Trustworthy Battery Management: A Digital Twin Approach Leveraging XAI and Blockchain

Judith Nkechinyere Njoku *, Cosmas Ifeanyi Nwakanma†, Jae-Min Lee *, and Dong-Seong Kim *
*IT Convergence Engineering, † ICT Convergence Research Center, Kumoh National Institute of Technology, Korea.

***Abstract*—This study presents a digital twin framework for predicting the state of health (SoH) in battery management systems (BMS). This framework integrates a single particle model with electrolytes (SPME) and a long-short-term memory (LSTM) network to model battery behaviour based on a NASA battery dataset. To ensure the security of battery data, data is recorded on the Ethereum blockchain and queried when needed for secure prediction. To ensure the interpretability of the predictions, an explainable AI (XAI) approach, SHAP, is employed. Experimentation shows the viability of the proposed framework in accurately predicting the SoH of physical batteries.**

***Index Terms*—battery management system, blockchain, digital twin, explainable AI, battery**

I. Introduction

Batteries are essential components in various technologies, such as electric vehicles, energy storage systems, and portable electronic devices [1]. The indispensable role of batteries highlights the importance of employing sophisticated battery management systems (BMS) to optimize their performance, safety, and longevity [2]. However, conventional BMS often face challenges in real-time adjustments and accurately determining the battery's state of health (SoH) which is crucial for maximizing battery usage and lifespan [3]. These difficulties stem from the intricate behavior of batteries, which changes based on usage conditions and over time. Furthermore, most BMSs operate on resource-constrained devices, which may lead to delays in providing analytics results or processing data. One solution to this challenge is the use of digital twins [2].

The digital twin is a solution that creates a virtual counterpart of the physical battery system [2]. Utilizing live data enables the simulation of battery performance and the prediction of future conditions, offering a deeper insight into its operation and potential issues. Numerous studies have successfully employed digital twins in BMS, highlighting its significant impact on the BMS ecosystem [2]. To successfully develop a viable battery digital twin, some studies have explored the use of battery models such as the single particle model (SPM) and single particle model with electrolyte (SPMe) [4] renowned for their well-balanced representation of battery electrochemistry, serving as the foundation of the digital twin and delivering precise insights into the battery's behavior. Another critical part of developing digital twins is incorporating artificial intelligence (AI) [3], [5]. AI algorithms utilize the battery model and real-time operational data to accurately predict SoH and other critical parameters, facilitating proactive decision-making.

A significant limitation of these digital twin solutions is their heavy reliance on AI algorithms [6] due to the "black box" nature of AI models obscuring their decision-making processes, potentially eroding user trust. This challenge has been addressed in other domains using explainable AI (XAI). XAI is a concept that is gradually permeating diverse domains due to its goal of ensuring that AI models are interpretable and trustworthy [7]. According to studies by [6]–[8], the interpretability of any approach is made up of three objectives, including efficiency, fidelity, and understandability. An interpretable model should be faithful to the data and the original model, easy to comprehend, and quickly understood by the end-users who seek to make informed decisions. XAI approaches like the local interpretable model-agnostic explanations (LIME) and Shapley additive explanations (SHAP) have been proposed as viable methods [8]. Another limitation of digital twins is that they rely heavily on data, which malicious users can easily compromise. This results in developing a digital twin that does not truly represent the actual physical battery system [9]. To address this challenge, some studies have explored blockchain to ensure that data employed to develop digital twins remain secure [10], [11]. This research aims to enhance the precision of SoH predictions, enhance data security within the BMS, and make the system's decisions more transparent and trustworthy for users.

The main contributions of this paper are: (i) A hybrid digital twin that incorporates the SPMe battery model and a long-short-term memory (LSTM) network is proposed for a more realistic prediction of battery SoH. (ii) The interpretability of the proposed digital twin model is demonstrated using the SHAP XAI approach. (iii) The security of the battery data used for developing the digital twin model is secured using the Ethereum blockchain. (iv) The work is verified in a BMS scenario using a publicly available dataset.

