Source-Free Cross-Domain State of Charge Estimation of Lithium-Ion Batteries at Different Ambient Temperatures

Liyuan Shen, Jingjing Li, Lin Zuo, Lei Zhu, *Senior Member, IEEE*, and Heng Tao Shen, *Fellow, IEEE*

***Abstract*—Machine learning methods for state of charge (SOC) estimation of lithium-ion batteries (LiBs) face the problem of domain shift. Varying conditions, such as different ambient temperatures, can cause performance degradation of the estimators due to data distribution discrepancy. Some transfer learning methods have been utilized to tackle the problem. At real-time transfer, the source model is supposed to keep updating itself online. In the process, source domain data are usually absent because the storage and acquisition of all historical running data can involve violating the privacy of users. However, existing methods require coexistence of source and target samples. In this article, we discuss a more difficult yet more practical source-free setting where there are only the models pretrained in source domain and limited target data can be available. To address the challenges of the absence of source data and distribution discrepancy in cross-domain SOC estimation, we propose a novel source-free temperature transfer network (SFTTN), which can mitigate domain shift adaptively. In this article, cross-domain SOC estimation under source-free transfer setting is discussed for the first time. To this end, we define an effective approach named minimum estimation discrepancy (MED), which attempts to align domain distributions by minimizing the estimation discrepancy of target samples. Extensive transfer experiments and online testing at fixed and changing ambient temperatures are performed to verify the effectiveness of SFTTN. The experiment results indicate that SFTTN can achieve robust and accurate SOC estimation at different ambient temperatures under source-free scenario.**

***Index Terms*—Lithium-ion batteries (LiBs), semisupervised learning, source-free domain adaptation, state of charge (SOC) estimation, transfer learning, unsupervised learning.**

Manuscript received 7 November 2022; revised 8 January 2023 and 14 February 2023; accepted 23 February 2023. Date of publication 2 March 2023; date of current version 20 April 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62176042, in part by the Natural Science Foundation of Sichuan, China under Grant 2023NSFSC0483, and in part by the Fundamental Research Funds for the Central Universities (UESTC) under Grant ZYGX2021YGCX016. Recommended for publication by Associate Editor G. Oriti. *(Corresponding author: Jingjing Li.)*

Liyuan Shen, Jingjing Li, Lin Zuo, and Heng Tao Shen are with the University of Electronic Science and Technology of China, Chengdu 610056, China (e-mail: 546466383@qq.com; lijin117@yeah.net; linzuo@uestc.edu.cn; shenhengtao@hotmail.com).

Lei Zhu is with the Shandong Normal University, Jinan 250061, China (e-mail: lijin17@yeah.net).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TPEL.2023.3251568.

Digital Object Identifier 10.1109/TPEL.2023.3251568

I. Introduction

With low manufacturing cost and high energy density, lithium-ion batteries (LiBs) have become the most widely used energy storage devices, which are considered to be the ideal power source for electric vehicles (EVs) [1]. Accurate state of charge (SOC) estimation helps us to assess the health status of LiBs and it ensures a safe working environment for power system of EV by protecting the battery from overcharge. The SOC is defined as the ratio of the remaining capacity to the normal capacity of the battery. However, the SOC can only be estimated indirectly through observable parameters of the battery, such as voltage and current, making estimating SOC accurately difficult [2].

Accurate SOC estimation has received a lot of attention and many accurate SOC estimation methods have been proposed. These methods can be roughly grouped into the following five categories: book-keeping-based methods, look-up table methods, filter-based methods, model-based methods, and data-driven methods [3], [4]. Book-keeping-based methods utilize battery discharging current data as input to estimate SOC, including Coulomb counting method and modified Coulomb counting method [5]. Coulomb counting method estimates the SOC value by measuring the discharging current of the battery and integrating the current over time. Look-up methods infer the SOC by looking up the table built with the relationship between SOC and measured parameters, such as open-circuit voltage (OCV). For filter-based methods, lots of adaptive filters are employed to infer the internal state of LiBs. Model-based methods aim to model the battery behavior by involving electrical and chemical properties of the battery. Such approaches usually require domain-specific knowledge and complicated procedures [6], [7]. Data-driven methods employing machine learning techniques are completely different from model-based methods. They are built upon extensive labeled data [8], [9]. Recently, more deep learning networks including long short-term memory (LSTM) [10], [11], convolutional neural network (CNN) [12], [13], and gated recurrent unit (GRU) [14] have been employed for accurate SOC estimation.

Despite the success of SOC estimation methods, there are still practical problems. One challenge is distribution discrepancy where the training data and testing data have different distributions. Data difference is caused by many factors including the difference of ambient temperatures, the aging of the battery pack,

and other external or internal factors. For example, in real-world scenarios, LiBs are working under varying ambient temperatures. One battery is trained at temperature of 25 °C in the laboratory while it works at 20 °C, which will cause performance degradation of pretrained estimators due to data distribution discrepancy. It is unrealistic to perform experiments covering every temperature. Even if the training data covers the whole temperature range, distribution discrepancy still exists because of many other factors. Furthermore, in real-time applications, the estimator is supposed to keep updating itself online to adapt to new working environment, which means there is no labeled target data to train a reliable estimator from scratch.

To address the problem of distribution discrepancy, many transfer learning methods [15], [16], [17] can be utilized. Transfer learning methods can be divided into the following three groups: fine-tuning methods, metric-based adaptation methods, and adversarial adaptation methods. The fine-tuning approaches freeze the weights of the pretrained source domain model on certain layers while other unfrozen layers are fine-tuned using the labeled target domain data. Metric-based adaptation methods minimize a statistic criterion directly to mitigate the distribution discrepancy adaptively, among which maximum mean discrepancy (MMD) [18] and maximum classifier discrepancy (MCD) [19] are widely used. Adversarial adaptation methods [20], [21], [22] utilize a domain discriminator to distinguish the source domain data from the target domain data through extracted features, while the feature extraction network aims to minimize domain divergence by generating features to fool the domain discriminator. These methods attempt to mitigate the domain shift by minimizing data distribution divergence.

Nevertheless, transfer SOC estimation methods [23], [24] fail under practical source-free setting where the source data are absent. For real-time adaptation, the SOC estimator is expected to adapt to new environment, which seems pretty easy because many transfer methods, such as MMD and MCD have been put forward and the only requirement is to obtain the source data that is used to pretrain the estimator. However, in real-world situations, source data are usually inaccessible because the storage and acquisition of all historical running data can involve violating the privacy of users of EVs. Besides, it is unrealistic for the battery management system (BMS) to store all the historical running data. The absence of source data invalidates abovementioned transfer methods that require the coexistence of source and target data. Thus, it is necessary to propose effective methods for source-free cross-domain SOC estimation.

To address the challenge of source data absence, a novel source-free temperature transfer network (SFTTN) is proposed in the article. We propose a simple yet effective formulation named minimum estimation discrepancy (MED), which can guide the feature extractor to learn a robust feature representation and transfer the knowledge of source domain from the source pretrained estimators. Specifically, the feature extractor learns to generate domain-invariant features by minimizing the estimation discrepancy of target samples. The main contributions of this article are summarized as follows.

