

Adapting Amidst Degradation: Cross Domain Li-ion Battery Health Estimation via Physics-Guided Test-Time Training

Anonymous Author(s)*

Abstract

Health modeling of lithium-ion batteries (LIBs) is crucial for safe and efficient energy management and carries significant socio-economic implications. Although Machine Learning (ML)-based State of Health (SOH) estimation methods have made significant progress in accuracy, the scarcity of high-quality LIB data remains a major obstacle. Although existing transfer learning methods for cross-domain LIB SOH estimation have significantly alleviated the labeling burden of target LIB data, they still require sufficient unlabeled target data (UTD) for effective adaptation to the target domain. Collecting this UTD is challenging due to the time-consuming nature of degradation experiments. To address this issue, we introduce a *practical* Test-Time Training framework, **BatteryTTT**, which *adapts the model continually using each UTD collected amidst degradation*, thereby significantly reducing data collection time. To fully utilize each UTD, BatteryTTT integrates the inherent physical laws of modern LIBs into self-supervised learning, termed **Physics-Guided Test-Time Training**. Additionally, we explore the potential of large language models (LLMs) in battery sequence modeling by evaluating their performance in SOH estimation through model reprogramming and prefix prompt adaptation. The combination of BatteryTTT and LLM modeling, termed **GPT4Battery**, achieves state-of-the-art generalization results across current LIB benchmarks. Furthermore, we demonstrate the practical value and scalability of our approach by deploying it in our real-world battery management system (BMS) for 300Ah large-scale energy storage LIBs.

CCS Concepts

• Test Time Training → Battery Health Estimation.

Keywords

Battery Health Estimation, Test Time Training, Data Scarcity, Large Language Model

ACM Reference Format:

Anonymous Author(s). 2018. Adapting Amidst Degradation: Cross Domain Li-ion Battery Health Estimation via Physics-Guided Test-Time Training. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 10 pages. <https://doi.org/XXXXXXX.XXXXXXX>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-XXXX-X/18/06

<https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

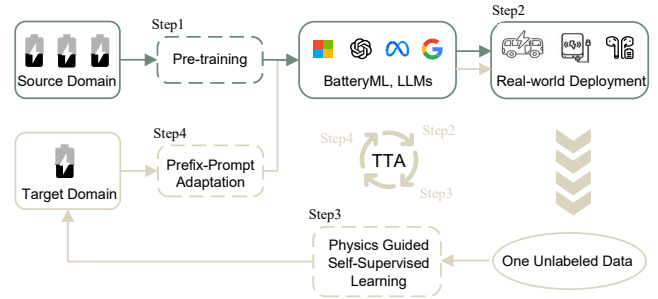


Figure 1: Overview of BatteryTTT framework, which consists of three major components: (Step 1) pre-training on experimental datasets; (Step 2) incremental data collection after deployment; and (Steps 3-4) test-time adaptation. Steps 2, 3, and 4 iterate until the LIB retires.

The rapid advancements in rechargeable Li-ion batteries (LIBs) have facilitated their widespread use across various sectors, including portable electronics, medical devices, renewable energy systems, and electric vehicles [11]. This ubiquity, however, introduces critical challenges associated with capacity degradation and performance evaluation. As an inherently interdisciplinary subject, battery aging modeling has emerged as a fundamental issue at the intersection of battery science and machine learning (ML) [25, 32, 33, 51]. Accurate State of Health (SOH) estimation for LIBs is crucial not only for ensuring safe and efficient energy management but also for optimizing the design and performance of next-generation batteries, thus having significant socio-economic implications.

With the rapid advancement of ML technology, data-driven SOH estimation models have achieved significant progress in both accuracy and computational efficiency [32, 51]. However, obtaining sufficient training data for LIBs is challenging due to the time-consuming nature of degradation experiments, which typically span months to years. Additionally, precisely labeling this data necessitates additional cycles under controlled temperature and current conditions in the laboratory. Consequently, the scarcity of high-quality LIB data continues to present a major obstacle in battery aging modeling [22, 33, 51].

To address the challenge of data scarcity, existing studies have extensively explored transfer learning methods for cross-domain LIB SOH estimation [22, 35, 39, 41, 46]. For instance, SSF-WL [46] introduced a self-supervised approach that pre-trains on unlabeled LIB data from a source domain and fine-tunes on a different type of LIB (target domain) using only a small portion of labeled data. This method achieves comparable results using just 30% of the labeled target data. Another notable study, woDegrad [22], proposed minimizing the domain gap between source and target LIBs by aligning

Table 1: Comparison between BatteryTTT and other transfer learning methods regarding the target LIB dataset prerequisites. Note: the time estimates are based on the KOKAM dataset [3].

| Methods | woDegrad | SSF-WL | BatteryTTT(Ours) |
|-----------------|------------|------------|----------------------|
| Unlabeled Data | 100% | 30% | 1 amidst degradation |
| Labeling | 0% | 30% | 0% |
| Collection Time | 8473 hours | 2542 hours | 0 hours |

their features in the latent space, eliminating the need for additional labeled target data to estimate SOH.

Unfortunately, while existing studies have significantly alleviated the burden of data labeling, they often overlook the challenge of collecting unlabeled target data (UTD). Due to the time-consuming nature of degradation experiments, gathering sufficient UTD for a typical LIB can take months to years, depending on the battery type. As a result, preparing sufficient UTD for previous algorithms to function effectively also demands a considerable amount of time. For example, migrating a pre-trained woDegrad model [22] to the KOKAM dataset [3], a widely used LIB dataset, would require a minimum of 8,473 hours to collect enough UTD from KOKAM for effective domain gap alignment, which is hardly practical in real-world applications. To the best of our knowledge, few studies have successfully estimated SOH without relying on a substantial amount of UTD.

The purpose of this study is to develop a *practical* transfer learning framework for cross-domain LIB SOH estimation that minimizes the reliance on UTDs. Inspired by the Test-Time Training (TTT) ¹ technique from computer vision [10, 38], we propose **BatteryTTT**. Unlike previous methods that require a substantial amount of UTDs collected over time, BatteryTTT adapts the model continually using each *individual UTD collected amidst degradation*. In Table 1, we compare the prerequisites of the target LIB dataset for BatteryTTT with those of other transfer learning methods. BatteryTTT significantly reduces the amount of required UTD and labeling, offering a more efficient approach compared to existing methods. To achieve this, BatteryTTT employs a proxy unsupervised task to utilize the UTD for gradient-based model updates. Although some unsupervised methods from existing TTT literature can be applied, such as self-prediction, they are not specifically designed for battery-related tasks, leading to suboptimal performance. To address this, we explore the integration of the inherent physical laws of modern LIBs into self-supervised learning within BatteryTTT, a framework we term **Physics-Guided Test-Time Training**. This approach leverages the 1-RC Equivalent Circuit Model (ECM) equations to guide the pre-trained model in making accurate self-predictions, leading to improved results. The details of our method and experimental results are discussed in the following sections.