II. Methodology

A. Digital twin framework

As illustrated in Fig. 1, the system gathers real-time battery data, subsequently recorded in a database and secured on the blockchain network. This process emphasizes the integrity and protection of the data, ensuring it is immutable. The system utilizes SPMe to simulate battery behavior using the *current* data from the actual battery, resulting in a new set of data combined with the original data and used to train an AI model

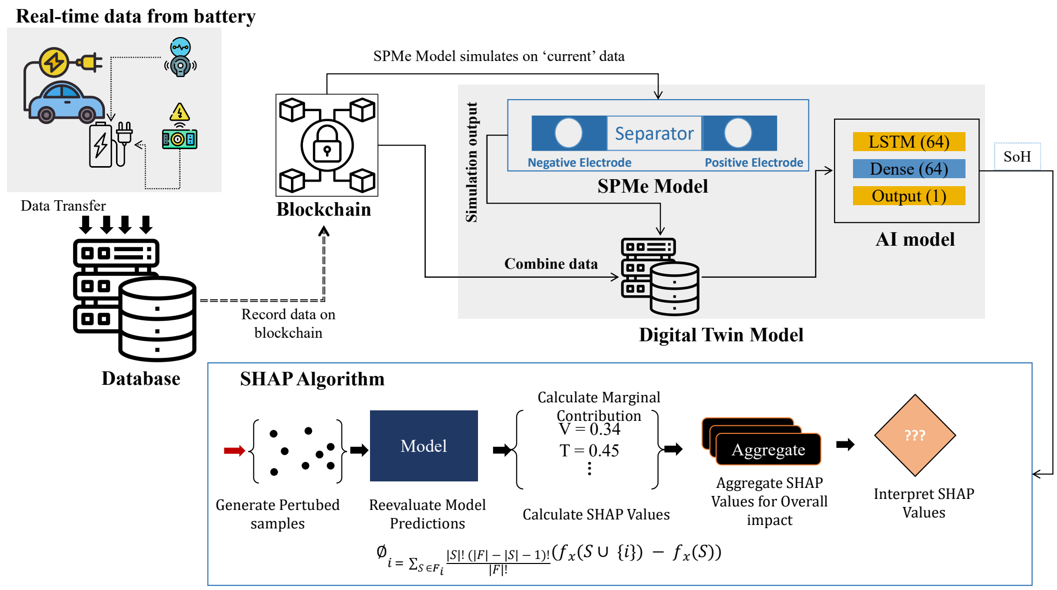


Fig. 1. Framework for Proposed Digital Twin for BMS

to forecast the battery SoH. The SPMe and AI are combined in a hybrid manner to form the digital twin model. The output of the digital twin model, referring to the battery SoH, is then subjected to the SHAP algorithm for evaluation and to identify the features that most significantly influence the output.

B. SPMe for battery modeling

The SPM is a reduced-order electrochemical model that assumes a uniform current density across a battery's electrode. To enhance the model under higher current operations, the SPMe extends the basic SPM by reintroducing electrolyte dynamics, thereby increasing the model fidelity. In the SPMe, the conservation of lithium within the solid phase of the electrode particles is described by Fick's second law of diffusion:

$$\frac{\partial c_s}{\partial t} = \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 D_s \frac{\partial c_s}{\partial r} \right), \tag{1}$$

where c_s represents the concentration of lithium in the solid phase, t is the time, r is the radial position within the electrode particle, and D_s is the solid-phase diffusion coefficient. For the electrolyte phase, the SPMe incorporates the transport of lithium ions, which is governed by the following equations:

$$\frac{\partial c_e}{\partial t} = -\nabla \cdot \mathbf{J}_e, \tag{2}$$

Where c_e is the electrolyte concentration and $\mathbf{J}_e$ is the molar flux of lithium ions in the electrolyte. The electrolyte molar flux is defined by:

$$\mathbf{J}_e = -D_e \nabla c_e - \frac{t_+ c_e}{F} \nabla \phi_e, \tag{3}$$

with D_e being the electrolyte diffusion coefficient, t_+ the transference number, F Faraday's constant, and ϕ_e the electrolyte electric potential.

C. SoH Estimation

In this study, we employed the NASA battery dataset [12] composed of battery datasets representing four distinct batteries subjected to various discharge cycles. We cleaned data to identify missing data points and handle them for consistency. We also conducted feature engineering to generate SoH estimates using the formula:

$$\text{SoH} = \left(\frac{C}{C_i} \right) \times 100. \tag{4}$$

C_f represents the capacity fade and can be represented as:

$$C_f = C_i - C. \tag{5}$$

where, C_i denotes the initial capacity and C is the current battery capacity.