1) In this article, a more realistic and challenging setting named source-free cross-domain SOC estimation where source data are absent at transfer stage is discussed for the first time. The absence of source data invalidates existing cross-domain methods and makes it extremely difficult to train a reliable cross-domain estimator.
2) A novel SFTTN is proposed to achieve accurate SOC estimation at different ambient temperatures without source data. By minimizing the estimation discrepancy of target data, the model learns robust feature representation and the distribution discrepancy is mitigated.
3) Pseudolabeling is employed to make better use of massive unlabeled target data. The estimator pretrained in the source domain is used to generate pseudolabels for unlabeled target data. We propose a label-selection method to select reliable and smooth pseudolabels.
4) Extensive transfer experiments at fixed and changing temperatures are performed under semisupervised and unsupervised scenarios. The results indicate that SFTTN can achieve robust and accurate SOC estimation under source-free condition. Furthermore, cross-battery transfer experiments and online testing have also been performed to show the superiority of SFTTN.

II. Related Work

A. SOC Estimation

For book-keeping-based methods, Ng et al. [25] proposed a smart SOC estimation method based on Coulomb counting to improve the estimation accuracy and state of health (SOH) was evaluated by the maximum releasable capacity. As for modified Coulomb counting, Ningrum et al. [26] used corrected current, the function of discharging current, to achieve accurate estimation. Direct-counting methods are simply but they require stable discharge current. A small amount of noise can have big impact on the results because of the integration operation.

For look-up table methods, Ren et al. [27] compared the low current OCV tests with the incremental OCV tests, which applied 0.2, 0.3, 0.5, and 1 C current-rates and investigated the optimal relaxation time for incremental OCV test from the perspective of test accuracy and test duration. Gismero et al. [28] analyzed the behavior of the batteries at different current and temperature conditions to adjust the charge measurement and used OCV to reset the SOC estimation for preventing the error accumulation. Other widely used empirical assessment methods include electromotive force [29], internal resistance [30], and electrochemical impedance spectroscopy (EIS) [31]. The OCV method is simple and accurate but it needs long resting time to reach the equilibrium state.

In filter-based methods, particle filter (PF), Kalman filter (KF), and its variants [32] are used most frequently. Tian et al. [33] combined an adaptive cubature Kalman filter (ACKF) with an LSTM to learn the nonlinear relationship between the measured parameters and SOC values. The ACKF was applied to smooth the outputs.

For model-based methods, Naseri et al. [34] proposed a block-oriented model composed of a classical equivalent circuit model (ECM) and a static nonlinearity block. Fan

et al. [35] designed a robust estimator based on an electrochemical model (EM). The reduced-order EM incorporated temperature-dependent properties and a robust AEKF based on the model was employed to actively compensate potential model uncertainties. Lin et al. [36] proposed an accurate ECM combined with adaptive unscented Kalman filter (AUKF) with approximately 0.95% root mean square error (RMSE).

As for data-driven methods, they are built without requirement of background knowledge. By training a deep neural network (NN), the measured voltage, current, temperature, and even reflected ultrasonic signals [37] can be mapped directly to SOC. With limited prior information about internal characteristics of the battery, data-driven approaches are able to achieve accurate SOC estimation. Hannan et al. [38] employed GRU for accurate SOC estimation achieving a minimum RMSE of 0.52% on the training dataset and 0.65% on the testing dataset. From the perspective of data collection, Li et al. [39] proposed an efficient data generation method for machine learning training. This method reduced the training and running time while achieving accurate estimation with mean absolute error (MAE) of 0.358%. For better time-varying and nonlinear feature extraction, Hannan et al. [40] proposed deep fully convolutional neural network (CNN) model, which outperformed other commonly used deep learning models with RMSE and MAE of 0.85% and 0.7%. Different from other works, Galiounas et al. [37] achieved accurate estimation with MAE of 0.75% using machine learning analysis of reflected ultrasonic signals from battery internal structures.

B. Cross-Domain Transfer

For fine-tuning methods, Shen et al. [41] utilized deep convolutional neural network (DCNN) for pretraining and then applied fine-tuning to transfer the learned parameters from the source domain to the target domain. Analogously, Vidal et al. [42] trained the DNN model for one battery type and then fine-tuned the model on another battery type using the pretrained parameter values as a starting point. Liu et al. [43] applied fine-tuning on a temporal convolutional network to transfer knowledge. However, the efficiency of fine-tuning methods largely depends on the data equality, which makes the accuracy of SOC estimation unstable. For metric-based methods, Bian et al. [44] exploited MMD metric on multiscale layers to mitigate the domain divergence.

Although abovementioned transfer methods show effectiveness, they fail to deal with more practical source-free transfer setting. By contrast, SFTTN shows potential in cross-domain SOC estimation in source-free condition. Furthermore, SFTTN has lower computational complexity than existing transfer methods.

III. Proposed Method

A. Problem Statement

Normally, SOC refers to the ratio of the remaining capacity to nominal capacity and the mathematical definition is shown as follows:

$$\text{SOC} = \frac{Q_{\text{remaining}}}{Q_{\text{rate}}} \tag{1}$$

Algorithm 1: Pseudo Code of SFTTN.

Input: N: Number of target samples;
$\boldsymbol{x} = \{x_i\}_1^N$: Unlabelled target training data;
f_{E1}, f_{E2}: Two different pre-trained source estimators;
f_F^S: Pre-trained source feature extractor;
$epoch_{thresh}$: Threshold number of training epochs;
λ_1, λ_2: Trade-off parameters.
Output: f_T: Transferred target model.

```
Generate pseudo labels ỹ ← (f_E1(f_F^S(x)) + f_E2(f_F^S(x)))/2
Initialize target feature extractor f_F^T with pre-trained source feature extractor f_F^S
for epoch ← 0 to epoch_thresh do
    Compute estimation value ŷ^1 ← f_E1(f_F^T(x))
    Compute estimation value ŷ^2 ← f_E2(f_F^T(x))
    Compute estimation loss
    L_SOC ← (1/N) Σ_{i←1}^N ((ỹ_i − ŷ_i^1)^2 + (ỹ_i − ŷ_i^2)^2)
    Compute estimation discrepancy
    L_Dis ← (1/N) Σ_{i←1}^N (|ŷ_i^1 − ŷ_i^2|)
    Compute loss function
    L ← λ_SOC L_SOC + λ_Dis L_Dis
    Compute model update Δf_F^T by applying gradient descent algorithm with L
    Update target feature extractor f_F^T ← f_F^T + Δf_F^T
    epoch ← epoch + 1
end for
Combine target feature extractor f_F^T with either estimator f_E1/f_E2 to get transferred target model f_T
```

where $Q_{\text{remaining}}$ is the remaining capacity and Q_{rate} is current rated capacity of the battery. There is an uncertain nonlinear relationship between SOC value and measurable parameters of the battery. The proposed model uses the following ones as input features among various parameters: voltage, current, and temperature. Let the total span of data series be L and the length of time window be M. The dataset is given by $\{(\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, (\boldsymbol{x}_N, \boldsymbol{y}_N)\}$, where $N = L/M$ represents the number of time slices. For the ith sequence $\mathbf{x}_i = \{\boldsymbol{P}_{i,1}, \ldots, \boldsymbol{P}_{i,M}\}$, the three-dimensional vector $\boldsymbol{P}_{i,t} = (V_t, I_t, T_t)$ represents the measured voltage, current, and temperature of the battery at time step t. The SOC ground truth labels of sequential $\mathbf{x}_i$ is given by $\mathbf{y}_i = \{\text{SOC}_{i,1}, \ldots, \text{SOC}_{i,M}\}$.