On the other hand, with the rapid advancement of large language models (LLMs) [8, 28, 29, 42] there has been growing interest in exploring their potential for processing *cross-disciplinary sequence data* beyond natural language, including protein sequence prediction [23] and time series analysis [19, 54]. Pioneering research has validated the efficacy of this paradigm and highlighted the

underlying zero-shot generalization capabilities of LLMs across various datasets. However, the application of LLMs to battery sequence modeling remains undiscovered. To address this gap, we evaluate the performance of LLMs for processing cross-domain LIB sequences in this study. Specifically, We employ the concept of model reprogramming to bridge the gap between language and battery modalities. Additionally, we develop a prefix prompt adaptation strategy to efficiently integrate an LLM into our BatteryTTT framework. This combination of strategies, termed **GPT4Battery**, achieves state-of-the-art generalization results among current LIB benchmarks.

By introducing the BatteryTTT framework and GPT4Battery model, we hope this study will inspire the research community to fully leverage advanced AI techniques, such as Test-Time Training (TTT) and large language models (LLMs), for more scenario-fitting AI4Science problems, thereby saving enormous time and accelerating scientific discovery. For the rest of this paper, Sec.2 presents background and related works, Sec. 3 introduces the preliminaries, Sec. 4 describes the methodology of BatteryTTT framework and GPT4Battery model, Sec. 5 conducts experiments, and Sec. 6 concludes.

2 Related Work

In this section, we introduce the related works in *Data-Driven Battery SOH Estimation* (Sec. 2.1), *Test-Time Adaptation* (Sec. 2.2) and *LLM for Cross-disciplinary Sequence Modeling* (Sec. 2.3), respectively.

2.1 Data-Driven Battery SOH Estimation

Data-driven battery SOH estimation as ascended as a pivotal topic in industrial artificial intelligence and data mining with the widespread adoption of modern LIBs in various applications, bringing a surge in demand for safe and efficient battery management[25, 32]. With the evolution of model architectures within the AI community, algorithms for battery state estimation have also progressed from statistical machine learning methods, such as Random Forest [4] and Gaussian Process Regression [30], to high-performance deep neural networks, including MLP [14], LSTM [16], and Transformer [51].

However, to obtain both battery training data and ground-truth labels requires time- and resource-consuming degradation experiments, posing a persistent hurdle in battery aging modeling [22, 32, 33, 51]. Noticing this significant issue, researchers in the battery field have tried to use transfer learning methods for generalizable SOH estimation [22, 35, 39, 41, 46]. However, there is a notable gap between current methods' assumptions about having access to the target LIB dataset and the real-world situation, especially regarding the unlabeled target data (UTD). while existing studies have significantly alleviated the burden of data labeling, they often overlook the challenge of collecting UTDs. Due to the time-consuming nature of degradation experiments, gathering sufficient UTD for a typical LIB can take months to years, depending on the battery type. This cost of time is unacceptable in real-world deployment (as shown in Table 1). Conversely, BatteryTTT fully utilizes each individual UTD collected amidst degradation for adaptation to the target domain, thereby significantly reducing data collection time

¹In the following sections, the terms Test-Time Training (TTT) and Test-Time Adaptation (TTA) may be used interchangeably.

2.2 Test-Time Training

Test-time Training (TTT)/Adaptation (TTA), also known as one-sample unsupervised domain adaptation, aims to adapt a model trained on the source domain to the target domain as every unlabeled test sample arrives [20, 38]. The process of TTA usually involves a self-supervised loss to extract information from the single target domain sample, such as rotation [38], mask [10] and entropy minimization [45]. Despite the study of various TTA methods, most are designed for (image) classification and cannot be applied to time series regression. For instance, Test-time Entropy Minimization (Tent) [45] found that the entropy of prediction strongly correlates with accuracy on the target domain, BACS [53], MEMO [52], and EATA [26] follow Tent’s approach and improve adaptation performance, making them the most representative TTA methods. Although some unsupervised methods from existing TTT literature can be applied, such as self-prediction, they are not specifically designed for battery-related tasks, leading to suboptimal performance. In this paper, we explore how to incorporate the inherent physics of LIBs into self-supervised learning, resulting in a more natural and powerful approach.

2.3 LLM for Cross-Disciplinary Sequence Modeling

With the rapid advancement of large language models (LLMs) [8, 28, 29, 42] there has been growing interest in exploring their potential for processing *cross-disciplinary sequence data* beyond natural language, including protein sequence prediction [23] and time series analysis [6, 13, 36, 54]. For protein sequence prediction, LLM have showed powerful cross-domain potential against protein language models by the design of vocabulary [9, 31]. For time series, these efforts have evolved from the initial direct application of large language models (LLMs) to sequence tasks [54], to designing a learned dictionary of prompts to guide inference [5], to attempting to align the semantic spaces between language and time series modalities [18, 27]. Although the effectiveness of large language models (LLMs) in handling cross-disciplinary sequence data has been demonstrated, the field of battery research has yet to benefit from this advancement. In this paper, we address this gap by repurposing an LLM for State of Health (SOH) estimation through model reprogramming. Additionally, we evaluate the generalization improvements achieved by adapting an LLM for cross-battery SOH estimation.

3 Preliminaries

3.1 Battery SOH Definition

As batteries undergo repeated charge and discharge cycles, their capacity gradually declines due to aging, which leads to performance degradation and potential safety issues. The State of Health (SOH) quantifies the battery’s remaining capacity relative to its initial capacity when new. Specifically, if we denote the nominal capacity of the LIB as C_{norm} and the full discharge capacity in the current cycle as C_{full} , the SOH is defined as the ratio of these two values, expressed as a percentage:

$$SOH = \frac{C_{full}}{C_{nom}} \times 100\%. \quad (1)$$

LIBs are typically considered to have reached the end of their life cycle when their SOH drops to approximately 75%.

3.2 Feature Engineering

In this study, we utilize *QdLinear* [33], a degradation feature derived from the linear interpolation of the voltage-capacity curve during charge cycles, to map the relationship with SOH. This feature is widely recognized and employed by mainstream SOH estimation algorithms [22, 25, 33, 51].

3.3 Problem Definition

We formalize the problem of cross-domain LIB SOH estimation as follows. Given a well-curated battery dataset from the source domain, denoted as $\mathcal{S} = \{(x_1, y_1), (x_2, y_2), \dots, (x_S, y_S)\}$, where $\mathbf{X} \in \mathbb{R}^{1 \times T}$ represents the extracted *QdLinear* feature with T time steps, and y denotes the corresponding SOH label. This source set contains S labeled lifelong samples. In contrast, for a different battery type in the target domain, we can acquire *only one* unlabeled feature at a time after real-world deployment, represented as $\mathcal{T} = \{x_1, x_2, \dots, x_T\}$. Our objective is to estimate each corresponding target label y_t , with y_1 being considered as SOH = 100% for a new battery.

4 Methodology

In this section, we present the methodology of our framework. After providing a systematic overview in Section 4.1, we focus on two key innovations: Physics-Guided Self-Supervised Learning (PG-SSL) and Prefix Prompt Adaptation (PPA), which are detailed in Sections 4.2 and 4.3. In Section 4.4, we explain how existing State of Health (SOH) estimation models are integrated into the BatteryTTT framework for cross-domain transfer learning, alongside the exploration of Large Language Models (LLMs) for battery sequence modeling.

