# GPT4Battery: Cross-battery State of Health Estimation via Physics Guided Test-time Prompt Learning with LLM

Anonymous Authors


## ABSTRACT

State of health (SOH) is a crucial indicator to evaluate the level of degradation of batteries that cannot be measured directly but requires estimation. Accurate SOH estimation enhances detection, control, and feedback for Li-ion batteries, allowing for safe and efficient energy management and guiding the development of new-generation batteries. Despite the significant progress in data-driven SOH estimation methods, the time- and resource-consuming degradation experiments to generate lifelong training data pose a barrier to timely safety monitoring and the development of new battery technologies. To tackle this problem, We propose GPT4Battery, the first foundation model for battery to utilize the strong generalization ability of the large language model (LLM) for cross-battery SOH estimation. We design a translator to reprogram the voltage-time battery charging data into text prototype representations to align data from different modalities, adapting the LLM to battery tasks. We also design a novel *physical guided test-time prompt tuning (PGTPT)* method to learn adaptive prompts on the fly with a single test sample, enhancing the model's generalization ability and fitting the real-world scenarios. PGTPT optimizes the prompt by guiding the LLM to generate a complete battery charging curve in accordance with the physics equations of the 1-RC ECM model of a LIB cell. The validation results demonstrate that the proposed model achieves state-of-the-art accuracy that is even on par with domain adaptation or fine-tuning methods that require additional training data on five widely recognized datasets collected from 65 batteries.



## CCS CONCEPTS

• **Large Language Models** → **Battery Health Estimation**.

## KEYWORDS

Battery health monitoring, large language model, prompt learning, test-time training


## 1 INTRODUCTION

The rapid advancements in rechargeable Li-ion batteries (LIBs) have led to their widespread adoption in various applications, ranging from portable electronics and medical devices to renewable energy integration and electric vehicles [13], and reaching a value of more than $400 billion and a market size of 4.7 TWh [18].

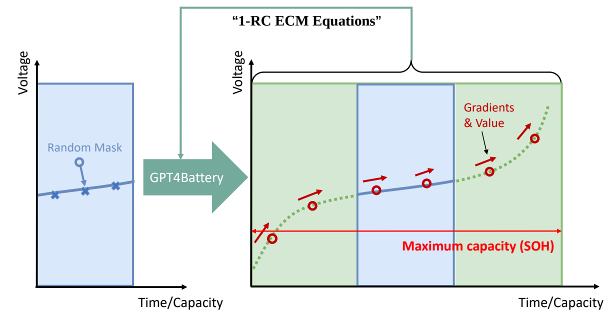



*ACM MM, 2024, Melbourne, Australia*





ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn


**(a) LIB feature charging curve** **(b) Generated complete curve by LLM**

**Figure 1: Guide the LLM to generate a complete battery charging curve in accordance with the physics equations of the 1-RC ECM model[1] of a LIB cell.**

A crucial aspect of LIB management is the State of Health (SOH), which cannot be measured directly but estimated to evaluate the degradation level of the batteries [27]. Accurate SOH estimation is essential for safe and efficient energy management and development of new-generation batteries [39].

Existing data-driven SOH estimation methods require lifelong data with precise SOH labels to establish the mapping relationship for every LIB [38]. Due to the extremely long life cycle of modern LIBs, obtaining the lifelong training data necessitates labor-intensive degradation experiments that often require many months to years (see the estimated collection time shown in Table 2), which greatly hinders their deployment to battery management systems (BMS) and the development of new battery technologies [32, 38]. Emerging transfer learning-based methods for SOH estimation [27, 41, 45, 49] relieves the burden of data collection to some extent; however, their assumption about the target LIB dataset is far from perfect (as shown in Table 1). For example, [45] requires a 25% labeled target LIB dataset and [27] demands lifelong unlabeled features of the target LIB. Under these settings, quite a bit of effort is still needed to acquire data of the target LIB dataset.

Instead, we assume that we only access *one* sample of the target LIB at a time, which is a more realistic scenario. For instance, by charging a new cell phone once daily, BMS can only collect a *single* unlabeled charging feature curve. It is of great value if we can leverage the health information[2] present within the existing sample itself (e.g., accessing only one sample from a new domain) to update the model, rather than spending a significant amount of

[1]Here the 1-RC ECM model is a widely used physical model to describe the electrical behavior of the battery, we will explain in Section 2 and Section 3.3.2 later.

[2]Here health information means the features presented in the LIB charging feature curve like Fig. 1a.

time collecting numerous samples for model updates. To the best of our knowledge, we are the first to tackle this problem.

On the other hand, in addition to the great success in language modeling [6, 50], foundation models have also shown significant progress in several other domains as computer vision [24, 26], time series [19, 59], audio [22] and graphs [46]. Foundation models are generally defined as models trained on broad data that can be adapted to various downstream tasks [4]. For general time series data that battery charging data belong, there are two typical paradigms for creating a foundation model: collecting adequate data and train a large model from scratch [12][11] or exploring the feasibility of pre-trained large language model (LLM), i.e., to reprogram the input type and/or fine-tune the LLM [59][7][19]. Among these works, GPT4TS [59] leveraged a modified GPT2 for general time series analysis and achieved results comparable to SOTA. While the effectiveness of LLMs in handling time series data has been demonstrated, its benefits have not yet been extended to the realm of battery research. Furthermore, as we will illustrate, adapting LLMs for SOH estimation is not a straightforward task.