In the context of transfer learning of SOC estimation, a domain $\mathcal{D}$ is generally composed of a feature space X and a marginal probability distribution $P(X)$ [45]. Considering the source data is absent under source-free setting, the notations can be formulated as follows:

$$\mathcal{D}_T = \{\boldsymbol{x}_i^T\}_{i=1}^{N_{\text{unlab}}^T} \cup \{\boldsymbol{x}_j^T, \boldsymbol{y}_j^T\}_{j=N_{\text{unlab}}^T+1}^{N_{\text{unlab}}^T+N_{\text{lab}}^T} \tag{2}$$

where $\mathcal{D}_T$ represents the target domain, N_{unlab}^T and N_{lab}^T are the numbers of unlabeled and labeled samples, respectively.

B. Source-Free Cross-Domain Transfer

In real-world estimation, the source data are absent for reasons of confidentiality and intellectual property. For a better transfer

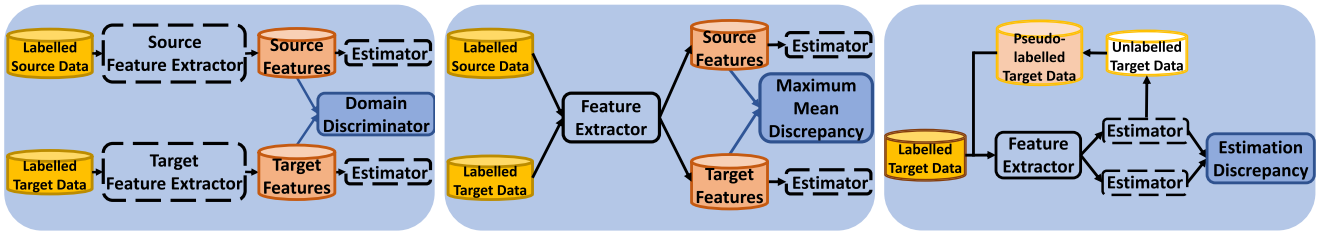

Fig. 1. Overview of the structures of (a) adversarial domain adaptation, (b) MMD, and (c) SFTTN. Adversarial training framework consists of two different feature extractors while MMD and SFTTN share one feature extractor. From the side of data input, both adversarial training and MMD require access of source and target samples. On the contrary, SFTTN can work on the absence of source data. The weights of models in dashed line are frozen while the others are trainable.

performance, a source-free transfer framework is proposed. Fig. 1 shows the different structures of adversarial domain adaptation, MMD, and SFTTN. Adversarial training framework consists of two different feature extractors while MMD and SFTTN share one feature extractor. From the side of data input, both adversarial training and MMD require access of source and target samples. On the contrary, SFTTN can work on the absence of source data.

The structure of SFTTN contains two parts: the feature extractor and estimators. There are two estimators pretrained with source data, which have learned domain-specific knowledge of source domain and the weights of the estimators are frozen. The feature extractor attempts to learn domain-invariant feature representation under the guidance of estimators by minimizing estimation discrepancy without source data.

Let $\mathcal{X}$, $\mathcal{Z}$, $\mathcal{Y}_{\text{data}}$ be the input, feature, and data label spaces. The transfer model consists of two parts: a feature extractor $f_F : \mathcal{X} \to \mathcal{Z}$ and SOC estimators $f_{E_1} : \mathcal{Z} \to \mathcal{Y}_{\text{data}}$, $f_{E_2} : \mathcal{Z} \to \mathcal{Y}_{\text{data}}$ whose parameters are θ_F, θ_{E_1}, and θ_{E_2}.

There are three steps at the transfer stage. First, to improve knowledge transfer performance, estimators are utilized to generate pseudolabels for unlabeled data. Then, target data are input into the feature extractor to generate time-varying features. Finally, the features are fed to the estimators to calculate target data estimation discrepancy for backward propagation. By minimizing the estimation discrepancy, target and source feature distributions can be aligned.

The pseudocode of SFTTN is shown in Algorithm 1.

C. Pseudolabeling

Before transfer training on the target domain, pseudolabeling, a simple yet effective semisupervised mechanism, is employed to generate pseudolabels for massive unlabeled target data for a better transfer performance. Let f_{S_1} and f_{S_2} be the models pretrained in source domain. The generation of pseudolabels is given as follows:

$$\widetilde{\boldsymbol{y}}_i = \frac{f_{S_1}(\boldsymbol{x}_i) + f_{S_2}(\boldsymbol{x}_i)}{2}, 1 \le i \le N^T_{\text{unlab}} \tag{3}$$

where $\widetilde{\boldsymbol{y}}_i$ represents the pseudolabels generated by pretrained model on time sequence slice $\boldsymbol{x}_i$, N^T_{unlab} is the number of unlabeled target data.

Some of the pseudolabels generated by the pretrained source model are away from real labels due to the existence of domain divergence between domains. Unlike the fixed number of categories in classification problem, the result of SOC estimation is relatively uncertain, and there is no confidence coefficient to select reliable labels. Thus, it is challenging and important to select reliable pseudolabels to improve the transfer learning instead of ruining it. A simple yet effective approach is used to calculate confidence coefficient of the labels to filter reliable ones. Drastic changes, such as sudden increment or sudden decreasement, can be found in the estimation results because of distribution discrepancy between domains. For each pseudolabel, the mean difference is calculated as the confidence coefficient

$$\text{Conf}_i = 1 - \sum_{j=1}^{N_{\text{Conf}}} (\widetilde{\boldsymbol{y}}_i - \widetilde{\boldsymbol{y}}_{i+I_j}) \tag{4}$$

where Conf_t denotes the confidence coefficient of pseudolabel, N_{Conf} is the number of pseudolabels taken into account, and I_j is the jth index increasement. Let $\tilde{\boldsymbol{y}} = \phi$, representing the pseudolabels selected from $\widetilde{\boldsymbol{y}}$. For each pseudolabel $\widetilde{\boldsymbol{y}}_i$, the following rule is applied:

$$\tilde{\boldsymbol{y}} = \begin{cases} \tilde{\boldsymbol{y}}, & \text{Conf}_i \le \tau \\ \tilde{\boldsymbol{y}} \cup \{\widetilde{\boldsymbol{y}}_i\}, & \text{Conf}_i > \tau \end{cases} \tag{5}$$

where $\tilde{\boldsymbol{y}}$ is the filtered pseudolabels, $\widetilde{\boldsymbol{y}}_i$ is the ith unfiltered pseudovalue, Conf_i is the confidence coefficient of $\widetilde{\boldsymbol{y}}_i$, and τ is the confidence threshold.