4.1 System Overview

Figure 1 depicts an overview of the BatteryTTT framework, which is composed of three major components: *pre-training on experimental datasets*, *incremental data collection in real-world deployment*, and *test-time adaptation*. Firstly, we utilize experimental datasets (source domain) to train a pre-trained model for SOH estimation. We then deploy this pre-trained model to real-world devices, such as the Battery Management System (BMS) of an electric car or a mobile phone. The BMS individually collects unlabeled test data during usage, and upon receiving a sample, we conduct a *test-time training* process to adapt the pre-trained model to this different type of LIB (target domain). Specifically, the test-time adaptation process consists of two steps: PG-SSL to construct an unsupervised loss from the unlabeled sample, and PPA to adapt the pre-trained model to this new domain in a parameter-efficient manner. After learning from incoming data, the adapted model is ready to make a prediction. This process operates continually until the LIB retires.

4.2 Physics-Guided Self-supervised Learning

In this subsection, we elaborate on how to fully exploit the inherent physical laws of a *single* LIB sequence to improve TTA performance through the design of physics-guided self-supervised

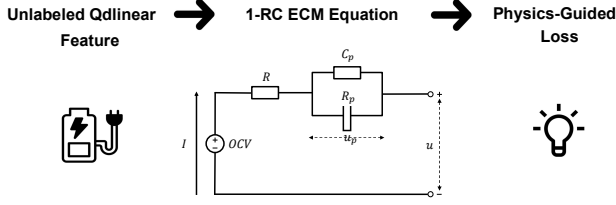


Figure 2: Transform an arbitrary unlabeled Qdlinear feature into a Physics-Guided loss.

learning (PG-SSL). Specifically, we first describe how to use the 1-RC ECM equation to transform an arbitrary unlabeled Qdlinear feature into a physics-guided loss (PGLoss). We will then explain how minimizing this PGLoss facilitates the estimation of the LIB SOH.

4.2.1 Thevenin's Equivalent Circuit Model (ECM) of LIB. The equivalent circuit model (ECM) is widely used battery model to describe the electrical behavior of the battery in terms of voltages, currents, resistances and capacitances [48] [43]. The first order Thevenin model is thought to be accurate and adequate to model the condition of the battery, and at the same time simple and computationally efficient [49]. The OCV is represented by an ideal voltage source of the battery. R accounts for the internal ohmic resistance. The parallel RC-branch, comprising R_p and C_p , is used to model battery polarization effect. u and I denotes the terminal voltage and current that can be collected in use. Based on Kirchhoff's law, the electrical behavior of the battery can be characterized as physical equations as:

$$u_{OCV} = u_R + u_p + u \quad (2)$$

$$\frac{R + R_p}{C_p R_p} I + \frac{1}{C_p R_p} u + \dot{u} = 0 \quad (3)$$

We define $\theta_1 = \frac{R+R_p}{C_p R_p}$ and $\theta_2 = \frac{1}{C_p R_p}$. Following recent works, the coefficients are functions of temperature. The state function becomes:

$$\theta_1(T)I + \theta_2(T)u + \dot{u} = 0 \quad (4)$$

This implies that the terminal voltage u , current I and temperature should, in principle, follow the ordinary differential equation (ODE) functions representing the battery's physical state, as described in Equation 4. By querying the real-time current and temperature value from the BMS, we can transform an arbitrary unlabeled Qdlinear feature into a PGLoss in an unsupervised manner.

4.2.2 Generate a complete Qdlinear feature and supervise it with PGLoss. In the field of battery research, a complete Qdlinear curve spanning from the lower to the upper voltage limits can describe LIB's aging mode and therefore can theoretically identify the accurate state of health [50] [7] [40] according to its definition². Inspired

²This complete Qdlinear (voltage-capacity) curve requires additional cycles under constrained temperature and current conditions in the laboratory and infeasible at deployment. The unlabeled Qdlinear feature we obtained at use is always a partial of it.

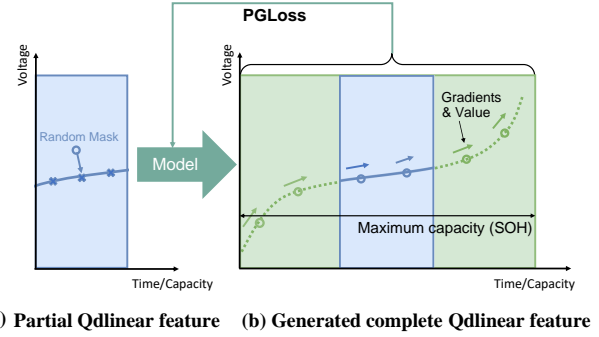


Figure 3: Guide the pre-trained model to generate a complete Qdlinear feature curve and supervise it with Physics-Guided loss.

by this, we want to design the objective by guiding the pre-trained model to generate a complete Qdlinear feature from the given partial one, which means to generate Figure 3 (b) from (a).

Specifically, given the input (partial) Qdlinear feature curve $\mathbf{x} \in \mathbb{R}^{1 \times T}$, we use the pre-trained model to generate a complete one $\hat{\mathbf{x}} \in \mathbb{R}^{1 \times T'}$, where $T' > T$. This complete voltage-capacity curve is then supervised with PGLoss (as shown in Figure 3). Additionally, we randomly mask a portion of \mathbf{x} to promote representation learning. In general, our objective can be formulated in the form of:

$$\mathcal{L}_{PG-SSL} = \|\hat{\mathbf{x}}[0 : T] - \mathbf{x}\|_F^2 + \lambda \|\theta_1 I + \theta_2 \hat{u} + \dot{\hat{u}} = 0\|_F^2 \quad (5)$$

Here, $\hat{\mathbf{x}}[0 : T]$ represents the overlap between the generated complete voltage-capacity curve and the given \mathbf{x} . \hat{u} denotes the generated complete voltage-time curve. The parameters θ_1 and θ_2 are associated with temperature, and I denotes the current; all of these values can be queried through the BMS during deployment. We will empirically demonstrate that the design of Physics-Guided PG-SSL is highly suitable for our SOH estimation task and leads to better performance than simple self-prediction, as shown in the experimental section.

4.3 Prefix Prompt Adaptation

In this subsection, we introduce Prefix Prompt Adaptation (PPA), a method that reduces the dimension of the solution space for easier optimization of test-time adaptation. Specifically, inspired by the demonstrated effectiveness of continuous prompt learning in the field of deep model fine-tuning [2, 17]. We add a small number of soft prompts as a prefix to the embedded input context for test-time updating, while keeping all other model parameters frozen. In this way, the dimension of learnable model parameters shall be significantly reduced and thus enabling the practical deployment of TTA on real-world edge devices such as a CPU when the pre-trained model is large (such as an LLM). Formally, given a test sample X_{test} , our goal is to find an optimal prompt \mathbf{p}^* :

$$\mathbf{p}^* = \arg \min_{\mathbf{p}} \mathcal{L}_{PGTPT}(\mathcal{F}, \mathbf{p}, X_{test}) \quad (6)$$

using the physics-guided self-supervised loss $\mathcal{L}_{\text{PGTPT}}$ from Equation 4. Here, \mathcal{F} denotes the pre-trained model. We will also demonstrate experimentally that PPA significantly reduces the cost of TTA while only marginally reducing its performance.