As the time required for dataset collection cannot keep pace with the development of new batteries, to collect extensive datasets and train a large LIB model is seldom sustainable. Therefore we leverage the strong generalization ability of LLMs and propose GPT4Battery, the first foundation model for cross-battery SOH estimation. We design a translator layer to reprogram the voltage-time battery charging data into text prototype representations that are more naturally suited to language models' capabilities, aligning data from different modalities and adapting the LLM to battery tasks. To further augment the model's zero-shot generalization ability, we propose a novel *physical guided test-time prompt tuning (PGTPT)* strategy to learn adaptive prompts on the fly by fully exploring the health information existing in the *single* test sample. PGTPT optimizes the prompt by guiding the LLM to generate a complete battery charging curve in accordance with the physics equations of the 1-RC ECM model of a LIB cell from a partial masked one at test time (e.g., Fig. 1b). The latter can theoretically identify the accurate SOH [48]. This strategy also effectively alleviates the cumulative errors resulting from temporal distribution shifts caused by variations in the mechanisms within the battery (such as side reactions and stability of solid electrolyte interface) [5]. Without the need for additional training data or annotations, GPT4Battery achieves state-of-the-art accuracy among extensive baseline methods for cross-battery SOH estimation. In summary, our main contributions are as follows:

- We propose GPT4Battery, the first foundation model for cross-battery SOH estimation, relieving the burden of months-to-years degradation experiments for data collection.
- We design a novel PGTPT strategy to learn adaptive prompts on the fly with a single test sample, enhancing the model's generalization ability and adaptability in real-world scenarios.
- We conduct extensive experiments on five widely recognized LIB datasets. GPT4Battery achieves state-of-the-art accuracy for cross-battery SOH estimation, on par with domain adaptation or fine-tuning methods that require additional training data.

By introducing GPT4Battery, we hope this work will inspire the research community to fully leverage the generalization potential of LLMs for diverse multimedia/multimodal applications, utilizing techniques such as model reprogramming and incorporating laws and domain-specific knowledge within their respective modal domains. For the rest of this paper, Sec. 2 and 5 present background and related works, Sec. 3 describes the framework of GPT4Battery, Sec. 4 conducts experiments, and Sec. 6 concludes.

## 2 BACKGROUND

In this section, we give a breif introduction to LIB state estimation, pointing out the shortages of present methods and clarifying the motivation of LLM-driven and physcal guided test-time learning.

### 2.1 Definition

SOH is a critical state which evaluates the degradation level of batteries [27]. Despite defined in various forms, *the ratio between the present capacity and the initial capacity* is widely thought to have high impact on battery management tasks such as driving range estimation and life prediction [2][38][32]. This work also focuses on this capacity definition as the health indicator:

$$SOH_{cycle} = \frac{Q_{cycle}}{Q_{nominal}} \tag{1}$$

where $Q_{nominal}$ is the nominal capacity of the brand new battery, and $Q_{cycle}$ is the fully charged capacity at the current cycle. However, the capacity measurement requires completely charging or discharging the batteries with specific protocols [55], which is not practical for batteries in use. This motivates the research of SOH estimation from daily operating data [48][27][38].

### 2.2 Modeling methods vs Data-driven methods

In the field of battery research, there are mainly two ways to estimate the battery state: modeling-based and data-driven [32]. Modeling-based methods involve building a simulation model equivalent to the LIB and using parameter estimation to recognize the real battery's state. Two most studied models are the equivalent circuit model (ECM) [51] which describes the battery via a resistor-capacitor circuit and the pseudo two-dimensional (P2D) model [10] which provides insights into the internal dynamics of the batteries such as lithium-ion diffusion and electrochemical kinetics. The good part is that these models provide detailed interpretability of how the LIB works. However, they have to make a trade-off between accuracy and computational efficiency for online applications since a detailed model leads to complex partial differential equations that have a high requirement of memory and computing power to solve [38] [32].

Conversely, the data-driven approach has both high accuracy and a very short inference time. However, two drawbacks still exist in data-driven methods, making up today's SOH estimation challenge: (1) generalization ability rely heavily on data and (2) lack of interpretability. Firstly, unlike general time series data, obtaining the battery data necessitates extremely expensive degradation experiments that often require many months to years [27], making the training data collection a time- and resource-consuming labor. Secondly, since it often spans over months to years in usage,

**Table 1: Compare the differences in the assumptions about the target dataset made by several methods.**

| Assumption\Method | Domain Adaptation | Fine-tuning | Zero-shot | Test-time training | **GPT4Battery** |
|---|---|---|---|---|---|
| Target Feature | ✓ | ✓ | × | one at a time | one at a time |
| Target Label | × | ✓ | × | × | × |

there are often variations in the mechanisms within the battery (such as side reactions and stability of solid electrolyte interface) [5]. Therefore, the battery data are more than likely to involve *temporal distribution shifts*, making a pure data-driven model fall in the mid-to-late life cycle.

## 2.3 From Data-driven to LLM-driven

To overcome the limitations of the pure data-driven approach, it is necessary to build a model that is highly generalizable and preferably interpretable.

Recently, a batch of work [14][59][19] has evaluated the excellent performance of LLM (pretrained on language) on zero-shot time series tasks, which inspired us to save the tedious data collection labor by leveraging LLM's generalization ability. For interpretability, thanks to the physics-informed machine learning (PIML) techniques, several architectures for integrating physics-based and machine learning have been studied [1, 17, 56]. These methods combine physical models such as Newman's pseudo-two-dimensional (P2D) [10] with basic neural networks as an unsupervised loss or more commonly, use physics-based models to generate training data for machine learning models. To achieve a better combination of the LLM's generalizability and physics-based models' interpretability, we aim to embed the knowledge and rules within physics-based models into the LLM in the form of prompts.

Meanwhile, we noticed that very few prior works have valued the health info inherent in the partial charging curve of the test sample. This sample reflects the latest and most real battery state. If properly guided by physical rules, this sample is a good supplement of data shortage.

To put it all together, we design a novel physics-guided test-time prompt tuning (PGTPT) algorithm to learn adaptive prompts with a single test sample. We optimize the prompt by guiding the LLM to generate a complete battery charging curve in accordance with the physical equations of Thevenin's 1-RC equivalent circuit model (ECM), enhancing the model's generalization ability and making up for the lack of pure data-driven model's interpretability.

# 3 GPT4BATTERY

Our model architecture is depicted in Fig. 2. We focus on fully exploring LLM's generalization potential for cross-battery SOH estimation via a translator to reprogram the input for modality alignment and test-time prompt learning guided by physics. The whole process does *not* require any fine-tuning of the backbone model. Our method encompasses four main components: (1) a input translator layer, (2) a pre-trained frozen LLM, (3) output projection, and (4) feature reconstruction. During training, all three tunable components are jointly trained along with the prompt. At test time, only the prompt is optimized by guiding the LLM to reconstruct a complete battery charging curve in accordance with the physics equations of the 1-RC ECM model of a LIB cell from a partial masked one.