After the selection of pseudolabels, the definitions of domains can be rewritten as follows:

$$\mathcal{D}_T = \{\boldsymbol{x}^T_i, \boldsymbol{y}^T_i\}_{i=1}^{N^T_{\text{pseudo}}} \cup \{\boldsymbol{x}^T_j, \boldsymbol{y}^T_j\}_{j=N^T_{\text{pseudo}}+1}^{N^T_{\text{pseudo}}+N^T_{\text{lab}}} \tag{6}$$

where N^T_{pseudo} is the number of pseudolabeled samples.

D. Feature Extractor

We employ two-dimensional CNN (2D-CNN) and bidirectional LSTM (BiLSTM) as the feature extractor to better extract features. The time series are sent from the input layer to feature extractor for temporal feature extraction in both forward and backward directions. Generally, the extracted high-dimensional features are not variables with specific physical significance, such as voltage or current. However, they usually have a stronger

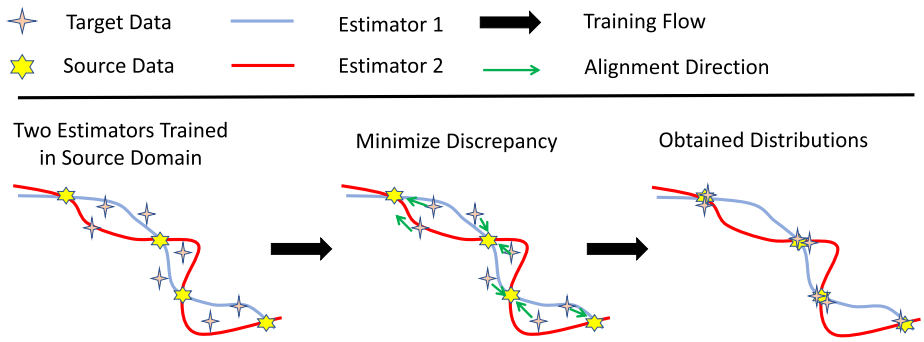

Fig. 2. Example of two different estimators with an overview of the proposed method. Discrepancy refers to the divergence between the estimation values of two estimators. As shown in the left subfigure, there are two different estimators trained in source domain which perform well for source data. Because of data distribution shift, some target samples are outside the support of the estimators. By minimizing estimation discrepancy on target data, target feature distribution is aligned toward source feature distribution as shown in the middle and right subfigures. (best viewed in color).

correlation with the dependent variable. Finally, the features are fed to the fully connected (FC) layer to estimate SOC values.

1) CNN Layer: Similar with [43], we use 2D-convolutional layers to extract nonlinear features. To alleviate overfitting, dropout [46] is utilized, where different neurons are removed from on a temporary basis. The rectified linear unit [47] is chosen as the activation function to accelerate the training speed and avoid the problem of gradient vanishment. The features from CNN layers are given as follows:

$$\boldsymbol{c}_i = f_C(\boldsymbol{x}_i) \tag{7}$$

where $\boldsymbol{c}_i$ is the output feature of the ith time sequence generated by the CNN layer, $\boldsymbol{x}_i$ is the ith time sequential slice of input data, f_C denotes the CNN layer model. Then, the feature set is passed to the BiLSTM layer.

2) BiLSTM Layer: Recurrent neural network models with LSTM are widely used for time series processing.

However, LSTM is limited by the fact that it can only read information in one direction. Recently, BiLSTM [48] has been proposed to replace traditional LSTM. BiLSTM can read inputs backwards and forwards, which allows it to use future context for better feature extraction. The hidden state of time slice i in BiLSTM layers is computed as follows:

$$h_{i,t} = f_{Bi}(c_{i,t}, \overrightarrow{h}_{i,t-1}, \overleftarrow{h}_{i,t+1}) \tag{8}$$

where h_t is the hidden state of time slice i at time t, f_{Bi} represents the BiLSTM layer model, $c_{i,t}$ is the input features at time t, $\overrightarrow{h}$ denotes the forward hidden state, and $\overleftarrow{h}$ denotes the backward hidden state.

The output feature of the combined CNN and BiLSTM layers can be reformulated as follows:

$$\boldsymbol{F}_i = f_F(\boldsymbol{x}_i) \tag{9}$$

where f_F represents the model of the feature extractor.

E. Minimum Estimation Discrepancy

In this article, we propose a novel transfer approach for cross-domain SOC estimation named MED. Inspired by MCD [19], MED exploit different source pretrained models to guide the target feature extraction. However, similar with other metric-based transfer methods, MCD requires coexistence of source and target samples while our proposed MED works well at source-free scenario where only source pretrained models and target data can be accessible.

The example with an overview of the proposed method is shown in Fig. 2. MED attempts to align target and source feature distributions by minimizing estimation discrepancy on target samples. The discrepancy refers to the divergence between the estimation values of two estimators.

First, FC layers are utilized as the estimator and the estimation values can be generated as follows:

$$\hat{\boldsymbol{y}}_i^1 = f_{E_1}(\boldsymbol{F_i}) \tag{10}$$

$$\hat{\boldsymbol{y}}_i^2 = f_{E_2}(\boldsymbol{F_i}) \tag{11}$$

where $\boldsymbol{F_i}$ is the features generated by feature extractor, f_{E_1} and f_{E_2} denote the models of estimator E_1 and E_2, $\hat{\boldsymbol{y}}_i^1$ and $\hat{\boldsymbol{y}}_i^2$ represent the predicted labels for the ith time slice of M time steps obtained by estimator E_1 and E_2, respectively. For each training slice, the model outputs the SOC estimation values in the form of a sequence $\hat{\boldsymbol{y}}_i = \{\hat{y}_{i,1}, \hat{y}_{i,2}, \ldots, \hat{y}_{i,M}\}$. The loss term on target estimation values can be defined

$$\mathcal{L}_{\text{SOC}} = \frac{1}{N}\sum_{i=1}^{N}(\ell(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^1) + \ell(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^2)) \tag{12}$$

where N is the number of samples, $\boldsymbol{y}_i$ represents the ground truth labels, $\hat{\boldsymbol{y}}_i^1$ and $\hat{\boldsymbol{y}}_i^2$ are predicted labels of the ith time slice of length M obtained by estimator E_1 and E_2, and $\ell(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = \sum_{j=1}^{M}(y_{i,j} - \hat{y}_{i,j})^2$.