D. LSTM model

A simple LSTM model was used in this study. The LSTM model employed in this study comprises three layers: An LSTM layer, a dense layer, and an output layer. The input features for the LSTM model include real battery data, including current charge, temperature measured, voltage measured, voltage charge, capacity, time, cycle number, resistance increase, and capacity fade. Additionally, outputs of the SPMe simulation like *voltage* and *current* are included to provide the model with rich information on the battery's internal state and reaction to discharging cycles. The model is trained to minimize the mean squared error (MSE) loss function.

E. SHAP XAI method

The SHAP framework offers a cohesive approach for explaining the predictions of AI models [13]. In this study, we employed SHAP to analyze and interpret the outputs of the digital twin model developed to estimate the battery SoH.

The process and implications of SHAP, as discussed in [13], are summarized as follows:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} [f_x(S \cup \{i\}) - f_x(S)]. \quad (6)$$

Where ϕ_i represents the SHAP value for feature i, F denotes the set of all features, S denotes a subset of features excluding i, $f\_x(S \cup \{i\})$ and $f\_x(S)$ represent the model outputs with and without the feature i, respectively, and

Fig. 1 illustrates how the SHAP approach was applied to the digital twin model to explain the SoH predictions.

Due to the presence of the LSTM layer in the model, a custom prediction wrapper function was created. This function reshaped the input data into a 3D format suitable for LSTM, performed the prediction, and reshaped the predictions into a 2D format for compatibility with SHAP.

A subset of the training data (X_train) was selected to create a background dataset for the SHAP explainer. This background was reshaped to align with the LSTM input requirements. A SHAP Kernel Explainer was instantiated using the custom LSTM prediction function and the reshaped background dataset. For the test dataset, denoted by X_test, SHAP values were computed. The dataset was reshaped to align with the LSTM model's input format. The SHAP summary and force plots were employed to visualize and analyze the contribution of individual features to the predictions.

The overall decision impact ratio (DIR), which represents the average impact of the features, was calculated as:

$$\text{DIR} = \frac{1}{n} \sum\_i = 1^n \bar{S}\_i, \quad (7)$$

where $\bar{S}\_i$ is the mean absolute SHAP value of the i^{th} feature.

Moreover, the confidence scores for the model's predictions were computed as follows:

$$C = \max(p\_k), \quad (8)$$

where C represents the confidence scores and p_k is the predicted probability for each class k.

F. Blockchain data recording

A smart contract (SC) based on the Ethereum blockchain was the backbone for storing battery data. The contract is written in Solidity and includes functions to add and retrieve battery data entries. Each entry in the Blockchain includes data such as the battery's cycle number, capacity, resistance increase, capacity fade, etc, and timestamp. Two main functions are defined in the SC, including the $addBatteryData$, which stores the battery's cycle number, capacity, resistance increase, capacity fade, etc, and timestamp and the $getBatteryData$, which retrieves entries based on their storage index in the storage array.

The battery data is initially collected and stored in a CSV file format. A Python script reads this file, processes the data, and interacts with the deployed SC to upload each entry to the blockchain. A Python script parses the CSV file, reading attributes for each battery entry. The script utilizes the Web3.py library to connect to the Ethereum network and calls the $addBatteryData$ function of the SC to upload each parsed entry. The SC is deployed to the Ethereum network using a platform that facilitates testing and migration. Battery data entries are added to the blockchain through transactions initiated by the Python script and by calling the $getBatteryData$ function to retrieve data entries.

Some performance metrics used to analyze and evaluate the efficacy and efficiency of the blockchain recording include:

- Gas costs: The total gas cost G for a transaction was computed as $G = g_u \times g_p$, where g_u is the gas used, and g_p is the gas price.
- Latency: The latency L of a transaction was estimated as $L = T_b \times N$. Where T_b is the average block time or the time it takes to mine one block on the blockchain, and N is the number of blocks that needed confirmation for the transaction to be considered secure.
- Throughput: The throughput Φ is the number of transactions that can be processed per unit of time. It can be approximated as: $\Phi = \frac{1}{T_b \times T_n}$, where T_n is the number of transactions per block.

III. Implementation

A. Digital twin model development

The simulations were conducted using Google Colab platform. The experiments utilized a Colab notebook configured with a Tesla T4 GPU. The simulations were performed using Python 3.7.12. Key libraries used include TensorFlow 2.8.0 and Keras 2.8.0 for AI model development and training, PyBaMM 22.9 for battery model simulations, and SHAP 0.40 for AI model interpretability. The battery *current* and *time* data were used as input in simulating an SPMe battery model to create new features such as *voltage*, *time*, and *current*. Other details of this experiment are described in Table. I.