## 3.1 Problem Statement

We first formalize the cross-battery SOH estimation problem. Firstly, given a well-collected battery dataset, let $\mathbf{X} \in \mathbb{R}^{1\times T}$ denote a partial constant current voltage-time curve with $T$ time steps and $y$ denote the SOH label. We have a source set of $S$-labeled lifelong samples $\mathcal{S} = \{(x_1, y_1), (x_2, y_2), ..., (x_S, y_S)\}$. For a new battery, we acquire the target feature sequentially through everyday charging and discharging, denoting $\mathcal{T} = \{(x_1, y_1), x_2, ..., x_T\}$. Our goal is to estimate every target $y_t$. Note that $y_1$ can be seen as 100% for a new battery.

## 3.2 Translate the Input to Text Prototype Representations

### 3.2.1 *Normalize, Patchify and Embedding.*

We consider the input partial voltage-time charging curve with a fixed voltage sampling window $\Delta V$ as an univariate time series data. Firstly, in a constant current charging process, the LIB charge is obtained by integrating the current at each time step[27]:

$$q_i(V) = \int_{V_{\min}}^{V_{\min}+i\Delta V} |I(t)|dt, i \in \{0, 1, \cdots, T\} \tag{2}$$

where $I(t)$ denotes the current, we expand it to a 1-d vector and normalize it with the LIB's nominal capacity $Q_{nominal}$ to make it adaptive to different LIBs:

$$\boldsymbol{q(V)} = [q_0(V), q_1(V), \dots q_T(V)]/Q_{nominal} \tag{3}$$

We denote this series data $\boldsymbol{q(V)}$ as $\mathbf{X} \in \mathbb{R}^{1\times T}$. Then we follow [33] to encode temporal information and local contexts of input series by aggregating consecutive timestamps into overlapped patched tokens with length $L_p$; thus the total number of input patches is $P = \lfloor \frac{(T-L_p)}{S} \rfloor + 2$, where $S$ denotes the horizontal sliding stride. Given these patches $\mathbf{X}_P \in \mathbb{R}^{P\times L_p}$. We then adopt a simple linear layer as the patch embedder to create dimensions $D$, obtaining the patch embeddings $\mathbf{X}_P \in \mathbb{R}^{P\times D}$, where $D$ is the dimension of word embedding for ease of translation later.

### 3.2.2 *Reprogram Patch Embeddings for modaliy alignment.*

Now we get the patched embedding form of the input sequences. To align the modality between time series data and language and activate the LLM's time series understanding capabilities, we reprogram the patch embeddings into LLM's pre-trained word embedding space.

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 [36]). Simply

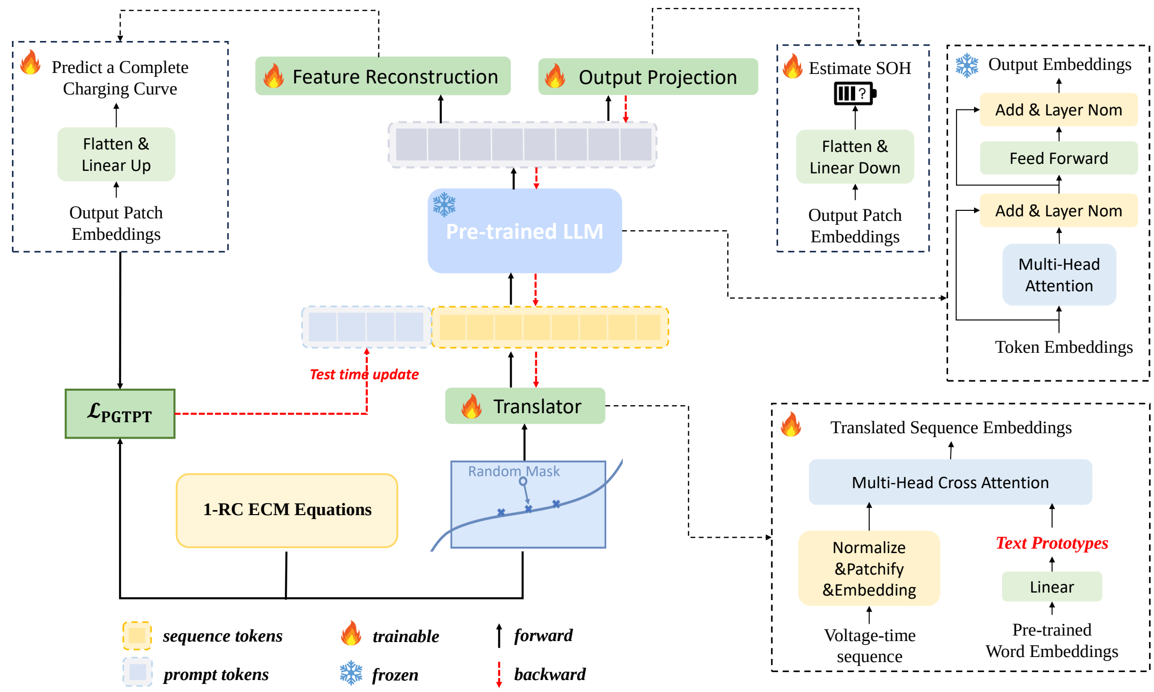



**Figure 2: The framework of GPT4Battery. Given an input voltage-time curve, we first pass it through a translator to reprogram the patch embeddings with condensed text prototypes to align two modalities. After pass the frozen LLM, the output patch embeddings go through an output projection layer to estimate the target SOH and a feature reconstruction layer to generate a complete voltage-time curve in accordance with the equations of the 1-RC model respectively.**

leveraging $\mathbf{E}$ will result in large and potentially dense reprogramming space, increasing the computation efficiency and difficulty of catching the relevant source tokens. Following [43] and [19], 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 time series patches and text prototypes with a multi-head cross-attention layer. This way, the text prototypes can learn cues in language which can then represent the relevant local patch information. Specifically, for each head $k = \{1, \cdots, K\}$, we define query matrices $\mathbf{Q}_k^{(i)} = \hat{\mathbf{X}}_P^{(i)} \mathbf{W}_k^Q$, key matrices $\mathbf{K}_k^{(i)} = \mathbf{E}' \mathbf{W}_k^K$ and value matrices $\mathbf{V}_k^{(i)} = \mathbf{E}' \mathbf{W}_k^V$, where $\mathbf{W}_k^Q, \mathbf{W}_k^K, \mathbf{W}_k^V \in \mathbb{R}^{D \times d}$. Specifically, $D$ is the dimension of the LLM's word token embedding and $d = \lfloor \frac{D}{K} \rfloor$. Then, we reprogram time series patches in each attention head defined as:

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

By aggregating each $\mathbf{Z}_k^{(i)} \in \mathbb{R}^{P \times d}$ in every head, we obtain $\mathbf{Z}^{(i)} \in \mathbb{R}^{P \times D}$. Now we have finished reprogramming the input battery data into sequence embeddings, effectively aligning the modality of time series data and natural language to leverage the LLM's capabilities.

## 3.3 Physics Guided Test-time Prompt Tuning with LLM

By reprogramming the input into text prototype representations and linear output projection, we guide the LLM to adapt to the downstream task of battery SOH estimation. However, as explained in Sec. 2 we can only get *one* test sample each time in the real-world battery usage scenarios, which is a process for years.

To meet the real-world setting and alleviate the accumulated errors brought by *temporal distribution shifts*, We design a novel physical guided test-time prompt tuning (PGTPT), which optimizes the prompt by guiding the LLM to generate a complete battery charging curve in accordance with LIB's 1-RC ECM model from a masked partial one.

*3.3.1 Optimize the Prompts at Test-time.* Different from directly fine-tuning the model, whether end-to-end or a subset of layers, prompts learning work outside the pre-trained model by only modifying the context of the model input. Therefore, this technique does not distort pre-trained features or result in domain-specific behaviors that lose LLM's out-of-distribution generalization [54] [21].

Test-time training[44] [40] is a strategy that was originally proposed to improve the generalization ability under distribution shifts by fully utilizing the feature of every test sample to update the model. In recent years, there has been work on combining prompt learning and test-time training to boost the generalization ability of large pre-trained models for out-of-distribution or zero-shot tasks. For instance,[40] made efforts to enhance CLIP's[35] zero-shot performance on visual tasks by tuning the aligned prompts during test time. Therefore, We also expect to boost LLM's generalization ability on our cross-battery task by optimizing the prompt at test-time, that is:

$$\boldsymbol{p}^* = \arg\min_{\boldsymbol{p}} \mathcal{L}(\mathcal{F}, \boldsymbol{p}, X_{\text{test}}) \tag{4}$$

with some carefully constructed unsupervised loss $\mathcal{L}$. $\mathcal{F}$ denotes the model and note that no extra labels or data is required beyond the zero-shot test sample. Everything is ready except for an unsupervised objective. We introduce the ECM model for now.

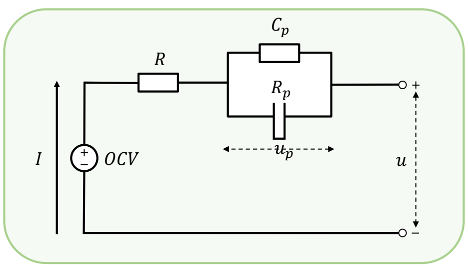



**Figure 3: 1-RC ECM Model for LIB**

*3.3.2 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 [55] [51]. 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 [56]. 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:

$$u_{OCV} = u_R + u_p + u \tag{5}$$

$$\frac{R + R_p}{C_p R_p} I + \frac{1}{C_p R_p} u + \dot{u} = 0 \tag{6}$$

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 coeffients are functions of temperature. The state function become:

$$\theta_1(T)I + \theta_2(T)u + \dot{u} = 0 \tag{7}$$

which means the terminal voltage $u$, current $I$ and temperature should in principle follow the battery's physical state ODE functions Eql. 7. The ECM model has good qualities such easy to implement and relatively accurate characterization of cells, inspiring us to employ it as physical constraints.

*3.3.3 Generate a complete battery curve that conforms to the laws of physics.* Because labels are not available for test-time tuning, we must select an unsupervised loss for prompt tuning. In the filed of battery research, a complete charging/discharging 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 [57] [9] [48]. Inspired by this, we want to design the objective by guiding the LLM to generate a complete battery charging curve from one masked partial charging curve, which can serve the Test-Time Training (TTT) for self-supervised learning. Specifically, given the input one test sample $\mathbf{X_{test}} \in \mathbb{R}^{1\times T}$, after reprogramming into token representations $\mathbf{Z} \in \mathbb{R}^{P\times D}$, we randomly mask a portion of tokens formalized by $\text{Mask}_r \mathbf{Z}^{(i)}$. We want to use the masked partial curve to generate a complete battery charging curve $\mathbf{X}' \in \mathbb{R}^{1\times T'}$, where $T' > T$. However, without any labeling of the complete curve, it is rarely possible to make meaningful generations. We can now use the laws of ECM modeling of LIB Eq.7 to guide the LLM for generation that in accordance with the laws of physics. In general, our objective can be formulated in the form of:

$$\mathcal{L}_{\text{PGTPT}} = \|\hat{\mathbf{X}}'[0:T] - \mathbf{X_{test}}\|_F^2 + \lambda\|\theta_1 I + \theta_2 \hat{u} + \dot{\hat{u}} = 0\|_F^2 \tag{8}$$

where $\hat{\mathbf{X}}'[0:T]$ denotes the generated complete voltage-time curve and known overlaps. $\hat{u}$ denotes the generated complete voltage-time curve, $\theta_1, \theta_2$ and $I$ correlates with the working condition collected by the BMS system (to see the Appendix for specifications). $\lambda$ is a hyper-parameter for weighting.

