An Innovative Transformer-Based Approach for State of Health Trajectory Prediction and Remaining Useful Life Estimation in Lithium-Ion Batteries

Michele Bellomo
Department of Energy
Politecnico di Milano
Milan, Italy
michele.bellomo@polimi.it

Alberto Dolara
Department of Energy
Politecnico di Milano
Milan, Italy
alberto.dolara@polimi.it

Francesco Grimaccia
Department of Energy
Politecnico di Milano
Milan, Italy
francesco.grimaccia@polimi.it

***Abstract*—In this article, we introduce an innovative approach based on the Transformer neural network architecture for future state of health (SoH) trajectory prediction and remaining useful life (RUL) estimation in lithium-ion batteries. In this methodology, the same model handles predictions for both early and late cycles, thanks to intelligent management of variable-length input and output sequences. Additionally, the future SoH curve is obtained through a one-shot multistep (OSMS) prediction, which enables faster predictions and fewer accumulated errors compared to the standard autoregressive approach. We tested this model on the PoliMi-TUB battery dataset, achieving better results than previous models based on Long Short-Term Memory (LSTM) networks.**

***Index Terms*—time series forecasting, transformer, SoH trajectory, RUL, lithium-ion batteries**

I. Introduction

Neural networks can predict multiple future steps in a sequence or time series using two main strategies: the autoregressive approach, which forecasts one step at a time and uses it as input for the next prediction, and the one-shot multistep (OSMS) approach, where all future steps are predicted simultaneously. The second approach is convenient when the number of future steps to predict is constant and known in advance, and it has the advantages of reduced error accumulation and faster predictions. However, the extension of OSMS approach to predictions involving a variable number of steps, possibly unknown a priori, is not trivial. As demonstrated in [1], filling shorter target sequences with padding and then proceeding with standard training is not a viable solution. This is because the padding value, regardless of its choice, interferes with the predictions themselves. In [2], the authors introduced a novel method based on a custom loss function and compatible with automatic differentiation [3] to mask the padded steps during the computation of the training loss. In this way, the choice of padding value has no effect on the training and the predictive performance. However, this method suffers from a dimensionality problem, as the last state of the recurrent neural network (RNN) is used to predict the entire future sequence, which can potentially be hundreds of steps long. Moreover, even though the output is predicted in a single pass, the input is still processed one step at a time, typically using a Long Short-Term Memory (LSTM) or a Gated Recurrent Unit (GRU), which limits computational performance.

In this article, we expand the framework presented in [2] to the neural network architecture Transformer [4]. Using Transformers instead of RNNs offers several advantages:

- Transformer is the current state of the art for sequence-to-sequence (Seq2Seq) prediction, particularly in the field of Natural Language Processing (NLP), where it has enabled the development of large language models (LLMs) such as ChatGPT [5], LLaMA [6] or DeepSeek [7], which achieve performance that was unimaginable just a few years ago. Beyond NLP, Transformers have also been gaining traction in the domain of numerical time series analysis [8].
- Transformer is able to process all input steps (tokens) in parallel rather than sequentially, as LSTMs or GRUs do. This is particularly relevant in the context of OSMS prediction, where the output is also obtained in a single pass, allowing for excellent inference-time speed.
- Transformer effectively addresses the challenge of high target vector dimensionality in the context of OSMS prediction. Unlike traditional RNN-based approaches, which rely solely on the final hidden state to generate the entire future sequence [2], our method follows a strategy inspired by Informer [9]. Specifically, partial information about future time steps, such as indices or timestamps, is included in input tokens that are deliberately left incomplete, as the information about future is not available and constitutes the prediction target. Transformer then processes these partially specified inputs to infer the missing component (target variable). By aggregating the predictions generated in parallel for each incomplete

This study was developed within the MUSA–Multilayered Urban Sustainability Action–project, funded by the European Union–NextGenerationEU, Project code ECS 00000037, under the National Recovery and Resilience Plan (NRRP) Mission 4 Component 2 Investment Line 1.5: Strenghtening of research structures and creation of R&D "innovation ecosystems", set up of "territorial leaders in R&D".

token, the full forecast of the future sequence is obtained.