4.4 Integrate Existing SOH Estimation Models into BatteryTTT

In this subsection, we demonstrate how to incorporate existing SOH estimation algorithms [51] into our test-time adaptation framework to complete the whole cross-domain SOH estimation process depicted in Figure 1. Existing SOH estimation studies [32, 51] extensively use statistical models and high-performance neural networks for LIB SOH estimation, such as Gaussian Process Regression [47], Random Forest [12], MLP [14], RNNs[16] and Transformer [44] for SOH estimation. Only deep learning methods can be incorporated into the TTA framework, as well as other transfer learning settings.

Specifically, our architecture follows a Y-shaped design, as described in [10, 34]: a feature extractor f is simultaneously followed by a self-supervised head g and a main task head h . Here, we substitute f with the encoder of existing neural networks for SOH estimation, such as GRU, LSTM, and Transformer, and g with the decoder. For the main regression task head h for SOH estimation, we use a linear projection from the dimension of the encoder features to 1, which is primarily a historic artifact [15].

During pre-training, we first train $g \circ f$ using the PG-SSL loss in Equation 5 in an unsupervised manner with the features of the source LIB dataset. Then we perform linear probing by combining the encoder f with the main task head h , keeping f frozen. Formally:

$$f_0, g_0 = \arg \min_{f, g} \frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\text{PG-SSL}}(x_i; f, g) \quad (7)$$

Followed by:

$$h_0 = \arg \min_h \frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\text{MSE}}(h \circ f_0(x_i), y_i) \quad (8)$$

The summation is over the training set with n samples, each consisting of input x_i and label y_i . During test-time adaptation, we optimize $g \circ f$ before making a prediction each time a test input x arrives. After optimization, we make a prediction on x as $h \circ f_x(x)$, formally as:

$$f_x, g_x = \arg \min_{f, g} \mathcal{L}_{\text{PG-SSL}}(x; f_0, g_0) \quad (9)$$

$$\text{prediction} = h_0 \circ f_x(x) \quad (10)$$

In particular, we argue for the use of our proposed prefix prompt adaptation in Equation 6 to avoid the substantial computation brought by fine-tuning Equation 9 at inference time.

4.5 Integrate an LLM into BatteryTTT (GPT4Battery)

In this subsection, we explore the potential of LLM in battery sequence modeling. Inspired by recent studies which utilize an LLM for protein sequence prediction [23] and time series analysis [18], we primarily employ the idea of *model reprogramming* to effectively

align the modalities of battery data and natural language, leveraging the reasoning and generalization abilities of LLMs for battery tasks.

Specifically, we first tokenize and map the input Qdlinear features into a high-dimensional space with the same dimensionality as the word space of the language model. Then, we fuse the information of battery sequence modality and language modality using a cross-attention layer.

However, for a word embedding space of an LLM $\mathbf{E} \in \mathbb{R}^{V \times D}$, where V is the vocabulary size. The vocabulary size can be inevitably large (for example, GPT2 has a V of 50257 [29]). Simply leveraging \mathbf{E} will result in large and potentially dense reprogramming space, increasing the computation complexity and difficulty of catching the relevant source tokens. Following [37] and [18], we maintain only a small collection of text prototypes by linearly probing \mathbf{E} , denoted as $\mathbf{E}' \in \mathbb{R}^{V' \times D}$, where $V' \ll V$. Then, we align the tokenized input patches and text prototypes with a multi-head cross-attention layer. Specifically:

$$\begin{aligned} \mathbf{Z}_k^{(i)} &= \text{attention}(\mathbf{Q}_k^{(i)}, \mathbf{K}_k^{(i)}, \mathbf{V}_k^{(i)}) \\ &= \text{softmax}\left(\frac{\mathbf{Q}_k^{(i)} \mathbf{K}_k^{(i)\top}}{\sqrt{d_k}}\right) \mathbf{V}_k^{(i)} \end{aligned}$$

By aggregating each $\mathbf{Z}_k^{(i)} \in \mathbb{R}^{P \times d}$ in every head, we obtain $\mathbf{Z}^{(i)} \in \mathbb{R}^{P \times D}$. This way, the text prototypes can learn cues in language which can then represent the relevant local patch information. We will experimentally demonstrate the improvement in generalizability achieved by incorporating LLMs and the effectiveness of model reprogramming.

5 Experiments

In this section, we empirically evaluate the proposed approach on six real-world LIB datasets, including five publicly available datasets for daily applications (with capacities ranging from a few Ah) and our own collected 300Ah large LIB dataset for energy storage. Specifically, we focus on (1) the overall improved generalization performance of TTA and the superior performance by combining GPT-2 and TTA (GPT4Battery), (2) an efficacy study of the two proposed designs, PG-SSL and PPA, (3) the ablation results of GPT4Battery, (4) inference efficiency of involving TTA, and (5) the scalability and deployment of our method on large energy storage LIBs.

5.1 Experiment Settings

5.1.1 Dataset preparation. We conducted experiments using five publicly available lithium-ion battery (LIB) datasets intended for daily commercial use, with capacities ranging from 0.74 Ah to 27 Ah. Additionally, we utilized four of our own LIB datasets collected for industrial-level energy storage, specifically with a capacity of 300 Ah, which we will also make publicly available for academic purposes. These datasets encompass a variety of widely used cathode active materials, capacities, and manufacturers. A summary of the dataset statistics is presented in Table 2.

Table 2: Main specifications of selected LIB datasets.

| Dataset | Electrode Material | Nominal Capacity | Voltage Range | Samples | Estimated Collect Time | Collector |
|-----------|--------------------|------------------|---------------|---------|------------------------|---------------------------------|
| CALCE | LCO | 1.1 (Ah) | 2.7-4.2 (V) | 2807 | 1397 (hour) | University of Maryland |
| SANYO | NMC | 1.85 (Ah) | 3.0-4.1 (V) | 415 | 644 (hour) | RWTH Aachen University |
| KOKAM | LCO/NCO | 0.74 (Ah) | 2.7-4.2 (V) | 503 | 8473 (hour) | University of Oxford |
| PANASONIC | NCA | 3.03 (Ah) | 2.5-4.29 (V) | 2770 | 1801 (hour) | Beijing Institute of Technology |
| GOTION | LFP | 27 (Ah) | 2.0-3.65 (V) | 4262 | 2238 (hour) | Beijing Institute of Technology |
| NHRY | LFP | 300 (Ah) | 2.5-3.5 (V) | 808 | 1200 (hour) | Ours |

5.1.2 Baselines. We compare our method with four types of baselines to demonstrate the efficacy of the proposed BatteryTTT framework and the GPT4Battery model: (1) Existing non-transfer learning machine learning (ML) methods for LIB State of Health (SOH) estimation, including Gaussian Process Regression [47], Random Forest [12], Multi-Layer Perceptron (MLP) [14], Recurrent Neural Networks (RNNs) [16], and Transformer [44]³; (2) Integration of existing models into our BatteryTTT framework; (3) State-of-the-art transfer learning methods for cross-domain LIB SOH estimation, such as woDegrad [22] and SSF-WL [46]⁴; and (4) Integration of large language models (LLMs) into the BatteryTTT framework (GPT4Battery).