## 3.4 Training and Cross-battery Inference

In this paper, we employ GPT-2 [36] as our pre-trained large language model (LLM) backbone. During training on the source dataset, the reconstruction and output layers are trained simultaneously along with all the token embeddings. It should be noted that we updated the output linear layers using the first cycle's feature of target LIB (SOH = 100%) when transferring to the target LIB dataset. At test time, we further optimize the prompt tokens on the fly with a *single* test sample by guiding the LLM to reconstruct a complete battery charging curve conforming to the physics equations of the 1-RC ECM model of a LIB cell.

# 4 EXPERIMENTS

In this section, we describe the benchmarks used to evaluate GPT4Battery, along with the implementation details. The cross-dataset validation results demonstrate that our method achieves state-of-the-art accuracy for cross-battery SOH estimation that is even on par with domain adaptation or fine-tuning methods that require additional training data. We also provide ablation experiments to analyze different network components and other design choices of our method.

## 4.1 Experiment Settings

*4.1.1 Datasets.* We employ the following datasets from CALCE [16], SANYO [23], KOKAM [3], PANASONIC, and GOTION HIGH-TECH [27]. These datasets cover widely used cathode active materials, a capacity ranging from 0.74Ah to 27Ah, and five different manufacturers, emphasizing our method's capability for cross-battery

**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 Institude of Technology |
| GOTION | LFP | 27 (Ah) | 2.0-3.65 (V) | 4262 | 2238 (hour) | Beijing Institude of Technology |

generalization. The statistics of the five kinds of LIBs are compared in Table 2.

*4.1.2 Baselines.* We compare GPT4Battery with three categories of methods as below:

- Four popular data-driven SOH estimation methods include Gaussian process regression (GPR), support vector regression (SVR), Random Forest (RD) [38], and CNN [47] in the field of battery research.
- A series of transformer-based methods such as patchTST [33] and iTransformer [25] , along with a very recent LLM-driven work GPT4TS [59] for general time series data.
- Transfer learning methods for cross battery SOH estimation with the need of some target data for domain adaptation or fine-tuning [27].

*4.1.3 Implementation details.* We mainly follow the data processing and experimental evaluation in [27] for fair comparisons. For the PGTPT, we initialize the prompt as the default hand-crafted one "analyze the series data", and optimize the corresponding 4 tokens in the text input embedding space based on a single test sample. We optimize the prompt to minimize the PGTPT loss for 1 step, using the AdamW optimizer with a learning rate of 0.0001. After we obtain output representations $\tilde{O} \in \mathbb{R}^{P \times D}$ through the frozen LLM, we follow patchTST[33] and flatten $\tilde{O}$ into a 1D tensor with the length $P \times D$ For the output projection and feature reconstruction layers, which is then linear projected down as SOH $\hat{y}$ and linear projected up as a complete battery charging curve $\mathbf{X}' \in \mathbb{R}^{1 \times T'}$.

## 4.2 Cross-battery Generalization Results

In this section , we first compare GPT4Battery with all the baseline methods using GOTION as the source LIB dataset. Table 3 reports the results of the mean absolute error (MAE) and the inference time in three categories of baselines. In the absence of target LIB data, classical data-driven methods fail to provide reliable estimation with their MAEs **over 10%** except for SVR (5.31%). The transformer-based methods show a much better generalization performance, but their evaluated MAEs are still higher than 5%.

The comparison with GPT4TS[59] is particularly noteworthy. GPT4TS is a very typical and recent LLM-driven work for general time series that involves fine-tuning the backbone language model. In our experiment, GPT4TS significantly reduces the mean absolute errors (MAEs) compared to transformer-based methods when applied to unseen LIBs, achieving values of less than 5%. This result is consistent with its excellent cross-dataset generalization ability observed in generic time series datasets. We notice average performance gains of 57.5% over GPT4TS, which is quite on par with domain adaptation and fine-tuning methods that require additional training data.

Therefore, in Table 4 and Figure 4, we further compare GPT4Battery with GPT4TS by pairewise combining all five kinds of LIBs, resulting a total of 25 combinations (including the cases where the source and target sets are the same). Limited by space, we denote in Fig. 4 "PANASONIC" by "PANAS.", and report the average value in "AVE.". From the figure, we note that the evaluation metrics achieved evident reduction in all cross-dataset combinations, i.e., **< 4%** for most cases. Notably, GPT4Battery achieved an average reduction of **27.9%**[3]. The largest average reduction is 57.6% observed when GOTION is the source dataset, and the smallest decrease is 5.5% when SANYO is the source dataset [4].

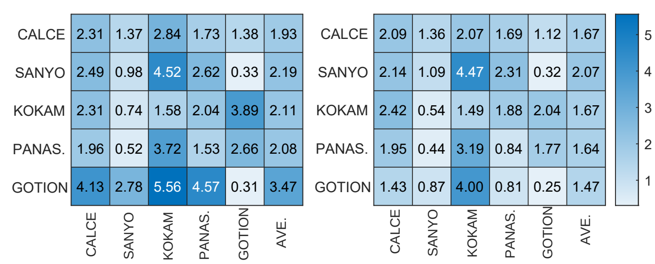



**Figure 4: Visualize of the improvement of GPT4Battery (right) over GPT4TS (left).**

## 4.3 Ablation Results

In this section, we descontruct GPT4Battery and provide ablation study on the effects of different components or design choices. For simplicity, evaluations in this section are conducted by taking GOTION as the source LIB data and the other five for cross-LIB evaluation (including the test set of GOTION). We first start from GPT4TS as the baseline and show the improvement brought by *input reprogramming*, *test-time training* and *physical guiding*, respectively. Next, we compare the design choices of PGTPT on different parameter groups of the LLM, showing that prompt tuning achieves the most accuracy gain.

*4.3.1 Ablating the different components of GPT4Battery.* Our results in Table 5 indicate that ablating either patch reprogramming or any other designs in PGTPT hurts the generalization performance

[3]This value is calculated by comparing the sum of the the MAEs of all pairwise combinations of LIB datasets (sum of 25 MAEs) of GPT4Battery over GPT4TS.
[4]57.6% and 5.5% is calculated by comparing the average MAE of GPT4Battery over GPT4TS when GOTION/SANYO is the source dataset.