We use this innovative approach to simultaneously estimate the future State of Health (SoH) trajectory and Remaining Useful Life (RUL) in lithium-ion batteries. SoH is defined as the ratio of the battery's current capacity to its nominal capacity, while RUL is the number of remaining cycles above a critical SoH threshold required for optimal and safe operation. In this application, we consider the RUL threshold to be 70%.

II. Methods

A. Model architecture

We design an only decoder network architecture with two vanilla Transformer blocks as described in [4], with an inner dimension of 2048 units, 8 attention heads and dropout 0.1. These values were chosen as reasonable for the type of problem and the available dataset, and they have not undergone any optimization, as beyond the scope of this work.

Every token (step) of the input sequences is embedded in a space of dimension 512. This embedding is produced by summing together the embedding related to the number of cycle (position), obtained through a sinusoidal encoding [4], and the encoding related to the SoH value, obtained through a linear dense layer. As in the Informer network, the input sequences also include partially filled components related to future steps. Components related to future steps encode only information about the position (number of cycle), as the corresponding SoH values are unavailable and have to be predicted.

The model architecture is summarized in Figure 1.

The network is implemented using `Keras` [10] and `KerasHub` [11].

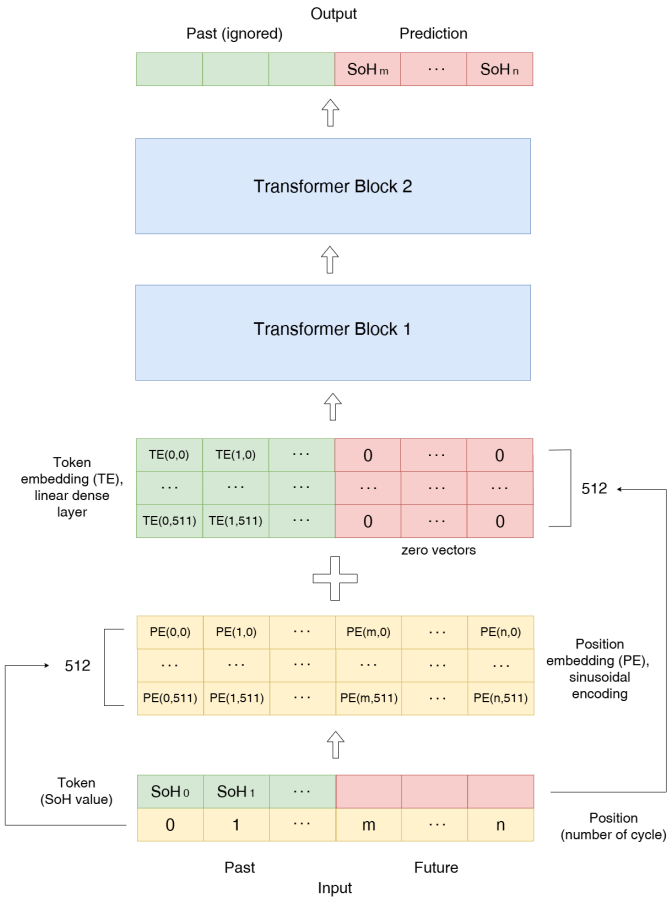


Fig. 1. Designed network architecture

B. Dataset

We use the PoliMi-TUB dataset to train and test the model. This dataset contains degradation data regarding six LG 2.5 Ah 18650 NMC battery cells recorded under various conditions. For more information on the applied cycles, temperatures, and measurements, please refer to [12].

C. Training and evaluation

The network is trained using an extension of the loss described in [2], optimized through gradient descent. The new loss not only ignores errors on steps with true SoH < 0.7, but also masks the target output components related to past steps, whose values have already been observed. The loss is minimized using Adam optimizer with a 0.001 learning rate, a batch size of 32, and an early stopping mechanism with patience 5 on validation error. No optimization was made on these parameters, as outside the scope of this analysis.