We reproduce the results of BatteryML and woDegrad based on the provided code and follow the approach details for SSF-WL. For a fair comparison, we adhere to the data pre-processing methods outlined in [51] and use *QdLinear* [1] as the unified feature set. We adopt the standard evaluation metrics of mean absolute error (MAE) and root mean squared error (RMSE).

5.1.3 Implementation details. Our models are implemented using Pytorch and trained on a single 3070Ti GPU. We utilize the AdamW optimizer [21] with a fixed learning rate of 1e-3 for pre-training and linear probing until convergence. The mask ratio is set to 30% during this phase. TTA is conducted using stochastic gradient descent (SGD) with a momentum of 0.9 due to its consistency in improving performance on distribution shifts [10]. Typically, we set a fixed learning rate of 1e-2 and iterate for 10 steps, as more steps only marginally improve performance based on our observations. The mask ratio during TTA will be specifically analyzed later. For further details, we have made the relevant code and data available at the following link: <https://anonymous.4open.science/r/gpt4battery-55FC>.

5.2 Main Performance

In this section, we report the main improvements in cross-domain generalization performance of our proposed TTA methods (e.g. PG-SSL and PPA), on five commercial LIB datasets for daily usage. Additionally, we demonstrate that by leveraging an LLM as the backbone, GPT4Battery achieves superior generalization performance compared to all baseline methods.

5.2.1 Improvement of Generalization Ability of TTA. Table 3 shows the main improvements in the generalization performance of our proposed TTA methods. We use the GOTION dataset as the source

Table 3: Improved performance of TTA on existing methods and comparison with current transfer learning methods.

| Models | CALCE | | SANYO | | KOKAM | | PANASONIC | | GOTION | |
|---------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| Random Forest | 5.76 | 4.9 | 7.87 | 6.73 | 5.76 | 4.88 | 4.96 | 4 | 0.62 | 0.54 |
| Light GBM | 6.8 | 5.42 | 6.31 | 5.62 | 6.52 | 5.31 | 6.06 | 4.97 | 0.55 | 0.47 |
| MLP | 3.93 | 3.52 | 6.1 | 5.93 | 13.1 | 10.9 | 5.08 | 4.38 | 0.54 | 0.44 |
| GRU | 2.18 | 2.86 | 9.17 | 9.19 | 3.07 | 3.44 | 2.53 | 3.5 | 0.74 | 0.84 |
| LSTM | 2.52 | 2.78 | 7.58 | 7.84 | 3.07 | 3.93 | 1.55 | 2.19 | 1.65 | 1.68 |
| Transformer | 2.27 | 2.57 | 8.1 | 8.24 | 15.3 | 17.1 | 1.9 | 2.35 | 1.28 | 1.4 |
| GRU+ (TTA) | 1.9 | 2.25 | 7.8 | 8.01 | 2.4 | 2.74 | 1.74 | 2.73 | 0.67 | 0.71 |
| LSTM+ | 2.08 | 2.36 | 6.71 | 7.04 | 2.93 | 3.67 | 1.31 | 2 | 0.65 | 0.77 |
| Transformer + | 1.83 | 1.97 | 7.03 | 7.18 | 13.2 | 14.6 | 1.32 | 1.85 | 0.34 | 0.43 |
| GPT-2+ | 1.52 | 1.89 | 1.35 | 1.71 | 7.95 | 8.01 | 1.28 | 1.95 | 0.38 | 0.47 |
| Bert+ | 2.01 | 2.33 | 1.46 | 1.82 | 7.77 | 8.04 | 1.52 | 2.2 | 0.41 | 0.52 |
| Llama-7b+ | 1.57 | 1.94 | 1.35 | 1.69 | 8.58 | 9.66 | 1.3 | 2.01 | 0.4 | 0.49 |
| woDegrad | 1.76 | 1.96 | 1.21 | 1.54 | 1.76 | 3.01 | 2.09 | 2.56 | 0.45 | 0.58 |
| SSF-WL | 1.55 | 1.93 | 1.08 | 1.24 | 6.21 | 5.1 | 1.44 | 2.06 | 0.51 | 0.72 |

dataset for its extensive label coverage and then include its own test set along with the remaining four datasets (CALCE, SANYO, KOKAM and PANASONIC) as the target datasets for generalizability testing. Overall, we observe that the TTA method shows a significant performance improvement of about 50% compared to the no-TTA method. Some models (such as Transformer, GPT2) equipped with TTA achieve a performance that rivals or even exceeds the performance of current transfer learning methods that require additional access to the target data in CALCE, PANASONIC and GOTION. Specifically, within all the methods that use TTA, the utilization of large language models also has made quite a difference in generalization performance improvement. For instance, GPT2+ achieves first or second rank performance on the CALCE, GOTION and PANASONIC datasets compared to all baseline methods. Llama+ also obtained the best and second-best performance on the SANYO and PANASONIC dataset, respectively, compared to the other TTA methods. We also observe that some model architectures dominate the performance on certain datasets over TTA, e.g., the RNN family (GRU, LSTM) outperforms the Transformer family on KOKAM as a whole, and the TTA-enhanced GRU+ achieves performance comparable to that of woDegrad.

5.2.2 Superior Performance of GPT4Battery over all Baseline Methods. In this section, we provide a detailed comparison of the generalizability gains and computation trade-offs achieved by incorporating LLMs. Figure 4 demonstrates that GPT2 equipped with TTA (which we name GPT4Battery) obtains the lowest MAE results on the CALCE dataset than all other models including current transfer learning methods (woDegrad and SSF-WL) that require additional assumptions to access the target dataset. GPT4battery also gets

³BatteryML [51] provides a comprehensive platform summarizing these models

⁴These methods rely on sufficient unlabeled target data (UTDs) to operate effectively, which requires impractical data collection time, as summarized in Table 1.

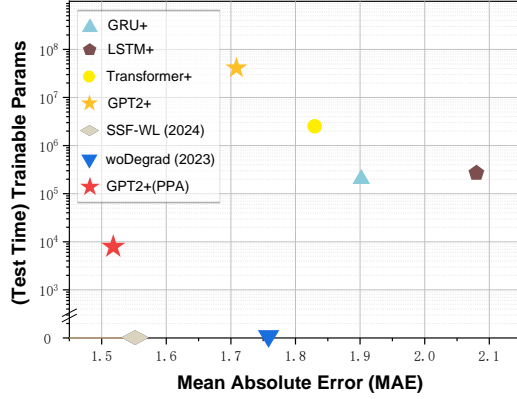


Figure 4: Superior Performance of GPT4Battery (on CALCE dataset).

very competitive results in other datasets as do other large models (e.g. Llama and Bert).

