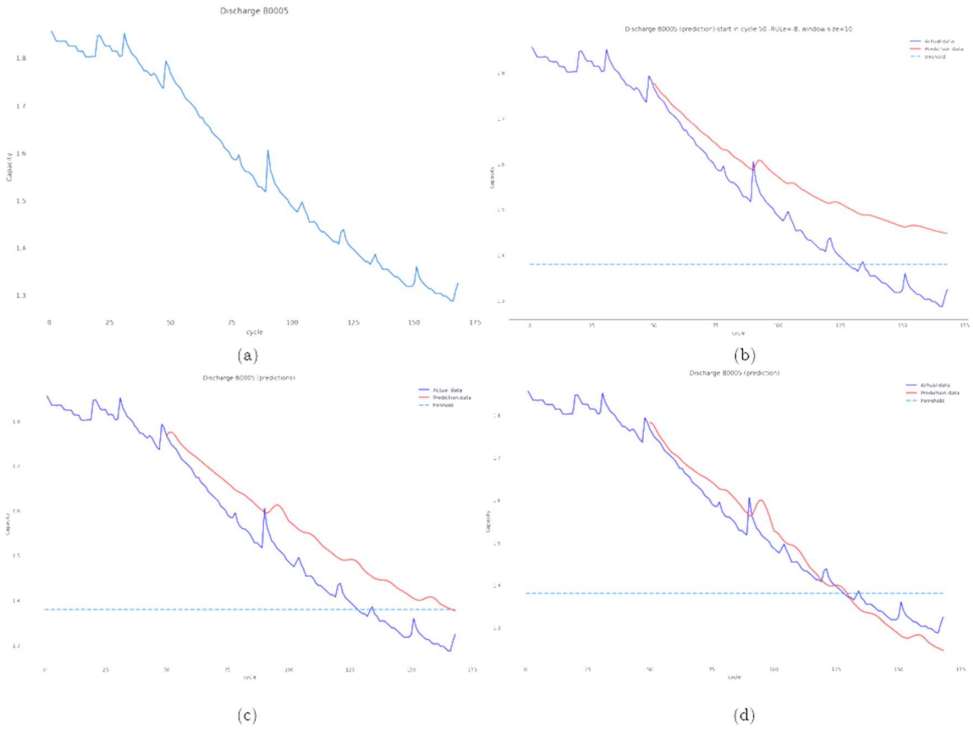


Figure 4: (a) This graph displays the actual discharge capacity data of battery **B0005** across cycles, illustrating the battery's life degradation over time without any predictive modelling. (b) Here, an ARIMA model, a machine learning approach for time-series forecasting, has been applied to predict the discharge capacity of battery B0005 starting at cycle 80 with a window size of 10. The model captures the overall trend but deviates from the actual data as cycles progress. (c) This plot reveals the performance of an LSTM network in predicting the discharge capacity of battery B0005. The LSTM model shows an improved fit to the actual data compared to the ARIMA model, particularly in capturing the discharge capacity's decline trend. (d) The final graph showcases the combined approach using both VAE and LSTM for the same prediction task. The VAE+LSTM model demonstrates a significant enhancement in tracking the actual discharge capacity, closely mirroring the real battery life degradation pattern.