**Table 3: Comparison of all baseline methods with GOTION as the source dataset. We calculate the MAE (as %) for each dataset and average four datasets as well as inference time (ms). A lower MAE score indicates better performance. Red: best, Blue:second best.**

| Datasets | CALCE | | SANYO | | KOKAM | | PANASONIC | | GOTION | | Average | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Methods/Metrics | MAE(%) | Time(ms) | MAE(%) | Time(ms) | MAE(%) | Time(ms) | MAE(%) | Time(ms) | MAE(%) | Time(ms) | MAE(%) | Time(ms) |
| GPR | 23.58 | 3.05 | 21 | 3.14 | 31.83 | 3.29 | 30.7 | 3.22 | 6.39 | 3.35 | 22.7 | 3.93 |
| RD | 8.74 | 3.92 | 4.32 | 4.43 | 9.02 | 4.15 | 10.52 | 3.63 | 0.29 | 4.02 | 8.38 | 2.84 |
| SVR | 4.27 | 3.86 | 5.62 | 3.02 | 6.44 | 3.69 | 5.53 | 3.22 | 4.66 | 4.05 | 5.31 | 3.66 |
| CNN | 10.31 | 4.71 | 17.9 | 4.49 | 14.64 | 4.66 | 25.46 | 3.81 | 0.28 | 3.05 | 13.72 | 3.55 |
| patchTST | 9.08 | 9.09 | 5.91 | 6.11 | 9.34 | 9.77 | 6.49 | 9.9 | 1.31 | 9.29 | 6.43 | 8.83 |
| iTransformer | 8.68 | 7.35 | 6.72 | 7.3 | 8.41 | 9.39 | 8.91 | 8.42 | 0.87 | 7.14 | 6.72 | 7.92 |
| GPT4TS | 4.13 | 4.86 | 2.78 | 4.25 | 5.56 | 4.14 | 4.57 | 5.5 | **0.31** | 4.56 | 3.47 | 4.66 |
| **GPT4Battery** | **1.43** | 43.73 | **0.87** | 68.72 | **4** | 255.24 | **0.81** | 61.17 | **0.25** | 47.98 | **1.47** | 95.37 |
| Domain Adaptation | **1.12** | 10.47 | 1.21 | 11.91 | **1.76** | 12.35 | 2.09 | 10.33 | # | # | **1.55** | 14.05 |
| Fine-tune | 3.35 | 11.35 | **0.88** | 25.3 | 6.21 | 13.18 | **1.44** | 11.16 | # | # | 2.97 | 15.25 |

**Table 4: Improvement of GPT4Battery over GPT4TS using pairwise combinations of five LIBs.**

| Source/target LIB | CALCE | | SANYO | | KOKAM | | PANASONIC | | GOTION | |
|---|---|---|---|---|---|---|---|---|---|---|
| Methods | GPT4TS | GPT4Battery | GPT4TS | GPT4Battery | GPT4TS | GPT4Battery | GPT4TS | GPT4Battery | GPT4TS | GPT4Battery |
| CALCE | 2.31 | 2.09 | 1.37 | 1.36 | 2.84 | 2.07 | 1.73 | 1.69 | 1.38 | 1.12 |
| SANYO | 2.49 | 2.14 | 0.98 | 1.09 | 4.52 | 4.47 | 2.62 | 2.31 | 0.33 | 0.32 |
| KOKAM | 2.31 | 2.42 | 0.74 | 0.54 | 1.58 | 1.49 | 2.04 | 1.88 | 3.89 | 2.04 |
| PANASONIC | 1.96 | 1.95 | 0.52 | 0.44 | 3.72 | 3.19 | 1.53 | 0.84 | 2.66 | 1.77 |
| GOTION | 4.13 | 1.43 | 2.78 | 0.87 | 5.56 | 4 | 4.57 | 0.81 | 0.31 | 0.25 |

**Table 5: Ablating the different components of GPT4Battery. Red: the best.**

| Method | CALCE | SANYO | KOKAM | PANAS. | GOTION | AVE. |
|---|---|---|---|---|---|---|
| GPT4Battery | **1.43** | **0.87** | **4** | **0.81** | **0.25** | **1.47** |
| w/o PG | 2.11 | 1.12 | 4.78 | 3.11 | 0.265 | 2.28 |
| w/o Masked TPT | 2.05 | 0.97 | 4.23 | 1.22 | 0.255 | 1.74 |
| w/o TPT | 2.13 | 1.34 | 5.21 | 3.44 | 0.297 | 2.48 |
| w/o patch reprogram | 2.01 | 1.13 | 4.31 | 1.31 | 0.265 | 1.8 |
| GPT4TS | 4.13 | 2.78 | 5.56 | 4.57 | 0.31 | 3.47 |

on unseen LIBs. Physical guiding is a novel design to serve the TTT for self-supervised learning. We believe the physical rules are anticipated as external knowledge, which can be naturally incorporated via prompting to facilitate the learning and inference. 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 test-time optimization strategy completely. In GPT4Battery, the act of patch 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 **22.4%** degradation on average performance. Additionally, the result shows that masking the sequence tokens is also beneficial, without which eliciting over 18.3% performance degradation averagely.

*4.3.2 PGTPT on different parameter groups of the LLM.* Existing test-time optimization methods have worked on different parameter groups of a model. Although it is intuitive to tune the prompts of the LLM, it is unclear whether this is the most effective choice. In Fig. 5, we evaluate the performance of test-time optimization on four different parameter groups of LLM, i.e., to fine-tune (1) the entire LLM model, (2) the position embeddings and layer normalization layers [28, 59], (3) the prompt tokens, or (4) the prompt and sequence tokens together. We run a grid search using NNI [29] to determine the learning rate and the number of optimization steps and report the best result for a fair comparison.