The results of applying Prefix Prompt Adaptation (PPA) to large language models (LLMs), as illustrated in Figure 4, are particularly noteworthy. Conventional fine-tuning of GPT-2’s positional encoding and layer normalization significantly increased the number of trained parameters at test time, by a factor of 10 to 100 compared to Transformer+ and RNNs+, while only (relatively) slightly reducing the mean absolute error (MAE) from 1.83 to 1.71. In contrast, our Prefix Prompt Adaptation (PPA) approach not only reduces the number of adjustable parameters by nearly 10,000 times, making GPT4Battery ten times more parameter-efficient than the RNN series, but also further reduces the MAE from 1.71 to 1.52, an improvement of approximately 10.7%. However, we should note the limitation that involving an LLM does increase the inference time even with PPA, which is a trade-off between accuracy and computation efficiency.

It is important to note that while applying PPA to regular Transformer+ and RNNs+ also reduces the number of trainable parameters to the order of 10^3 , it can impair their generalization performance, resulting in a slightly inferior performance compared to full-parameter fine-tuning. Therefore, we overall report the best performance for their full parameter fine-tuning, while reporting the best performance for LLMs using PPA in Table 3 and Figure 4. The performance of other models using PPA is thoroughly ablated in Sec. 5.3.1.

5.3 Efficacy Analysis

In this section, we analyze the effectiveness and important design choices of each component. Specifically, we evaluate the two TTA designs: Physical-Guided Self-supervised Learning (PG-SSL) and Prefix Prompt Adaptation (PPA).

5.3.1 Effect and Computation Trade-offs of Prefix Prompt Adaptation. By adjusting only a small number of soft prompts prefix to the input context, this design can significantly reduce the adjustable parameters to 10^3 orders of magnitude regardless of model size, as illustrated in Figure 5 on the GPT2, Transformer and LSTM models. Applying PPA to Transformer and LSTM slightly impact the

accuracy, as shown in Figure 5. Applying PPA to LLMs however, significantly reduces the Mean Absolute Error (MAE) from 3.89 to 1.52, a reduction of approximately 60.9%. Moreover, PPA achieves better generalization performance than traditional parameter-efficient fine-tuning of the positional encoding and layer normalization layers [54] of the LLM. We attribute this to the language-agnostic pattern recognition and inference capabilities acquired through pre-training on text corpora [13, 24]. Guided by learnable prefix prompts, these capabilities can also be generalized to battery sequence data.

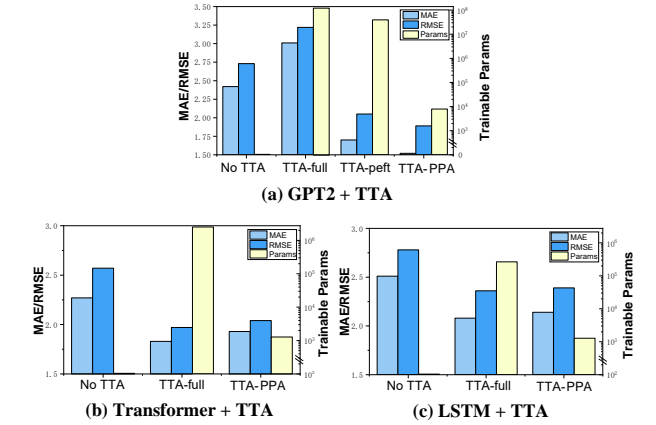


Figure 5: Efficacy and computation trade-offs of Prefix Prompt Adaptation (PPA) on different models.

5.3.2 Efficacy and Parameter Sensitivity of Physical-Guided Self-supervised Learning. In this section, we analyze the effectiveness of PG-SSL and conduct a sensitivity analysis of another important parameter affecting the self-supervised loss, the mask ratio. We performed ablation studies on losses using physical constraints and pure mask reconstruction. Additionally, we analyzed the effect of different masking rates on the MAE reduction of TTAs. We employ the transformer model and full-parameter tuning for TTA, conducting experiments on the adaptation to the CALCE and PANASONIC datasets.

Figure 6 shows that the inclusion of the physical-guided loss seamlessly enhances the performance in both MAE and RMSE reduction. Comparatively, the MAE reduction of using PG-SSL is slightly more significant in the CALCE data set (Fig. 6 (a)) than in the PANASONIC data set (Figure 6 (b)). This is because the former is a much more nonlinear dataset, making pure reconstruction loss less effective in capturing representations. We also observe from both (a) and (b) of Figure 6 that a larger mask ratio of 0.7 to 0.9 promotes learning a better representation with or without PG-SSL, while a small mask ratio of 0.5 to 0.6 fails. This is consistent with the MAE-based self-supervised learning observations in the computer vision domain, where masking a high proportion of the input image, e.g., 75%, yields a non-trivial and meaningful self-supervisory task. In summary, the combination of masking input and PG-SSL results in the best TTA results.

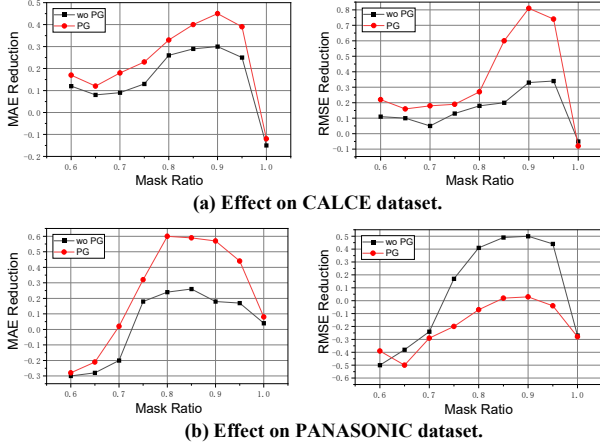


Figure 6: Effect on MAE/RMSE Reduction of Physical-Guided Self-supervised Learning and Sensitivity of Mask Ratio.

5.4 Ablation Study of GPT4Battery

In this section, we consider the best performing GPT2+TTA as a self-contained model (GPT4Battery) and provide ablation study on the effects of different components or design choices. Our results in Table 4 indicate that ablating either model reprogramming or any other designs in test-time adaptation hurts the generalization performance on unseen LIBs. In the absence of physical guidance, we observe a notable average performance degradation of **55.1%**, which becomes more pronounced (i.e., exceeding **70%**) when discarding the TTA strategy completely. The act of model reprogramming also stands as a pivotal element in cross-modality alignment, enabling the LLM to understand the LIB’s sequence data with the help of text prototypes. Ablation of reprogramming results leads to over **20%** degradation on average performance.