With estimation values generated, the estimation discrepancy can be calculated as follows:

$$d(\hat{\boldsymbol{y}}_i^1, \hat{\boldsymbol{y}}_i^2) = \frac{1}{M}\sum_{j=1}^{M}|\hat{\boldsymbol{y}}_{i,j}^1 - \hat{\boldsymbol{y}}_{i,j}^2| \tag{13}$$

where $d(\hat{\boldsymbol{y}}_i^1, \hat{\boldsymbol{y}}_i^2)$ denotes the estimation discrepancy for the ith target time slice of length M. Then, the learning objective can

TABLE I
BATTERY DATASET DESCRIPTION

Dataset	Year	Institution	Battery	Capacity	Ambient Temperature($^\circ C$)	Driving Schedule
Panasonic 18650PF [48]	2018	The University of Wisconsin-Madison	$LiNiCoAlO_2$	2.9 Ah	–20,–10,0,10,25	Cycle1-4,UDDS,US06,LA92,HWFET,NN
LG 18650HG2 [49]	2020	McMaster University	$Li(NiMnCo)O_2$	3 Ah	–20,–10,0,10,25	mix1-8,UDDS,US06,LA92,HWFET
A123 [50]	2012	CALCE	$LiFePO_4$	2.23 Ah	–10,0,10,20,25,30,40,50	DST, FUDS

be given as follows:

$$\mathcal{L}_{\text{Dis}} = \frac{1}{N}\sum_{i=1}^{N} d(\hat{\boldsymbol{y}}_i^1, \hat{\boldsymbol{y}}_i^2). \tag{14}$$

The total learning objective can be summarized as follows:

$$\min_{\theta_F} \mathcal{L} = \lambda_{\text{SOC}}\mathcal{L}_{\text{SOC}} + \lambda_{\text{Dis}}\mathcal{L}_{\text{Dis}} \tag{15}$$

where λ_{SOC} and λ_{Dis} are tradeoff parameters, $\mathcal{L}_{\text{SOC}}$ and $\mathcal{L}_{\text{Dis}}$ are given by (12) and (14).

F. Loss Function in Semisupervised Scenario

In semisupervised scenario, both labeled data and unlabeled data can be available, so the loss term of the estimator is slightly different from (12) and it can be formulated as follows:

$$\mathcal{L}_{\text{SOC}} = \frac{1}{N}\sum_{i=1}^{N}(\ell(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^1) + \ell(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^2)), N = N_{\text{pseudo}}^T + N_{\text{lab}}^T \tag{16}$$

where N_{pseudo}^T and N_{lab}^T are the numbers of pseudolabeled and labeled target samples, respectively.

G. Loss Function in Unsupervised Scenario

In unsupervised scenario, only massive unlabeled data in the target domain is accessible. To train a reliable SOC estimator, reliable pseudolabels need to be generated. Under the proposed universal transfer framework, the estimation can be trained using the same loss function as (15) with a slightly difference on the loss term of $\mathcal{L}_{\text{SOC}}$. Due to the lack of labeled data, the number of samples that can be calculated is N_{pseudo}^T. Then, we can formulate the loss term of the estimator

$$\mathcal{L}_{\text{SOC}} = \frac{1}{N}\sum_{i=1}^{N}(\ell(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^1) + \ell(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^2)), N = N_{\text{pseudo}}^T. \tag{17}$$

The definition of the rest loss term $\mathcal{L}_{\text{Dis}}$ is the same as (14).

IV. EXPERIMENTS

In this section, the description of datasets and experiments implementation details will be discussed.

A. Dataset Description

To evaluate the cross-domain SOC estimation performance of our method at real-world BMS conditions and compare our method with other methods, we choose Panasonic 18650PF dataset [49], LG 18650HG2 dataset [53], and A123 dataset [51] as the evaluation datasets and carry out abundant experiments. The dataset specification is shown in Table I.

Panasonic 18650PF dataset is tested on a brand-new Panasonic NCR18650PF battery and collected by the University of Wisconsin-Madison. It is a commonly used validation dataset in the community. In the Panasonic dataset, a series of drive cycles including Cycle1, Cycle2, Cycle3, Cycle4, US06, HWFET, UDDS, LA92, and neural network (NN) were performed repeatedly at ambient temperatures of 25 °C, 10 °C, 0 °C, −10 °C, and −20 °C. Cycles 1–4 consist of random mixture of US06, HWFET, UDDS, LA92, and NN drive cycles. NN drive cycle consists of combination of portions of US06 and LA92 drive cycles. The drive cycle power profile is calculated for an electric Ford F150 truck with a 35 kWh battery pack scaled for a single 18650PF cell.

In the LG dataset, the tests were performed at McMaster University in Hamilton, ON, Canada by Dr. Phillip Kollmeyer. A brand new 3Ah LG HG2 cell was tested in an 8 cu.ft. thermal chamber with a 75 amp, 5 V Digatron Firing Circuits Universal Battery Tester channel with a voltage and current accuracy of 0.1% of full scale.

In the A123 dataset, experiments with A123 cell were conducted wherein two dynamic tests were used: Dynamic stress test (DST) and the Federal urban driving schedule (FUDS), former to identify the model parameters while later to validate the performance of the SOC estimation. The experiments were performed repeatedly at ambient temperatures of 50 °C, 40 °C, 30 °C, 25 °C, 20 °C, 10 °C, 0 °C, and −10 °C.

Data normalization is performed to prevent different ranges of training data from affecting the training process. The normalization on input s can be defined as follows:

$$s' = \frac{s - s_{\min}}{s_{\max} - s_{\min}} \tag{18}$$

where s' denotes the normalized data, $s_{\min}$ and $s_{\max}$ represent the maximum and minimum values of the row data. Fig. 3 shows the measured parameters and SOC values of LA92 tested at five discrete temperatures.

B. Implementation Details

SFTTN is implemented in PyTorch with NIVIDIA GeForce RTX 2080 Ti employed as the GPU platform. Model structure setting is shown in Table II. As shown in Table III, SFTTN requires the lowest computational complexity (FLOPs, parameters, training time for each epoch and average estimation time for each second of testing data) compared with existing transfer

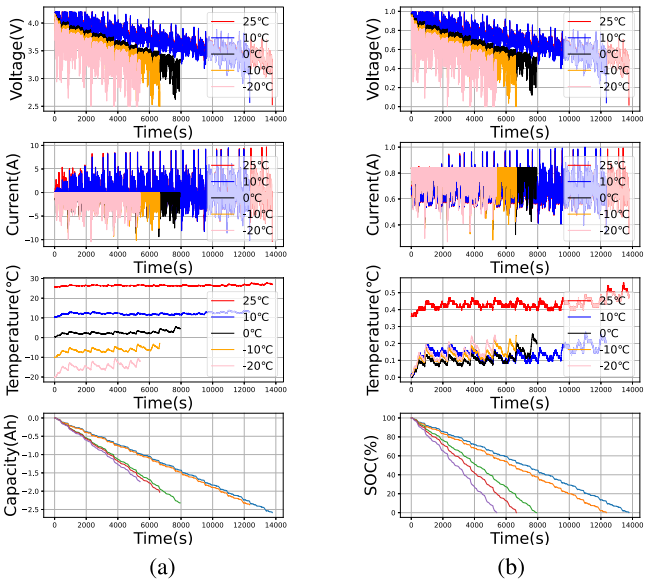

Fig. 3. Voltage, current, temperature, and capacity of LA92 drive cycle in raw (a) and normalized (b) measured at five discrete ambient temperatures. (a) Raw parameters of LA92. (b) Normalized parameters of LA92.

TABLE II
MODEL STRUCTURE AND PARAMETERS

Structure	Setting	Trainable parameters
2D-CNN layer	3×1, 256	197 888
BiLSTM layer	64 hidden units	734 992
FC layer	$128 \rightarrow 128, 128 \rightarrow 1$	16 641

TABLE III
COMPUTATIONAL COMPLEXITY OF DIFFERENT TRANSFER METHODS

Method	FLOPs	Parameters	Training time	Estimation time
MMD [51]	13.42 M	0.95 M	51.3 s	2.40 ms
Adversarial training [20]	15.63 M	0.97 M	41.56 s	2.43 ms
SFTTN (ours)	10.02 M	0.95 M	38.12 s	2.35 ms

methods including MMD and adversarial training. Low computational complexity of SFTTN allows it to be easily applied on current BMS for practical applications. Furthermore, model distillation technology can compress the scale of the SFTTN model to address more stringent memory and efficiency requirements.