B. Blockchain data recording

Our research utilized Remix IDE, an open-source web and desktop application, for writing, deploying, and testing SCs directly in a browser.

For local testing and development, we used Ganache, a part of the Truffle Suite, which provides a personal Ethereum blockchain to run tests, execute commands, and inspect state while controlling how the chain operates. This tool was instrumental in allowing us to develop applications with a faster, more customized setup than the main Ethereum network.

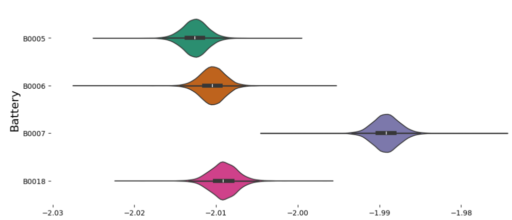
TABLE I
SIMULATION PARAMETERS AND EXPERIMENT SETTINGS

Parameter/Setting	Description/Value
PyBaMM Simulation Parameters	
Battery Model	Single Particle Model with Electrolyte (SPMe)
Solver	CasADi solver with 'safe' mode
Experiment Type	Discharge at C-rate for 1 hour, followed by rest
Number of Cycles	100 cycles with data logging every 10 seconds
Parameter Values	Default values modified for temperature and aging effects
AI Model Training Settings	
Framework	TensorFlow and Keras
Neural Network Type	Long Short-Term Memory (LSTM)
Number of Layers	2 LSTM layers with 64 units each
Activation Function	ReLU for hidden layers, linear for output layer
Loss Function	Mean Squared Error (MSE)
Optimizer	Adam optimizer
Batch Size	32
Number of Epochs	100
Training Data Split	70% training, 20% validation, 10% test
Feature Scaling	Min-Max normalization

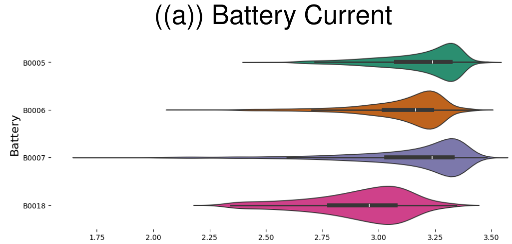
Web3.js was utilized to create client-side applications that interact with the Ethereum blockchain. This JavaScript library enabled our applications to send transactions, interact with SCs, and retrieve data from the blockchain, which is needed for the real-time data recording our project required. More details of the blockchain setup are represented in Table. II

TABLE II
BLOCKCHAIN DEVELOPMENT TOOLS

Tool	Version/Type	Usage
Ethereum	Blockchain Platform	Deployment of SCs
Solidity	0.8.x	SC programming language
Remix IDE	Web version	SC compilation and testing
Ganache	CLI v6.12.2	Local blockchain testing
Truffle Suite	v5.3.x	SC compilation, & deployment
Web3.js	1.4.x	Interface for blockchain interaction
IPFS	-	Decentralized storage solution



((a)) Battery Current



((b)) Battery Voltage

Fig. 2. Illustration of the current and voltage distribution for the batteries.

IV. RESULTS AND DISCUSSION

The experiments conducted were based on two different batteries: $B0005$ and $B0006$. Fig. 2 illustrates the violin plots showing the current and voltage distribution for four batteries in the dataset. From these plots, the batteries have a relatively consistent pattern of current and voltage, with very minimal variations.

The SPMe simulation results for battery $B0005$ is represented in Fig. 3, comparing the simulated and measured battery voltage and current. At the same time, the results for the simulated voltage show some discrepancies compared to the measured voltage; the simulated current shows minimal discrepancies.

The results of the AI model training are presented in Table III and highlight the MSE and MAE results obtained for the two batteries after testing. These results show a minimal error between the predicted battery SoH and the actual battery SoH.