Given the small number of cells, we use a cross-validation-like approach to evaluate the model's performance in a robust way. Specifically, we train six models on six different datasets. Each dataset is obtained by reserving the data from one cell as a test set and keeping the data from the other five cells for training and validation, with the proportions 80% for training and 20% for validation.

After training, the performance of the six models are evaluated by comparing the predictions on the relative test sets with the true data. To compare errors in SoH trajectories, the pairs of predicted/true SoH sequences are cut as soon one of the two reaches 70% SoH, and the Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) are calculated on the cut sequences. The estimated numbers of remaining useful cycles, equal to the remaining numbers of cycles before reaching 70% SoH, is calculated directly from the uncut predicted sequence and compared with the same quantity calculated on the true observed sequences using MAE.

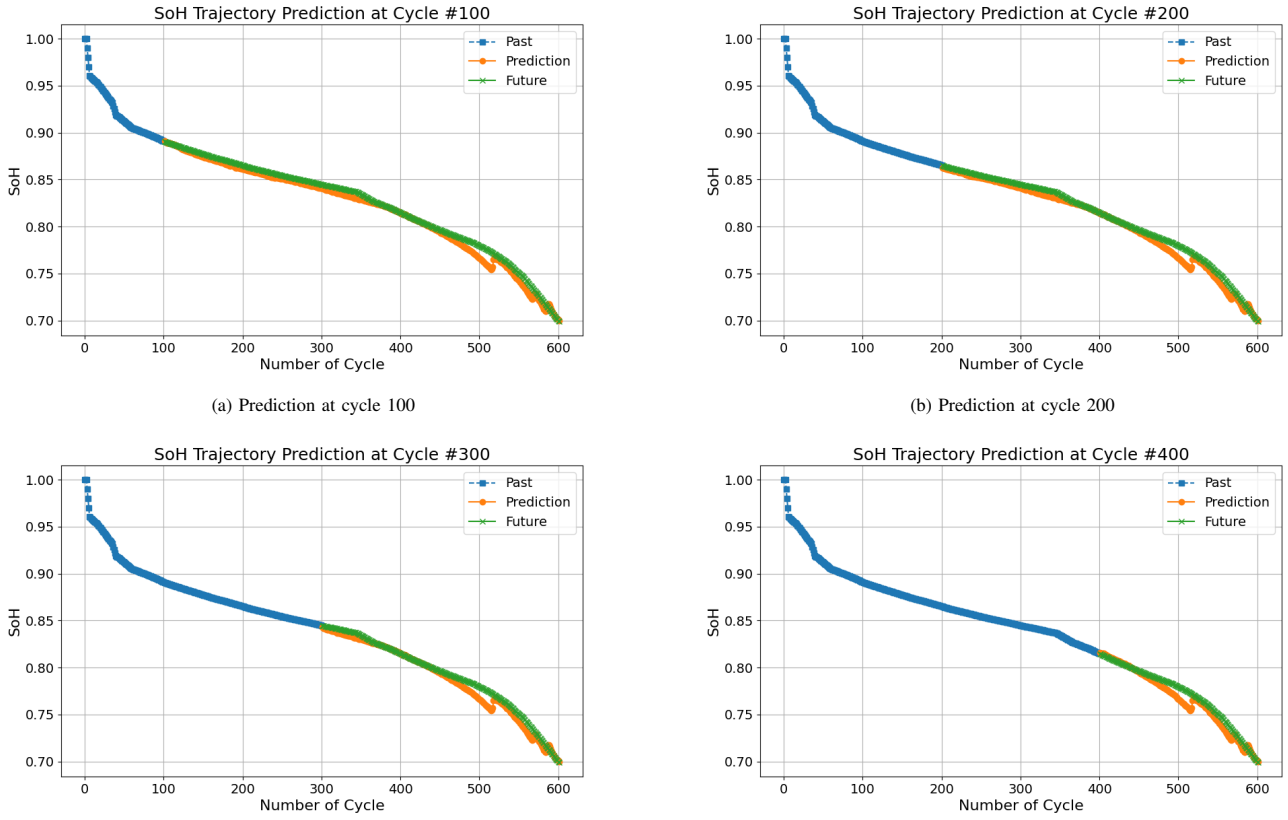
III. Results

The overall RMSE and MAE on SoH trajectory predictions, calculated by averaging the errors obtained in the different test sets, are respectively 0.0165 and 0.0129. The overall MAE on the RUL is 25.70 cycles.