Each experiment is repeated three times to get the mean results. In this article, MAE and RMSE are used as the performance evaluation metrics:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i|$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i)^2} \tag{19}$$

where N is the number of evaluated samples, $\boldsymbol{y}_i$ and $\hat{\boldsymbol{y}}_i$ are the real SOC value and the predicted value.

TABLE IV
DATASET SETTING FOR CROSS-DOMAIN TRANSFER EXPERIMENT UNDER DIFFERENT AMBIENT TEMPERATURES

Group	Source domain	Target domain
1	25 °C	10 °C / 0 °C / −10 °C / −20 °C
2	10 °C	25 °C / 0 °C / −10 °C / −20 °C
3	0 °C	25 °C / 10 °C / −10 °C / −20 °C
4	−10 °C	25 °C / 10 °C / 0 °C / −20 °C
5	−20 °C	25 °C / 10 °C / 0 °C / −10 °C

C. Experiment Setting

In the experiments transferring from Panasonic 18650PF to LG 18650HG2, Panasonic dataset is the training dataset and LG dataset is the testing dataset. In other experiments, Panasonic dataset is the training and testing dataset. The specific transfer setting is shown in Table IV.

1) SOC Estimation at Fixed Ambient Temperatures: Before transfer estimation experiments, we evaluate the pretraining performance of our model at each fixed ambient temperature. One drive cycle is used as the testing data and the rest of the drive cycles under the same ambient temperature are used as the training data. Thus, nine experiments are performed for each ambient temperature.

2) Source-Free Cross-Domain Transfer Experiment on Unsupervised Scenario: In this article, source-free transfer SOC estimation under unsupervised scenario is discussed for the first time where only massive unlabeled target data can be available. The experiments are performed under five discrete ambient temperatures with one drive cycle performed as the testing data and the remaining ones served as the unlabeled training data.

3) Source-Free Cross-Domain Transfer Experiment From Panasonic 18650PF to LG 18650HG2 and A123: Our method can even pretrain the model with data of battery type and then retrain it for another battery type. In the experiments, we first pretrain the model using Panasonic 18650PF dataset at each ambient temperature. Then, the model is tested using LG 18650HG2 dataset and A123 dataset for all ambient temperatures.

4) Source-Free Online Testing at Changing Temperatures: In real-world applications, the ambient temperature keeps changing. The SOC estimator is supposed to update itself adaptively to adapt to new testing data. To evaluate the effectiveness of our method in practical scenarios, we perform online testing experiments at changing temperatures. We take Panasonic 18650PF dataset as the validation dataset. The nine drive cycles are repeated with a starting chamber temperature of −20 °C, which is then allowed to drift upwards such that the battery temperature rises up to 20 °C during the drive cycle.

As shown in Fig. 4, the process contains pretraining and transfer stages. The pretraining stage is offline. It takes historical battery data to pretrain models in source domain. Different from abovementioned experiments, for onboard BMS applications, the transfer stage is online. Testing data (target domain data) without labels will be collected by the BMS in real time.

Another difference is that the training data is not input into the model at once. One drive cycle is split into time slices. The slices are input into the model by piece in order of time. During

TABLE V
TRADITIONAL SOC ESTIMATION RESULTS UNDER FIXED AMBIENT TEMPERATURES

Network structure	Metrics	Ambient temperature					Avg.
		25 °C	10 °C	0 °C	−10 °C	−20 °C	
GRU [14]	MAE(%)	1.09	1.38	0.641	1.12	1.70	1.19
	RMSE(%)	3.07	3.30	1.60	1.94	2.55	2.49
LSTM [10]	MAE(%)	1.09	1.74	0.631	**1.11**	2.39	1.62
	RMSE(%)	3.08	3.93	1.61	1.96	3.49	3.04
AUKF [54]	MAE(%)	0.785	1.02	1.12	-	-	**0.98**
	RMSE(%)	1.27	1.25	1.32	-	-	**1.28**
EIS + GPR [55]	MAE(%)	1.34	3.82	8.68	2.72	2.75	3.86
	RMSE(%)	1.46	4.41	9.96	3.45	3.55	4.57
Method in [24]	MAE(%)	2.74	**0.58**	1.91	2.89	4.94	2.61
	RMSE(%)	2.27	**0.45**	1.70	2.17	3.95	2.11
CNN+BiLSTM (ours)	MAE(%)	**0.76**	1.37	**0.535**	**1.11**	**1.69**	1.09
	RMSE(%)	**1.07**	1.81	**0.689**	**1.46**	**2.18**	1.44

The best estimation results are indicated in bold.

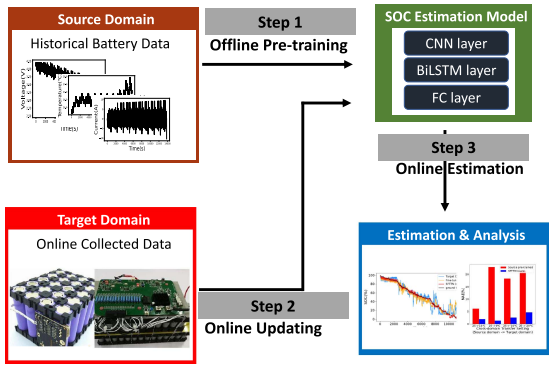

Fig. 4. Application of cross-domain SOC estimation at real-world conditions (online testing).

online testing, current piece is used for training and the next piece is used for testing, just like that in practical applications. During online testing, the model keeps updating itself. In short, the model is dynamic rather than static.

V. RESULTS AND ANALYSIS

A. Source-Free Cross-Domain Transfer Experiment on Semisupervised Scenario

We compare the performance of our method with other methods on traditional SOC estimation. We choose several representative methods as the baseline methods, including GRU [14], LSTM [10], AUKF [54], EIS [55], and a method in [24]. The results are shown in Table V. As shown in the results, the errors of most cases are very close. AUKF achieves the smallest average errors with 0.98% and 1.28%. Our method achieves errors with 1.09% and 1.44%, which are very close to the best results. Specifically, in the case of 0 °C, our methods achieves the best results with errors at 0.535% and 0.689%. The results verify that our method can achieve the state-of-the-art performance on traditional SOC estimation task at fixed ambient temperatures.

B. Source-Free Cross-Domain Estimation Results Under Unsupervised Scenario

Up until now, there is no related work for source-free SOC estimation under unsupervised scenario. To evaluate the transfer performance of proposed SFTTN under unsupervised scenario, one drive cycle is selected as the testing data while the remaining target data are unlabeled.