The SHAP summary and force plots are represented in Figs 4 and 5. According to Fig. 4, for both batteries, the *capacity* feature has a high positive impact on the model's output, indicating that higher capacity values contribute to an increase in prediction. The *simulated current* and *voltage* show variability in their impact. In battery $B0005$, the *simulated current* has a predominantly negative contribution. According to Fig. 5, the *base value*, which is the average prediction of the model across all instances, is 75.91 and 76.69 for batteries $B0005$ and $B0006$, respectively. The red features highlight features that push the prediction higher, while the blue features represent features that lower the prediction. The final prediction for the instance is high at 89.66 for battery $B0005$ and lower at 61.62 for battery $B0006$.

TABLE III
SUMMARY OF RESULTS FOR DIGITAL TWIN

Metric	B0005	B0006
Initial Data Shape	45,112, 16	44,354, 16
Cleaned Data Shape	45,112, 10	44,354, 10
Final Data Shape	45,112, 13	44,354, 13
Mean Squared Error (MSE)	0.000407399	0.00120599
Mean Absolute Error (MAE)	0.0153018	0.032276
Average Confidence	77.11832	75.54231
Decision Impact Ratio (DIR)	2.578448	2.5488

TABLE IV
BLOCKCHAIN TRANSACTION RESULTS FOR RECORDING AND QUERYING DATA FOR BATTERIES B0005 AND B0006.

Metric	Battery B0005		Battery B0006	
	Record	Query	Record	Query
Gas Cost (units)	21000	25000	22000	26000
Latency (seconds)	0.15	0.55	0.12	0.55
Throughput (tx/sec)	5	6	6	8

As highlighted in Table IV, recording transactions for battery B0005 costs 21,000 gas units, which aligns with the average benchmarks of the Ethereum network. In contrast, battery B0006 incurred a slightly higher gas cost of 22,000 units, which

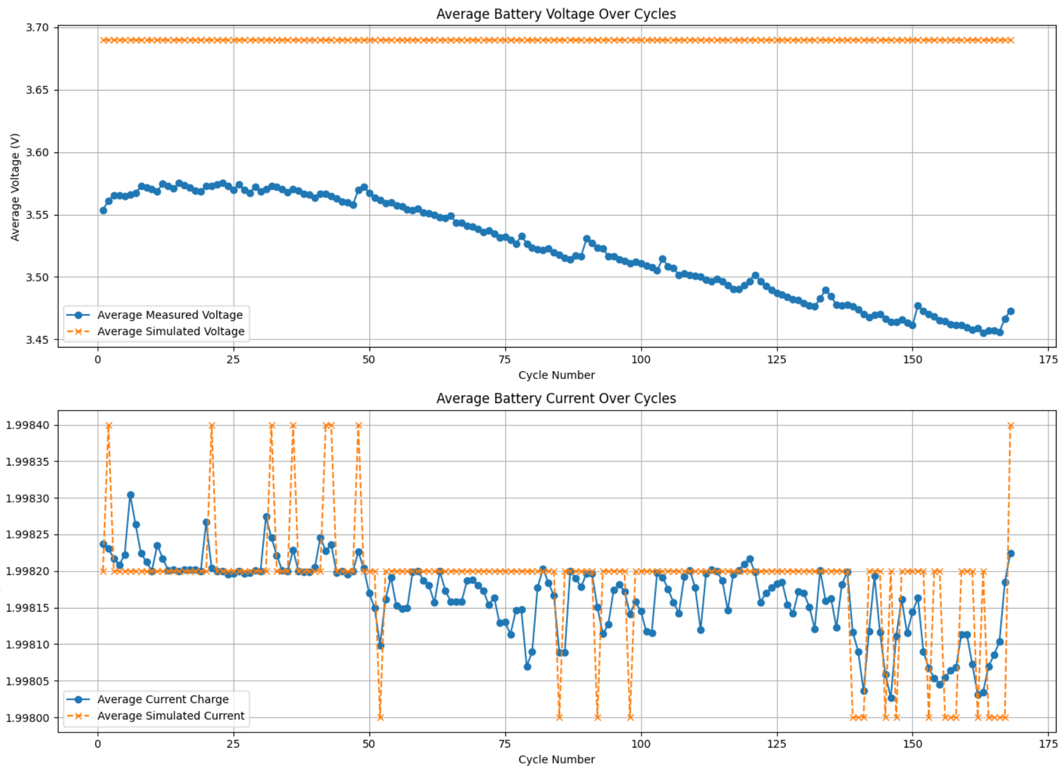
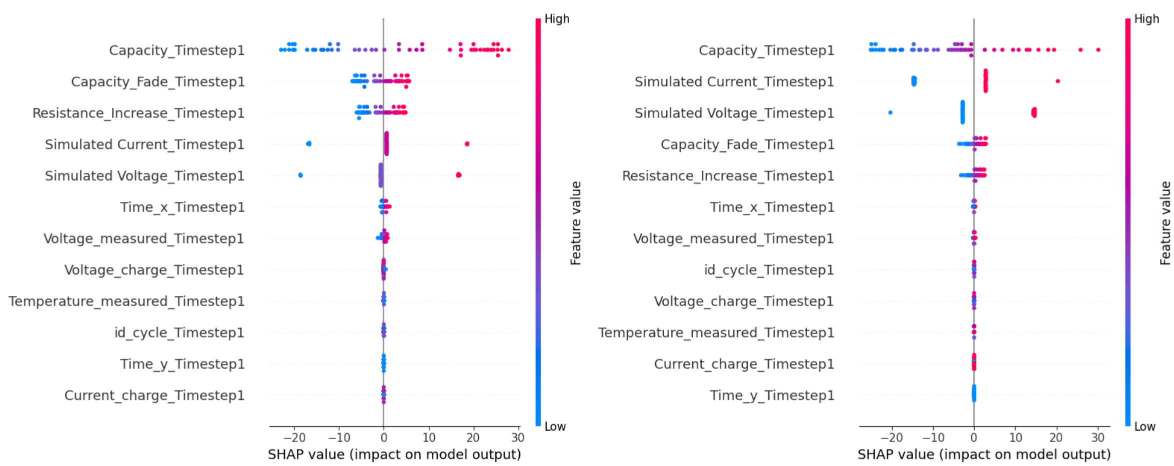