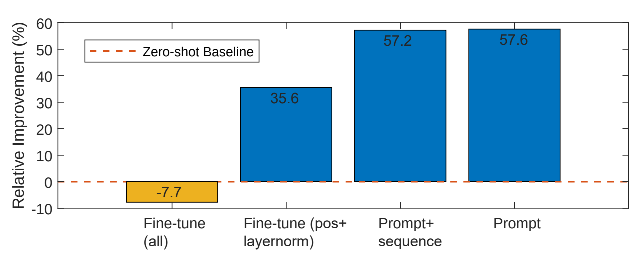



**Figure 5: PGTPT on different parameter groups of the LLM.**

The result in Table 5 reports the relative improvements over GPT4TS on average for the five datasets. The report indicates that tuning the prompt tokens yields the most effective results, with a relative improvement of **57.6%** over GPT4TS. Tuning both the prompt and sequence tokens also yielded comparable results. In addition, we notice that fine-tuning the whole LLM at test time may get drift predictions and obtain an even lower accuracy. This observation is in alignment with previous work in the field of computer vision that fine-tuning can distort pre-trained features and underperform out-of-distribution [21].

# 5 RELATED WORK

In this section, we introduce relate works in *Data-Driven Battery SOH Estimation* (Sec. 5.1), *LLM for Time Series Data* (Sec. 5.2), *Physics Informed Neural Networks* (Sec. 5.3), and *Test-Time Training* (Sec. 5.4) respectively.

## 5.1 Data-Driven Battery SOH Estimation

Data-driven approaches for battery SOH estimation display greater benefits in accuracy and online computation efficiency than modeling-based models such as equivalent circuit models (ECMs) and physics-based models (PBMs) [32]. Popular and widely used SOH methods in battery fields including algorithms such as Bayesian Ridge Regression (BPR), Gaussian process regression (GPR), Random Forest (RF), and Deep ensemble of neural networks (DNNe) [38].

However, it requires time- and resource-consuming degradation experiments to generate lifelong time data for training data-driven models, making a series of data-efficient approaches popular [15, 58]. For instance, [45] developed an LSTM-FC network to predict SOH by fine-tuning only the first 25 % of the target dataset. Similarly, [53] retrained LSTM using only two target battery cells during the transfer learning process. Besides, [27] integrated a swarm of deep neural networks with domain adaptation to enable SOH estimation in the absence of target battery labels. However, existing transfer learning methods still hold an imperfect setting about the target LIB dataset. Under their settings, quite a bit of effort is still needed to collect the target LIB data. Limitations of modeling-based battery state estimation methods and comparisons between them are also well explained in Section 2, inspiring us to build a model that is highly generalizable and preferably interpretable, and more importantly, has a more scenario-fit assumption about target LIB dataset.

## 5.2 LLM for Time Series Data

While LLMs are typically pre-trained on extensive text corpora, they have been proven to be effective in pattern recognition and reasoning over complex sequences of tokens in approximating flexible distributions over numbers [14, 30]. This capability can be well extended to time series data, as demonstrated in this work and in recent concurrent and follow-up studies [8, 14, 42, 59]. For example, GPT4TS [59] leveraged a modified GPT2 on general time series analysis and achieved results comparable to SOTA. TEMPO [7] introduces a learned dictionary of prompts that are retrieved at inference time to reprogram the LLM for time series forecasting. Similarly, Time-llm [19] and $S^2$IP-LLM [34] reprogram the input time series with text prototypes for modality alignment to repurpose the LLM. As an emerging and promising field of study, a recent survey [20] also offers a comprehensive overview of research that utilizes LLMs not only for time series but also for more intricate spatio-temporal data mining. This further underscores the proficiency of LLMs in understanding and reasoning on temporal data. Although LLMs are widely used in time series, the filed of battery research hasn't yet benefited from it. And the limits of its generalization across domains are still not clearly explored.

## 5.3 Physics Informed Neural Networks

Physics-Informed Neural Networks (PINNs) are neural networks that incorporate PDEs and their initial/boundary conditions (IC/BCs) as soft constraints during training [37]. In areas where the training data is more limited, domain knowledge in the form of Partial Differential Equations (PDEs) can describe phenomena such as physics laws or dynamic evolution, serving as inductive biases as a supplement to the lack of data for training the deep learning model. In the field of battery, the most common approach in the literature is to use physics-based models to generate training data for machine learning models [52] . There also have been study on combining basic neural networks with physical models such as P2D to enhance the model's interpretability. However, these methods have not been widely adopted by battery researchers [31].

## 5.4 Test-Time Training

Test-Time Training (TTT) is a general approach for improving the performance of predictive models when there are distribution shifts between training and testing domains by fully utilize the only *one* unlabeled test sample [44]. A typical TTT model has a supervised main task head and a test-time tunable self-supervised head for the unlabeled testing data. We believe that the *only one sample* assumption about the target dataset of TTT fits very well with the real-world scenario of battery usage, where the BMS collects the charging feature curve of LIB once a time as the battery ages in usage. To the best of our knowledge, no prior works have endeavored in this direction and we are the first to explore the utilization of *one* partial charging curve obtained during battery usage and attempt mining the health information in existing sample to enhance the model's generalization ability.

# 6 CONCLUSION

In this work, we investigated how to fully exploit the generalization potential of pre-trained language foundation models for the state of health estimation cross different LIBs. For the field of battery, we provide a reliable solution to battery state estimation that does not require large amounts of data, saving years of labor of data collection while also having some degree of physical interpretability. For the field of AI, we hope this work will inspire the community to fully leverage the generalization potential of LLM for diverse multimedia/multimodal applications. We also hope that this work will inspire the community to explore multimodal fusion techniques including model reprogramming and careful consideration of laws and knowledge within the specific modal domain.

In the future, we plan to dig more on the variant features in addition to SOH estimation that may affect LIBs design, for example, battery charging, power, energy, and validity. Also, it may be helpful to analyze the correlated factors to find out the key issues in modern LIBs design by statistical approaches, which may help improve the corresponding effectiveness.