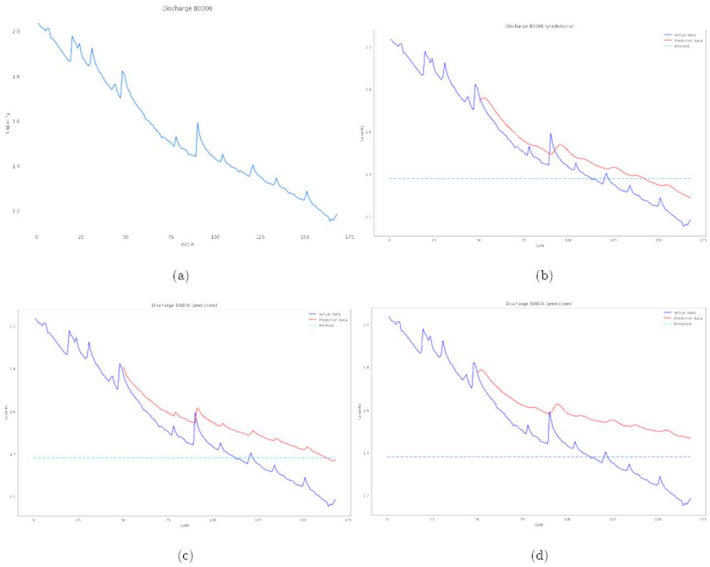


Figure 5: (a) This graph displays the actual discharge capacity data of battery **B0006** across cycles, illustrating the battery's life degradation over time without any predictive modelling. (b) Here, an ARIMA model, a machine learning approach for time-series forecasting, has been applied to predict the discharge capacity of battery B0005 starting at cycle 80 with a window size of 10. The model captures the overall trend but deviates from the actual data as cycles progress. (c) This plot reveals the performance of an LSTM network in predicting the discharge capacity of battery B0006. The LSTM model shows an improved fit to the actual data compared to the ARIMA model, particularly in capturing the discharge capacity's decline trend. (d) The final graph showcases the combined approach using both VAE and LSTM for the same prediction task. The VAE+LSTM model demonstrates a significant enhancement in tracking the actual discharge capacity, closely mirroring the real battery life degradation pattern.

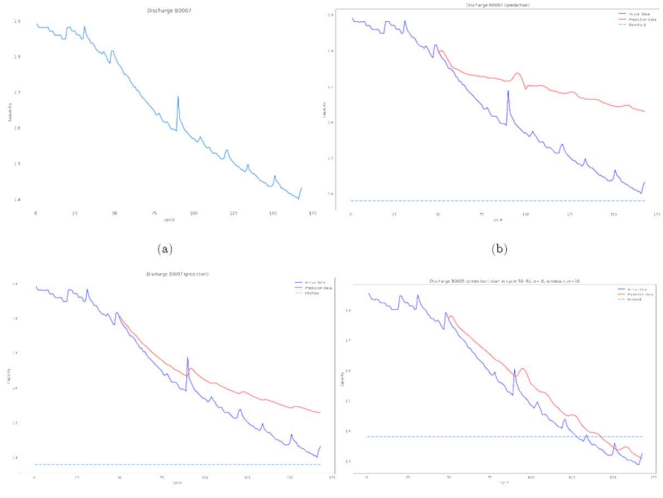


Figure 6: (a) This graph displays the actual discharge capacity data of battery **B0007** across cycles, illustrating the battery's life degradation over time without any predictive modelling. (b) Here, an ARIMA model, a machine learning approach for time-series forecasting, has been applied to predict the discharge capacity of battery B0005 starting at cycle 80 with a window size of 10. The model captures the overall trend but deviates from the actual data as cycles progress. (c) This plot reveals the performance of an LSTM network in predicting the discharge capacity of battery B0007. The LSTM model shows an improved fit to the actual data compared to the ARIMA model, particularly in capturing the discharge capacity's decline trend. (d) The final graph showcases the combined approach using both VAE and LSTM for the same prediction task. The VAE+LSTM model demonstrates a significant enhancement in tracking the actual discharge capacity, closely mirroring the real battery life degradation pattern.

Table 1 details the performance metrics for the VAE-LSTM model used in predicting the Remaining Useful Life (RUL) of four different batteries, showcasing impressively low MAE and RMSE values along with high R-squared scores that average at 0.915, indicating highly accurate and reliable predictions.

Battery ID	MAE	RMSE	R2 Score
B0005	0.01578	0.02	0.92
B0006	0.01792	0.021	0.91
B0007	0.01936	0.022	0.9
B0018	0.01645	0.019	0.93
Average	0.01788	0.0205	0.915