The SOC estimation errors of test drive cycle from one ambient temperature to other temperatures are shown in Fig. 5. Considering there are no related methods for source-free transfer, the model pretrained in the source domain is used for comparison. As shown in the figures, without the guidance of labeled source data and target data, the estimation errors of pretrained model are rather large, which mostly fall into 10%–20% while SFTTN achieves significant improvement over 70%, indicating the importance of proposing effective cross-domain transfer SOC estimation under source-free unsupervised scenario. Compared with source pretrained model, the errors using SFTTN method are greatly reduced. Nevertheless, the estimation performance under unsupervised setting is not as good as that under semisupervised condition. The problem is caused by the limitation of the pseudolabels and unaligned feature representation. The quality of the pseudolabels depends on the pretrained estimators and data correlation between source and target domains. It is worth noting that on some cases the errors of source pretrained model can be small. For instance, the RMSE in the case of 0 °C → 25 °C is 27.4% while that of 0 °C → −10 °C achieves the smallest within 2.8%. It is because that the difference of the domain shift causes different errors. Tested data between the ambient temperatures of 0 °C and −10 °C shares slighter domain shift than that of data between 0 °C and 25 °C.

The total results from one temperature to others are shown in Table VI. Similar with semisupervised experiments, MMD metric-based model and adversarial training model with source data are included as the baseline models for comparison to show the superiority of our method. The proposed SFTTN

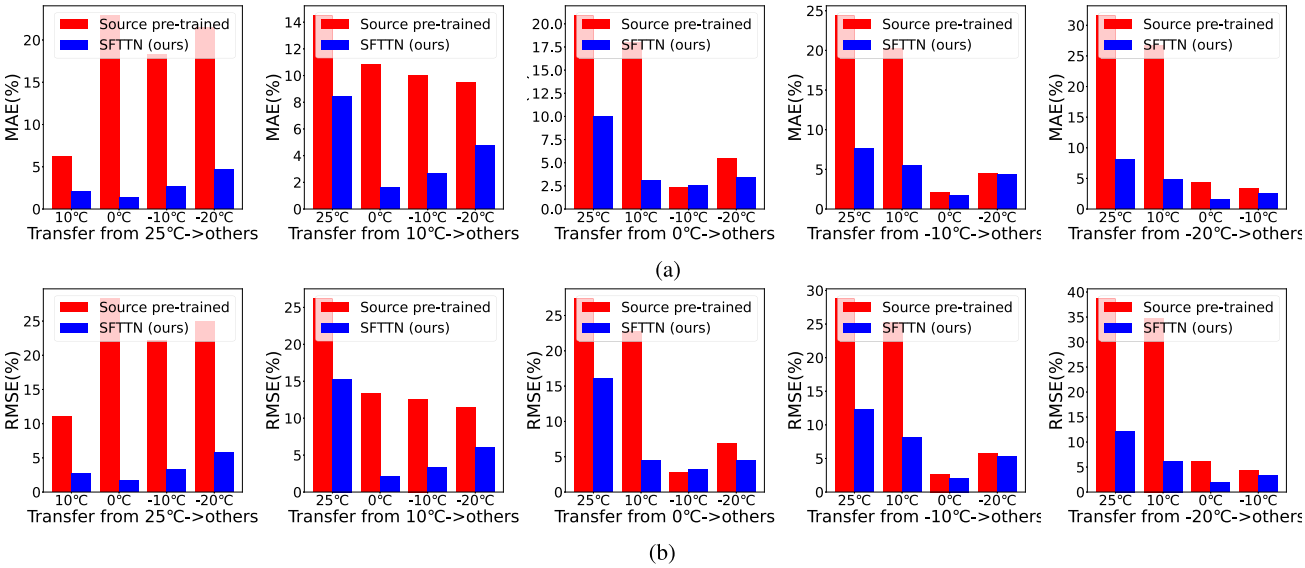

Fig. 5. Cross-domain SOC estimation results of different groups of transfer settings on (a) MAE and (b) RMSE under unsupervised scenario.

TABLE VI
SOURCE-FREE CROSS-DOMAIN TRANSFER ESTIMATION RESULTS FROM ONE TEMPERATURE TO OTHERS UNDER UNSUPERVISED SCENARIO

Method	Metrics	25 °C $\rightarrow \mathcal{R}$	10 °C $\rightarrow \mathcal{R}$	0 °C $\rightarrow \mathcal{R}$	−10 °C $\rightarrow \mathcal{R}$	−20 °C $\rightarrow \mathcal{R}$	Avg.
Source pre-trained	MAE(%)	17.2	11.2	11.7	12.8	16.5	13.9
	RMSE(%)	21.6	15.9	14.9	15.6	21.0	17.8
MMD ⋆ [18]	MAE(%)	17.1	10.3	11.6	12.6	16.4	13.6
	RMSE(%)	21.6	14.9	14.9	15.8	20.9	17.6
Adversarial training ⋆ [20]	MAE(%)	13.1	8.45	13.9	16.1	15.5	13.4
	RMSE(%)	16.6	10.48	17.3	19.5	18.6	16.5
SFTTN † (ours)	MAE(%)	**2.67**	**4.36**	**4.74**	**4.81**	**4.26**	**4.17**
	RMSE(%)	**3.38**	**6.70**	**7.06**	**6.96**	**5.91**	**6.00**

⋆ denotes transfer with both source and target data.
† denotes source-free transfer.
$\mathcal{R}$ denotes other four ambient temperatures.
The best estimation results are indicated in bold.

outperforms other baseline models in all the transfer cases. Although adversarial training and MMD methods make some improvement compared with source pretrained model in some cases, the MAEs and RMSEs are still large and they require coexistence of source and target data.

C. Source-Free Cross-Domain Transfer Results From Panasonic 18650PF to LG 18650HG2 and A123

Similar with abovementioned experiments, we take source pretrained model, MMD method and adversarial training method as the baseline methods while only SFTTN can work without source data. The results of cross-battery experiments are shown in Tables VII and VIII. It is worth noting that the errors of pretrained model are even smaller than those of MMD and adversarial training. The model without transfer stage even outperforms methods, which integrate transfer approaches, which seems contrary to our cognition. The abnormal results indicate the generalization problem that often arises in MMD and adversarial training. Even the domain discrepancy is minimized, the model can still be far from the ideal one. Despite all the risks, our method achieves best performance for all the tested cases with errors no larger than 3.6%.

The results indicate that SFTTN can quickly and effectively retrain an SOC estimator for one battery from a pretrained model of another battery. Normally, if we want to train an SOC estimator for one battery type, we need to collect massive reliable data in the laboratory (data-driven methods) or to analysis the chemical properties and construct model for the battery (model-based methods). Other transfer methods need training data from both batteries while our method only needs real-time running data from one battery. However, with our method, the training cost can be greatly reduced.