# REFERENCES


[1] Muratahan Aykol, Chirranjeevi Balaji Gopal, Abraham Anapolsky, Patrick K Herring, Bruis van Vlijmen, Marc D Berliner, Martin Z Bazant, Richard D Braatz, William C Chueh, and Brian D Storey. 2021. Perspective—combining physics and machine learning to predict battery lifetime. *Journal of The Electrochemical Society* 168, 3 (2021), 030525.

[2] Akash Basia, Zineb Simeu-Abazi, Eric Gascard, and Peggy Zwolinski. 2021. Review on State of Health estimation methodologies for lithium-ion batteries in the context of circular economy. *CIRP Journal of Manufacturing Science and Technology* 32 (2021), 517–528.

[3] Christoph Birkl. 2017. Oxford battery degradation dataset 1. (2017).

[4] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258* (2021).

[5] Michel Broussely, Ph Biensan, F Bonhomme, Ph Blanchard, S Herreyre, K Nechev, and RJ Staniewicz. 2005. Main aging mechanisms in Li ion batteries. *Journal of power sources* 146, 1-2 (2005), 90–96.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[7] 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).

[8] 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]

[9] 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.

[10] Marc Doyle, Thomas F Fuller, and John Newman. 1993. Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell. *Journal of the Electrochemical society* 140, 6 (1993), 1526.

[11] Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. 2024. UniTS: Building a Unified Time Series Model. *arXiv preprint arXiv:2403.00131* (2024).

[12] Azul Garza and Max Mergenthaler-Canseco. 2023. TimeGPT-1. *arXiv preprint arXiv:2310.03589* (2023).

[13] 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

[14] 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).

[15] Te Han, Zhe Wang, and Huixing Meng. 2022. End-to-end capacity estimation of Lithium-ion batteries with an enhanced long short-term memory network considering domain adaptation. *Journal of Power Sources* 520 (2022), 230823.

[16] Wei He, Nicholas Williard, Michael Osterman, and Michael Pecht. 2011. Prognostics of lithium-ion batteries based on Dempster–Shafer theory and the Bayesian Monte Carlo method. *Journal of Power Sources* 196, 23 (2011), 10314–10321.

[17] Yicun Huang, Changfu Zou, Yang Li, and Torsten Wik. 2023. MINN: Learning the dynamics of differential-algebraic equations and application to battery modeling. *arXiv preprint arXiv:2304.14422* (2023).

[18] Martin Linder Mikael Hanicke Evan Horetsky Dina Ibrahim Sören Jautelat ukas Torscht Alexandre van de Rijt Jakob Fleischmann, Patrick Schaufuss. 2023. Battery 2030: Resilient, sustainable, and circular. (2023). https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/battery-2030-resilient-sustainable-and-circular#/ January 16, 2023.

[19] 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).

[20] Ming Jin, Qingsong Wen, Yuxuan Liang, Chaoli Zhang, Siqiao Xue, Xue Wang, James Zhang, Yi Wang, Haifeng Chen, Xiaoli Li, Shirui Pan, Vincent S. Tseng, Yu Zheng, Lei Chen, and Hui Xiong. 2023. Large Models for Time Series and Spatio-Temporal Data: A Survey and Outlook. arXiv:2310.10196 [cs.LG]

[21] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. 2022. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054* (2022).

[22] Siddique Latif, Moazzam Shoukat, Fahad Shamshad, Muhammad Usama, Yi Ren, Heriberto Cuayáhuitl, Wenwu Wang, Xulong Zhang, Roberto Togneri, Erik Cambria, and Björn W. Schuller. 2023. Sparks of Large Audio Models: A Survey and Outlook. arXiv:2308.12792 [cs.SD]

[23] Weihan Li, Neil Sengupta, Philipp Dechent, David Howey, Anuradha Annaswamy, and Dirk Uwe Sauer. 2021. One-shot battery degradation trajectory prediction with deep learning. *Journal of Power Sources* 506 (2021), 230024.

[24] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems* 36 (2024).

[25] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. 2023. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625* (2023).

[26] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. 2022. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11976–11986.

[27] 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.

[28] Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. 2022. Frozen pretrained transformers as universal computation engines. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 7628–7636.

[29] Microsoft. 2021. *Neural Network Intelligence*. https://github.com/microsoft/nni

[30] 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]

[31] Sina Navidi, Adam Thelen, Tingkai Li, and Chao Hu. 2024. Physics-informed machine learning for battery degradation diagnostics: A comparison of state-of-the-art methods. *Energy Storage Materials* 68 (April 2024), 103343. https://doi.org/10.1016/j.ensm.2024.103343

[32] 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.

[33] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2022. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730* (2022).

[34] 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).

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.

[36] 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.

[37] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics* 378 (2019), 686–707.

[38] 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.

[39] 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.

[40] Manli Shu, Weili 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.

[41] 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

[42] 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]

[43] 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).

[44] 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.

[45] 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

[46] Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2023. GraphGPT: Graph Instruction Tuning for Large Language Models. arXiv:2310.13023 [cs.CL]

[47] Jinpeng Tian, Rui Xiong, Weixiang Shen, Jiahuan Lu, and Fengchun Sun. 2022. Flexible battery state of health and state of charge estimation using partial charging data and deep learning. *Energy Storage Materials* 51 (2022), 372–381.

[48] 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.

[49] 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

[50] 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).

[51] 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.

[52] Hao Tu, Scott Moura, Yebin Wang, and Huazhen Fang. 2023. Integrating physics-based modeling with machine learning for lithium-ion batteries. *Applied Energy* 329 (2023), 120289.

[53] Yixiu Wang, Jiangong Zhu, Liang Cao, Bhushan Gopaluni, and Yankai Cao. 2023. Long Short-Term Memory Network with Transfer Learning for Lithium-ion Battery Capacity Fade and Cycle Life Prediction. *Applied Energy* 350 (2023), 121660.

[54] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. 2022. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7959–7971.

[55] 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.

[56] 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.

[57] 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.

[58] Zhuang Ye and Jianbo Yu. 2021. State-of-health estimation for lithium-ion batteries using domain adversarial transfer learning. *IEEE Transactions on Power Electronics* 37, 3 (2021), 3528–3543.

[59] 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).