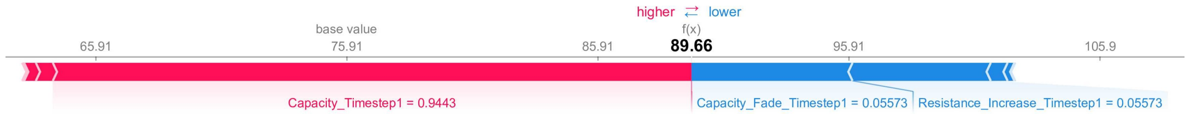
Fig. 3. Simulation results for Battery B0005, comparing simulated versus real battery data for voltage and current over 168 cycles.



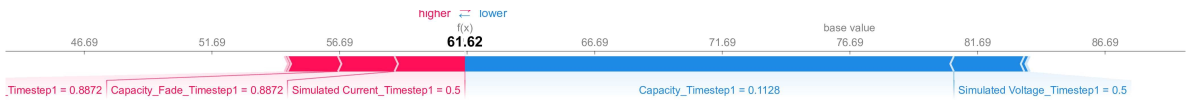
((a)) Battery B0005 ((b)) Battery B0006

Fig. 4. SHAP summary plots for Battery B0005 and Battery B0006, showing the features influencing the SHAP values on the model output.

may suggest a greater volume of transactions or more complex data. When querying the blockchain, battery B0005 exhibited a higher gas cost of 25,000 units compared to recording, which may be attributed to the retrieval of state-modifying transactions. Similarly, battery B0006 saw an increase in gas cost to 26,000 units during queries, further emphasizing the intensive nature of read operations on the Ethereum network. The performance of the two batteries in terms of recording transaction latency was satisfactory, with B0005 and B0006 achieving times of 0.15 seconds and 0.12 seconds, respectively. These figures align with Ethereum's anticipated block times, ensuring minimal delay in data recording. However, the latency



((a)) Battery B0005



((b)) Battery B0006

Fig. 5. SHAP force plots for Battery B0005 and Battery B0006 highlighting the impact of specific features on the model's output.

for querying was higher for both batteries at 0.55 seconds, suggesting that the read operations may involve more complex computational processes or experience temporary network congestion, leading to slower response times. The throughput for recording transactions was consistent, with B0005 and B0006 processing 5 and 6 transactions per second, respectively. This demonstrates the network's efficiency in handling multiple transactions. Additionally, the throughput for querying increased for both batteries, reaching 6 tx/sec for B0005 and 8 tx/sec for B0006, which is essential in high-frequency data environments.