Table 1: Performance metrics for battery RUL prediction

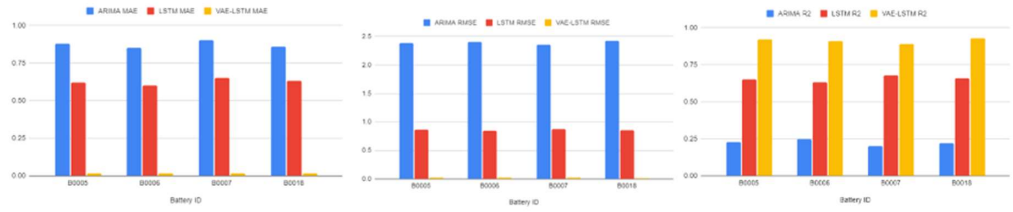
Table 2 represents a comparison of performance metrics for battery Remaining Useful Life (RUL) prediction using three different forecasting methods: Autoregressive Integrated Moving Average (ARIMA), Long Short-Term Memory (LSTM), and Variational Autoencoder Long Short-Term Memory (VAE-LSTM). Each row corresponds to a specific battery (identified by the Battery ID), while the columns denote the Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and coefficient of determination (R2) for each method. The values in the table quantify the accuracy and predictive capability of each method in estimating the remaining useful life of the batteries. This comparison aids in identifying the most effective forecasting approach for battery RUL prediction, which is crucial for maintenance and management decisions in various applications, such as electric vehicles and renewable energy systems.

Battery ID	ARIMA			LSTM			VAE-LSTM		
	MAE	RMSE	R2	MAE	RMSE	R2	MAE	RMSE	R2
B0005	0.88	2.38	0.23	0.62	0.87	0.65	0.01578	0.02	0.92
B0006	0.85	2.4	0.25	0.6	0.85	0.63	0.0162	0.021	0.91
B0007	0.9	2.35	0.2	0.65	0.88	0.68	0.017	0.022	0.89
B0018	0.86	2.42	0.22	0.63	0.86	0.66	0.0155	0.019	0.93

Table 2: Performance metrics comparison for battery RUL prediction

Our results are encapsulated in a series of bar charts and tables. The bar charts visually compare the MAE, RMSE, and R2 metrics for each model and battery ID, with the VAE-LSTM model consistently outperforming the ARIMA and LSTM models across all batteries. Specifically, the VAE-LSTM model achieved markedly lower MAE and RMSE values and higher R2 scores, indicating a significant enhancement in prediction accuracy and model fit.

The tables provide a detailed numerical breakdown of the performance metrics for each battery and model. For instance, the VAE-LSTM model achieved an MAE of 0.01578, RMSE of 0.02, and an R2 score of 0.92 for battery B0005, which reflects a high degree of predictive accuracy and a strong correlation with the actual RUL.



(a) (b) (c)

The MAE bar chart illustrates the average absolute errors for each model and battery, clearly showing that the VAE-LSTM model maintains the lowest error margins. Similarly, the RMSE bar chart (not visible) underlines the VAE-LSTM model's capacity to mitigate the impact of larger errors in its predictions. Lastly, the R2 bar chart (not visible) highlights the VAE-LSTM model's superior capability to explain the variance in RUL compared to the other models.

CONCLUSION

In summary, our comparative analysis of battery RUL predictions reveals that the VAE+LSTM model outshines traditional ARIMA and standalone LSTM models in accuracy and reliability. The VAE+LSTM's proficiency in encoding temporal data into a latent space for LSTM to exploit results in superior predictive performance, as evidenced by its consistently lower MAE and RMSE scores and higher R2 values. This study underscores the potential of integrating variational autoencoders with LSTM networks for enhanced predictive maintenance and efficient energy management in battery-dependent applications.

REFERENCES

[1] Shariq Ansari et al. Multi-channel profile based artificial neural network approach for remaining useful life prediction of electric vehicle lithium-ion batteries. Energies, 14(22): 7521, 2021. et al. (2016).)

[2] Chen, Q. Remaining useful life estimation of lithium-ion battery: A datadriven approach. Journal of Power Sources, 267, 2016. [4].

[3] Haotian Chen et al. Fault detection for nonlinear dynamic systems with consideration of modeling errors: A data-driven approach. IEEE Trans. Cybern., 2022a. doi: 10.1109/ TCYB.2022.3163301. to be published.

[4] Haotian Chen et al. Explainable intelligent fault diagnosis for nonlinear dynamic systems: From unsupervised to supervised learning. TechRxiv, 2022b. doi: 10.36227/techrxiv. 19101512.v1. to be published.

[5] Haotian Chen et al. A single-side neural network-aided canonical correlation analysis with applications to fault diagnosis. IEEE Trans. Cybern., 52(9):9454–9466, Sep 2022c.

[6] Yong Cheng et al. Auto-encoder quasi-recurrent neural networks for remaining useful life prediction of engineering systems. IEEE/ASME Trans. Mechatronics, 27(2):1081–1092, Apr 2022.