Table 4: Ablating the different components of GPT4Battery. Red: the best.

| Method | CALCE | SANYO | KOKAM | PANAS. | GOTION |
|-------------------------|-------------|-------------|-------------|-------------|-------------|
| GPT4Battery | 1.52 | 1.35 | 7.95 | 1.28 | 0.25 |
| w/o PG-SSL | 2.11 | 1.12 | 8.34 | 3.11 | 0.265 |
| w/o Masked TTA | 2.05 | 0.97 | 8.44 | 1.22 | 0.255 |
| w/o TTA | 2.13 | 1.34 | 9.51 | 3.44 | 0.297 |
| w/o model reprogramming | 4.13 | 2.78 | 10.56 | 4.57 | 0.31 |

5.5 TTA Efficiency

We present the average running time across five datasets for three representative models equipped with TTA and two current transfer learning methods. Our focus is primarily on model inference time, as TTA does not significantly impact training duration. Table 5 demonstrates that although TTA-equipped methods achieve substantially higher accuracy, they are 10 to 100 times slower than existing methods. This trade-off highlights that our approach sacrifices some speed for enhanced accuracy. However, the resulting 10^2 speed level remains sufficient for modern BMS requirements, where

individual response times typically need to be under 500ms in our deployed system. More importantly, our framework can eliminate the need for 644 to 8,473 hours of degradation experiments for data collection, as summarized in Tables 1 and 2.

Table 5: Efficiency Analysis of TTA.

| Method | GPT2+ | Transformer+ | LSTM+ | woDegrad | SSF-WL |
|----------------------------|-------|--------------|-------|----------|---------|
| Overall Inference Time (s) | 32.6 | 6.1 | 4.3 | 0.35 | 0.3 |
| One Inference Time (ms) | 51.17 | 9.57 | 6.75 | 0.55 | 0.47 |
| Model parameters | 7680 | 1280 | 1280 | 68774000 | 2601369 |

5.6 Scalability on Large Energy Storage LIBs

In this section, we report the scalability of the proposed framework through deployment on our battery management system (BMS) for industry-level energy storage LIBs (300Ah).

5.6.1 LIB Dataset. We collect the NHRY dataset by performing degradation experiments on 8 industry-level energy storage LIBs (300Ah) from 4 brands (Ningde, Haichen, Ruipu, Yiwei). Test environment and procedures comply with the China Standard BMS for Energy Storage GB/T 34131-2023. A total of 800 degradation cycles were experienced ranging over two months from December 2022 to January 2023.

5.6.2 Online Performance. For four different brands of LIBs with the same capacity, we mix two of the brands as the source domain and evaluate the online performance of the LIBs from the remaining two brands, resulting in a total of two combinations for the cross-battery setting. We report the MAE/RMSE metric along with the inference time (ms) using representative baselines.

Table 6: Online performance on Large Energy Storage LIBs.

| | Combination 1 | | | Combination 2 | | |
|--------------|---------------|-------|-----------------|---------------|-------|-----------------|
| | MAE | RMSE | Per Infer. Time | MAE | RMSE | Per Infer. Time |
| LSTM | 4.33 | 4.59 | 4.23 | 2.44 | 2.69 | 3.23 |
| Transformer | 5.62 | 5.89 | 7.61 | 3.48 | 3.78 | 6.32 |
| LSTM+ | 0.775 | 0.788 | 54.53 | 0.55 | 0.62 | 44.85 |
| Transformer+ | 1.25 | 1.38 | 88.57 | 1.02 | 1.23 | 77.83 |
| GPT4Battery | 0.734 | 0.761 | 103.47 | 0.314 | 0.418 | 105.68 |

6 Conclusion

In this paper, we propose a novel test-time adaptation (TTA) framework for cross-domain LIB state of health (SOH) estimation. This one sample adaptation setting allows the model to continually adapt to the target domain with every single unlabeled test sample, perfectly aligning with the nature of battery degradation features, which can only be obtained one by one during the long aging process. This setting also addresses the limitations of existing transfer learning methods, which assume additional access to the target LIB dataset, thereby saving months to years of labor in data collection. By introducing GPT4Battery, we hope this work will inspire the research community to fully leverage advanced AI techniques, such as Test-Time Adaptation (TTA) and large language models (LLMs), for more AI4Science problems, thereby saving enormous time and accelerating scientific discovery.