The predictions obtained for a test set cell at different cycle numbers are shown in Figure 2.

IV. Discussion

The RMSE and MAE obtained with this new Transformer model are comparable to those obtained using an LSTM-based model in [2] on the same dataset with the same cross-validation method. However, there is a significant improve-



(a) Prediction at cycle 100

(b) Prediction at cycle 200

(c) Prediction at cycle 300

(d) Prediction at cycle 400

Fig. 2. SoH trajectory predictions at different cycle numbers on a test cell

ment in the RUL estimation extracted from the SoH trajectory, with an approximately halved average error. Moreover, a visual analysis of the predictions shows that they are accurate regardless of the initial cycle number, whereas in the model presented in [2], a paradoxical effect was observed whereby the predictions worsened as the number of cycles increased.

From the plot in Figure 2, it can be observed that, although fairly accurate, the predicted SoH trajectories exhibit over a short segment an unusual non-monotonic behavior. This behavior is likely due to the limited number of samples in the dataset and may disappear as more data become available. However, this problem can be addressed even in a small data regime by adding a regularization term to the loss function that penalizes non-monotonic predictions.

V. Conclusion and Future Work

In this article, we presented a new state-of-the-art methodology based on Transformers for the simultaneous prediction of the future SoH trajectory and the RUL estimation. This approach enables predictions of variable and a priori unknown length, allowing the same model to be used regardless of the number of past cycles. Moreover, predictions are obtained in a single shot, reducing both accumulation error and inference time. We compared the results with those of an LSTM-based model, achieving better performance, especially in terms of RUL estimation.

In future work, we plan to incorporate additional information into the model, such as temperature and discharge rate at which the past cycles were performed. We also intend to expand the dataset used for training and testing, potentially including cells of different types and from different manufacturers.

References

[1] M. Bellomo, S. Giazitzis, S. Badha, F. Rosetti, E. Ogliari, A. Dolara, and F. Grimaccia, "A novel data-driven approach for the simultaneous prediction of state of health trajectory and remaining useful life in lithium-ion batteries," in *2024 IEEE International Conference on Environment and Electrical Engineering and 2024 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*. IEEE, 2024, pp. 1–5.

[2] M. Bellomo, S. Giazitzis, S. Badha, F. Rosetti, A. Dolara, and E. Ogliari, "Deep learning regression with sequences of different length: An application for state of health trajectory prediction and remaining useful life estimation in lithium-ion batteries," *Batteries*, vol. 10, no. 8, p. 292, 2024.

[3] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, "Automatic differentiation in machine learning: a survey," *Journal of machine learning research*, vol. 18, no. 153, pp. 1–43, 2018.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[5] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[6] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[7] D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi *et al.*, "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning," *arXiv preprint arXiv:2501.12948*, 2025.

[8] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, "Transformers in time series: A survey," *arXiv preprint arXiv:2202.07125*, 2022.

[9] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[10] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[11] M. Watson, F. Chollet, D. Sreepathihalli, S. Saadat, R. Sampath, G. Rasskin, , S. Zhu, V. Singh, L. Wood, Z. Tan, I. Stenbit, C. Qian, J. Bischof *et al.*, "Kerashub," https://github.com/keras-team/keras-hub, 2024.

[12] P. Eleftheriadis, "PoliMi-TUB dataset - LG 18650HE4 li-ion battery," Mendeley Data, V1, 2024. [Online]. Available: https://data.mendeley.com/datasets/6hyhsjbwkb/1