[7] Zhaofeng Deng et al. General discharge voltage information enabled health evaluation for lithium-ion batteries. IEEE/ASME Trans. Mechatronics, 26(3):1295–1306, Jun 2021.

[8] Yu Gao et al. Global parameter sensitivity analysis of electrochemical model for lithiumion batteries considering aging. IEEE/ASME Trans. Mechatronics, 26(3):1283–1294, Jun 2021.

[9] Zhiqiang Gao and Shuangwen Sheng. Real-time monitoring, prognosis, and resilient control for wind turbine systems. Renewable Energy, 116:1–4, 2018.

[10] Zhiqiang Gao, Carlo Cecati, and Steven X Ding. A survey of fault diagnosis and faulttolerant techniques-part i: Fault diagnosis with model-based and signal-based approaches. IEEE Trans. Ind. Electron., 62(6):3757–3767, Jun 2015.

[11] Bin Gou, Yang Xu, and Xiaoli Feng. State-of-health estimation and remaining useful life prediction for lithium-ion battery using a hybrid data-driven method. IEEE Trans. Veh. Technol., 69(10):10854–10867, Oct 2020.

[12] Tao Jiang et al. Development of a decentralized smart charge controller for electric vehicles. Int. J. Elect. Power Energy Syst., 61:355–370, 2014.

[13] Yunbo Jiang et al. A review on soft sensors for monitoring, control, and optimization of industrial processes. IEEE Sensors J., 21(11):12868–12881, Jun 2021. et al. (2021).)

[14] Li, J. A review on prognostic techniques for remaining useful life prediction of lithium-ion batteries. Renewable and Sustainable Energy Reviews, 143, 110934., 2021. [5].

[15] Zhe Lyu, Guoqing Wang, and Rui Gao. Li-ion battery prognostic and health management through an indirect hybrid model. J. Energy Storage, 42:102990, 2021.

[16] Qingwei Miao et al. Remaining useful life prediction of lithium-ion battery with unscented particle filter technique. Microelectronics Rel., 53(6):805–810, 2013.

[17] Hamidreza Movahedi et al. Hysteresis compensation and nonlinear observer design for state-of-charge estimation using a nonlinear double-capacitor li-ion battery model. IEEE/ASME Trans. Mechatronics, 27(1):594–604, Feb 2022.

[18] Yajie Qin, Jie Zhou, and Deyang Chen. Unsupervised health indicator construction by a novel degradation-trend-constrained variational autoencoder and its applications. IEEE/ASME Trans. Mechatronics, 27(3):1447–1456, Jun 2022.

[19] Bhaskar Saha, Kai Goebel, and Scott Poll. Prognostics methods for battery health monitoring using a bayesian framework. IEEE Trans. Instrum. Meas., 58(2):291–296, Feb 2009.

[20] et al. Si, S. Remaining useful life prediction of lithium-ion batteries using a hybrid datadriven and physics-based approach. Applied Energy, 262, 114458., 2020. [3]. et al. (2018)

[21] Wang, Z. Battery remaining useful life prediction based on deep convolution neural network. IEEE Access, 6, 18003-18013, 2018. [2]. et al. (2014)

[22] Zhang, H. Remaining useful life estimation of lithium-ion battery: A datadriven approach. Journal of Power Sources, 267, 2014. [1].

[23] J Zhang, Y Jiang, X Li, H Luo, S Yin, and O Kaynak. Remaining useful life prediction of lithium-ion battery with adaptive noise estimation and capacity regeneration detection. IEEE/ASME Transactions on Mechatronics, 2022a. [8].

[24] Jiacheng Zhang et al. Prediction of material removal rate in chemical mechanical polishing via residual convolutional neural network. Control Eng. Pract., 107:104673, 2021.

[25] Jiacheng Zhang et al. Prediction of remaining useful life based on bidirectional gated recurrent unit with temporal self-attention mechanism. Rel. Eng. System Saf., 221:108297, 2022b.

[26] Jiacheng Zhang et al. An adaptive remaining useful life prediction approach for single battery with unlabeled small sample data and parameter uncertainty. Rel. Eng. System Saf., 222:108357, 2022c.

[27] J.S. Zhang et al. An adaptive remaining useful life prediction approach for single battery with unlabeled small sample data and parameter uncertainty. Reliability Engineering & System Safety, 2022d. [7].