References

- [1] Peter M Attia, Kristen A Severson, and Jeremy D Witmer. 2021. Statistical learning for accurate and interpretable battery lifetime prediction. *Journal of The Electrochemical Society* 168, 9 (2021), 090547.
- [2] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. 2022. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274* (2022).
- [3] Christoph Birkel. 2017. Oxford battery degradation dataset 1. (2017).
- [4] Leo Breiman. 2001. Random forests. *Machine learning* 45 (2001), 5–32.
- [5] Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. 2023. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. *arXiv preprint arXiv:2310.04948* (2023).
- [6] Ching Chang, Wei-Yao Wang, Wen-Chih Peng, and Tien-Fu Chen. 2024. LLM4TS: Aligning Pre-Trained LLMs as Data-Efficient Time-Series Forecasters. *arXiv:2308.08469* [cs.LG]
- [7] Cheng Chen, Rui Xiong, Ruixin Yang, and Hailong Li. 2022. A novel data-driven method for mining battery open-circuit voltage characterization. *Green Energy and Intelligent Transportation* 1, 1 (2022), 100001.
- [8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
- [9] Noelia Ferruz, Steffen Schmidt, and Birte Höcker. 2022. ProtGPT2 is a deep unsupervised language model for protein design. *Nature communications* 13, 1 (2022), 4348.
- [10] Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei Efros. 2022. Test-time training with masked autoencoders. *Advances in Neural Information Processing Systems* 35 (2022), 29374–29385.
- [11] Konstantinos N. Genikomsakis, Nikolaos-Fivos Galatoulas, and Christos S. Ioakimidis. 2021. Towards the development of a hotel-based e-bike rental service: Results from a stated preference survey and techno-economic analysis. *Energy* 215 (2021), 119052. <https://doi.org/10.1016/j.energy.2020.119052>
- [12] Pierre Geurts, Damien Ernst, and Louis Wehenkel. 2006. Extremely randomized trees. *Machine learning* 63 (2006), 3–42.
- [13] Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. 2024. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems* 36 (2024).
- [14] Simon Haykin. 1998. *Neural networks: a comprehensive foundation*. Prentice Hall PTR.
- [15] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 16000–16009.
- [16] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
- [17] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual prompt tuning. In *European Conference on Computer Vision*. Springer, 709–727.
- [18] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. 2023. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728* (2023).
- [19] Ming Jin, Yifan Zhang, Wei Chen, Kexin Zhang, Yuxuan Liang, Bin Yang, Jindong Wang, Shirui Pan, and Qingsong Wen. 2024. Position Paper: What Can Large Language Models Tell Us about Time Series Analysis. *arXiv preprint arXiv:2402.02713* (2024).
- [20] Jian Liang, Ran He, and Tieniu Tan. 2024. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision* (2024), 1–34.
- [21] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).
- [22] Jiahuan Lu, Rui Xiong, Jinpeng Tian, Chenxu Wang, and Fengchun Sun. 2023. Deep learning to estimate lithium-ion battery state of health without additional degradation experiments. *Nature Communications* 14, 1 (2023), 2760.
- [23] Liuzhenghao Lv, Zongying Lin, Hao Li, Yuyang Liu, Jiayi Cui, Calvin Yu-Chian Chen, Li Yuan, and Yonghong Tian. 2024. Prollama: A protein large language model for multi-task protein language processing. *arXiv preprint arXiv:2402.16445* (2024).
- [24] Suvir Mirchandani, Fei Xia, Pete Florence, Brian Ichter, Danny Driess, Montserrat Gonzalez Arenas, Kanishka Rao, Dorsa Sadigh, and Andy Zeng. 2023. Large Language Models as General Pattern Machines. *arXiv:2307.04721* [cs.AI]
- [25] Man-Fai Ng, Jin Zhao, Qingyu Yan, Gareth J Conduit, and Zhi Wei Seh. 2020. Predicting the state of charge and health of batteries using data-driven machine learning. *Nature Machine Intelligence* 2, 3 (2020), 161–170.
- [26] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofu Chen, Shijian Zheng, Peilin Zhao, and Minghui Tan. 2022. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*. PMLR, 16888–16905.
- [27] Zijie Pan, Yushan Jiang, Sahil Garg, Anderson Schneider, Yuriy Nevmyvaka, and Dongjin Song. 2024. IP-LLM: Semantic Space Informed Prompt Learning with LLM for Time Series Forecasting. *arXiv preprint arXiv:2403.05798* (2024).
- [28] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training. (2018).
- [29] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.
- [30] Carl Edward Rasmussen and Hannes Nickisch. 2010. Gaussian processes for machine learning (GPML) toolbox. *The Journal of Machine Learning Research* 11 (2010), 3011–3015.
- [31] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. 2021. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences* 118, 15 (2021), e2016239118. <https://doi.org/10.1073/pnas.2016239118> <https://www.pnas.org/doi/pdf/10.1073/pnas.2016239118>
- [32] Darius Roman, Saurabh Saxena, Valentin Robu, Michael Pecht, and David Flynn. 2021. Machine learning pipeline for battery state-of-health estimation. *Nature Machine Intelligence* 3, 5 (2021), 447–456.
- [33] Kristen A Severson, Peter M Attia, Norman Jin, Nicholas Perkins, Benben Jiang, Zi Yang, Michael H Chen, Muratahan Aykol, Patrick K Herring, Dimitrios Fraggedakis, et al. 2019. Data-driven prediction of battery cycle life before capacity degradation. *Nature Energy* 4, 5 (2019), 383–391.
- [34] Manli Shu, Wei Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. 2022. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems* 35 (2022), 14274–14289.
- [35] Xing Shu, Jiangwei Shen, Guang Li, Yuanjian Zhang, Zheng Chen, and Yonggang Liu. 2021. A Flexible State-of-Health Prediction Scheme for Lithium-Ion Battery Packs With Long Short-Term Memory Network and Transfer Learning. *IEEE Transactions on Transportation Electrification* 7, 4 (2021), 2238–2248. <https://doi.org/10.1109/TTE.2021.3074638>
- [36] Dimitris Spathis and Fahim Kawsar. 2023. The first step is the hardest: Pitfalls of Representing and Tokenizing Temporal Data for Large Language Models. *arXiv:2309.06236* [cs.LG]
- [37] Chenxi Sun, Yaliang Li, Hongyan Li, and Shenda Hong. 2023. TEST: Text prototype aligned embedding to activate LLM’s ability for time series. *arXiv preprint arXiv:2308.08241* (2023).
- [38] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. 2020. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*. PMLR, 9229–9248.
- [39] Yandan Tan and Guangcai Zhao. 2020. Transfer Learning With Long Short-Term Memory Network for State-of-Health Prediction of Lithium-Ion Batteries. *IEEE Transactions on Industrial Electronics* 67, 10 (2020), 8723–8731. <https://doi.org/10.1109/TIE.2019.2946551>
- [40] Jinpeng Tian, Rui Xiong, Weixiang Shen, Jiahuan Lu, and Xiao-Guang Yang. 2021. Deep neural network battery charging curve prediction using 30 points collected in 10 min. *Joule* 5, 6 (2021), 1521–1534.
- [41] Jinpeng Tian, Rui Xiong, Weixiang Shen, Jiahuan Lu, and Xiao-Guang Yang. 2021. Deep neural network battery charging curve prediction using 30 points collected in 10 min. *Joule* 5, 6 (2021), 1521–1534. <https://doi.org/10.1016/j.joule.2021.05.012>
- [42] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
- [43] Manh-Kien Tran, Manoj Mathew, Stefan Janhunen, Satyam Panchal, Kaamran Raahemifar, Roydon Fraser, and Michael Fowler. 2021. A comprehensive equivalent circuit model for lithium-ion batteries, incorporating the effects of state of health, state of charge, and temperature on model parameters. *Journal of Energy Storage* 43 (2021), 103252.
- [44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
- [45] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2020. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726* (2020).
- [46] Tianyu Wang, Zhongjing Ma, Suli Zou, Zhan Chen, and Peng Wang. 2024. Lithium-ion battery state-of-health estimation: A self-supervised framework incorporating weak labels. *Applied Energy* 355 (2024), 122332.
- [47] Christopher KI Williams and Carl Edward Rasmussen. 2006. *Gaussian processes for machine learning*. Vol. 2. MIT press Cambridge, MA.
- [48] Rui Xiong, Linlin Li, and Jinpeng Tian. 2018. Towards a smarter battery management system: A critical review on battery state of health monitoring methods. *Journal of Power Sources* 405 (2018), 18–29.
- [49] Zhaoyi Xu, Yanjie Guo, and Joseph Homer Saleh. 2022. A physics-informed dynamic deep autoencoder for accurate state-of-health prediction of lithium-ion battery. *Neural Computing and Applications* 34, 18 (2022), 15997–16017.

- [50] Sijia Yang, Caiping Zhang, Jiuchun Jiang, Weige Zhang, Yang Gao, and Linjing Zhang. 2021. A voltage reconstruction model based on partial charging curve for state-of-health estimation of lithium-ion batteries. *Journal of Energy Storage* 35 (2021), 102271.
- [51] Han Zhang, Xiaofan Gui, Shun Zheng, Ziheng Lu, Yuqi Li, and Jiang Bian. 2024. BatteryML: An Open-source Platform for Machine Learning on Battery Degradation. In *The Twelfth International Conference on Learning Representations*.
- [52] Marvin Zhang, Sergey Levine, and Chelsea Finn. 2022. Memo: Test time robustness via adaptation and augmentation. *Advances in neural information processing systems* 35 (2022), 38629–38642.
- [53] Aurick Zhou and Sergey Levine. 2021. Bayesian adaptation for covariate shift. *Advances in neural information processing systems* 34 (2021), 914–927.
- [54] Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. 2023. One Fits All: Power General Time Series Analysis by Pretrained LM. *arXiv preprint arXiv:2302.11939* (2023).

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009