V. Conclusion

This study presents a digital twin framework that integrates SPMe for modeling battery behavior, LSTM for predicting battery SoH, SHAP for interpretability of results, and blockchain for securing battery data. The performance of the SPMe and AI battery modeling shows that the models can reproduce the behavior of the real battery with minimal error. The SHAP XAI method explained the features that contributed to the predictions. Finally, the results of the blockchain recording highlight the potential of blockchain in securing battery data. Future research will investigate methods of improving battery modeling and for better feature engineering.

Acknowledgments

This work was partly supported by Innovative Human Resource Development for Local Intellectualization program through the Institute of IITP grant funded by the Korea government(MSIT) (IITP-2025-RS-2020-II201612, 33%) and by Priority Research Centers Program through the NRF funded by the MEST(2018R1A6A1A03024003, 33%) and by the MSIT, Korea, under the ITRC support program(IITP-2024-RS-2024-00438430, 34%)

References

[1] L. Lu, X. Han, J. Li, J. Hua, and M. Ouyang, "A review on the key issues for lithium-ion battery management in electric vehicles," *Journal of Power Sources*, vol. 226, pp. 272–288, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378775312016163

[2] Y. Zhang, F. Liang, S. Li, C. Zhang, S. Zhang, X. Liu, S. Zhao, S. Yang, Y. Xia, J. Lin, B. Guo, H. Cheng, M. Wang, M. Jiang, and D. Wang, "A review on battery thermal management and its digital improvement-based cyber hierarchy and interactional network," *International Journal of Energy Research*, vol. 46, no. 9, pp. 11 529–11 555, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/er.7957

[3] N. G. Panwar, S. Singh, A. Garg, A. K. Gupta, and L. Gao, "Recent advancements in battery management system for li-ion batteries of electric vehicles: Future role of digital twin, cyber-physical systems, battery swapping technology, and nondestructive testing," *Energy Technology*, vol. 9, no. 8, p. 2000984, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ente.202000984

[4] J. N. Njoku, C. I. Nwakanma, J.-M. Lee, and D.-S. Kim, "Model comparison and selection for battery digital twin development using pybamm," in *The 33rd Joint Conference on Communication and Information (JCCI)*, 2023.

[5] Vandana, A. Garg, and B. K. Panigrahi, "Multi-dimensional digital twin of energy storage system for electric vehicles: A brief review," *Energy Storage*, vol. 3, no. 6, p. e242, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/est2.242

[6] S. Suhail, M. Iqbal, R. Hussain, and R. Jurdak, "Enigma: An explainable digital twin security solution for cyber–physical systems," *Computers in Industry*, vol. 151, p. 103961, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166361523001112

[7] K. Kobayashi and S. B. Alam, "Explainable, interpretable, and trustworthy ai for an intelligent digital twin: A case study on remaining useful life," *Engineering Applications of Artificial Intelligence*, vol. 129, p. 107620, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197623018043

[8] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, D. Pedreschi, and F. Giannotti, "A survey of methods for explaining black box models," 2018. [Online]. Available: https://arxiv.org/abs/1802.01933

[9] E. E.-D. Hemdan, W. El-Shafai, and A. Sayed, "Integrating digital twins with iot-based blockchain: Concept, architecture, challenges, and future scope," *Wireless Personal Communications*, pp. 1–24, 2023.

[10] C. Zhang, L. Zhu, and C. Xu, "Bsdp: Blockchain-based smart parking for digital-twin empowered vehicular sensing networks with privacy protection," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 5, pp. 7237–7246, 2023.

[11] M. Pincheira, M. Vecchio, and F. Antonelli, "Smarttwin: A blockchain-based software framework for digital twins using iot," in *Blockchain and Applications, 4th International Congress*, J. Prieto, F. L. Benítez Martínez, S. Ferretti, D. Arroyo Guardeño, and P. Tomás Nevado-Batalla, Eds. Cham: Springer International Publishing, 2023, pp. 67–77.

[12] B. Saha and K. Goebel, "Battery Data Set," NASA Ames Prognostics Data Repository, 2007, nASA Ames Research Center, Moffett Field, CA. [Online]. Available: http://ti.arc.nasa.gov/project/prognostic-data-repository

[13] N. Kostopoulos, D. Kalogeras, D. Pantazatos, M. Grammatikou, and V. Maglaris, "Shap interpretations of tree and neural network dns classifiers for analyzing dga family characteristics," *IEEE Access*, vol. 11, pp. 61 144–61 160, 2023.