D. Source-Free Online Testing at Changing Temperatures

We take source pretrained model, MMD method and adversarial training method as the baseline methods. The results of online testing at changing temperatures are shown in Table IX. As shown in the table, the errors tested at changing temperatures are

TABLE VII
SOURCE-FREE CROSS-DOMAIN TRANSFER RESULTS FROM PANASONIC 18650PF TO LG 18650LG2

Method	Metrics	25 °C	10 °C	0 °C	−10 °C	−20 °C	Avg.
Source pretrained	MAE(%)	12.3	7.49	9.97	11.7	14.5	11.2
	RMSE(%)	16.3	9.37	12.7	14.3	19.1	14.4
MMD ⋆ [18]	MAE(%)	19.7	8.39	18.5	18.8	12.6	15.6
	RMSE(%)	24.9	10.8	22.7	23.9	15.3	19.5
Adversarial training ⋆ [20]	MAE(%)	28.2	13.3	24.2	22.1	21.2	21.8
	RMSE(%)	34	16.3	29.0	26.8	24.3	26.1
SFTTN † (ours)	MAE(%)	**2.24**	**3.05**	**1.96**	**1.82**	**1.64**	**2.14**
	RMSE(%)	**3.40**	**3.27**	**3.16**	**2.80**	**2.60**	**3.05**

⋆ denotes transfer with both source and target data.
† denotes source-free transfer.
The best estimation results are indicated in bold.

TABLE VIII
SOURCE-FREE CROSS-DOMAIN TRANSFER RESULTS FROM PANASONIC 18650PF TO A123

Method	Metrics	50 °C	40 °C	30 °C	25 °C	20 °C	10 °C	0 °C	−10 °C	Avg.
Source pre-trained	MAE(%)	13.5	8.74	9.79	13.5	15.2	8.42	11.2	19.2	12.4
	RMSE(%)	15.1	11.0	11.2	14.9	17.8	10.2	11.5	21.5	14.2
MMD ⋆ [18]	MAE(%)	19.0	12.5	13.5	17.9	20.4	12.7	16.6	22.1	16.8
	RMSE(%)	21.7	15.3	15.3	19.4	23.6	13.5	19.9	26.5	19.4
Adversarial training ⋆ [20]	MAE(%)	21.7	13.4	15.8	22.3	18.9	14.2	18.8	20.7	18.2
	RMSE(%)	24.2	16.7	19.2	25.6	20.7	16.2	20.4	23.9	20.9
SFTTN † (ours)	MAE(%)	**2.16**	**1.88**	**2.97**	**2.06**	**2.49**	**1.98**	**2.00**	**1.98**	**2.19**
	RMSE(%)	**2.74**	**2.15**	**3.58**	**2.27**	**2.78**	**2.06**	**2.39**	**3.54**	**2.69**

⋆ denotes transfer with both source and target data.
† denotes source-free transfer.
The best estimation results are indicated in bold.

TABLE IX
SOURCE-FREE ONLINE TESTING RESULTS AT CHANGING TEMPERATURES

Method	Metrics	25 °C → $\mathcal{C}$	10 °C → $\mathcal{C}$	0 °C → $\mathcal{C}$	−10 °C → $\mathcal{C}$	−20 °C → $\mathcal{C}$	Avg.
Source pre-trained	MAE(%)	35.9	44.5	11.8	13.6	8.68	22.9
	RMSE(%)	46.2	53.5	21.0	20.9	12.8	30.9
MMD ⋆ [18]	MAE(%)	9.56	27.7	7.19	8.86	7.52	12.2
	RMSE(%)	12.2	37	12.8	14.3	12.8	17.8
Adversarial training ⋆ [20]	MAE(%)	21.9	9.37	8.09	9.31	6.78	11.1
	RMSE(%)	30.6	12.3	12.5	11.9	9.80	15.4
SFTTN † (ours)	MAE(%)	**4.36**	**6.73**	**6.51**	**6.54**	**6.41**	**6.11**
	RMSE(%)	**6.46**	**9.32**	**7.93**	**10.0**	**9.11**	**8.56**

⋆ denotes transfer with both source and target data.
† denotes source-free transfer.
$\mathcal{C}$ denotes changing temperature from −20 °C to 20 °C.
The best estimation results are indicated in bold.

a little larger than those tested at fixed ambient temperatures. The average errors of our method are 6.11% and 8.56%. Considering our method can work with only unlabeled target data, which can largely reduce time cost, the results are within acceptable range.

Our method remains effective at changing temperatures. The model can quickly update itself with real-time collected data. By contrast, other transfer models are static, which can cause large errors at changing environment. The results indicate the superiority of SFTTN in online applications.

E. Discussion

There are some interesting issues worth discussing.

First issue is the relationship between SOC and SOH. SOH refers to the percentage of the maximum available capacity to

the rated capacity, representing the health degradation of the battery. Interestingly and expectedly, SFTTN can also be applied in cross-domain SOH estimation. From another perspective, battery of different SOH can be seen as different types of batteries. Thus, SFTTN has great potential in cross-domain SOH estimation.

Second issue is about the impact of significant factors in real-time applications that can affect the performance of model, such as measurement uncertainty, wrong initialization, and battery aging. For other transfer methods, abovementioned factors can cause performance degradation of the estimation model. However, SFTTN can deal with these problems perfectly by transferring them into source-free cross-domain transfer issue. First, abovementioned discussion has shown that SFTTN can transfer models between batteries with different aging degrees. Second, suppose a model is trained with inaccurate measured parameters and wrongly initialized model weights. The pretrained model can perfectly estimate previous wrong data, but cannot predict accurate values for following correctly measured parameters, because the training data and testing data are differently distributed. Since there is a gap between the data distributions, SFTTN can be utilized to transfer the wrongly trained model and adapt it to following correct data.

VI. Conclusion

In this article, we propose a novel transfer framework named SFTTN to tackle the challenges of distribution discrepancy and data limitation in cross-domain SOC estimation under different ambient temperatures. We propose a more difficult yet more practical source-free scenario where source data are absent and only source pretrained model and target data can be available. For better domain-invariant feature representations learning, we employ 2D-CNN layers and BiLSTM layers as the feature extractor. By pretraining differently initiated estimators in source domain and minimizing the estimation discrepancy of target data gained by the estimators, the domain shift is alleviated. Abundant comparative experiments demonstrate that SFTTN is able to achieve comparable or better performance comparing with other methods on both traditional SOC estimation at fixed temperatures and cross-domain SOC estimation at varying ambient temperatures. There are still drawbacks of SFTTN. For example, just like other data-driven methods, the performance of SFTTN depends on accurate measurement of parameters. Relevant discussion has shown that SFTTN can also be utilized for cross-domain SOH estimation.

References

[1] H. Weiss, T. Winkler, and H. Ziegerhofer, "Large lithium-ion battery-powered electric vehicles—from idea to reality," in *Proc. ELEKTRO*, 2018, pp. 1–5.

[2] M. U. Ali, A. Zafar, S. H. Nengroo, S. Hussain, M. Junaid Alvi, and H.-J. Kim, "Towards a smarter battery management system for electric vehicle applications: A critical review of lithium-ion battery state of charge estimation," *Energies*, vol. 12, no. 3, 2019, Art. no. 446.

[3] M. O. Qays, Y. Buswig, M. L. Hossain, and A. Abu-Siada, "Recent progress and future trends on the state of charge estimation methods to improve battery-storage efficiency: A review," *CSEE J. Power Energy Syst.*, vol. 8, no. 1, pp. 105–114, 2020.

[4] R. Xiong, J. Cao, Q. Yu, H. He, and F. Sun, "Critical review on the battery state of charge estimation methods for electric vehicles," *IEEE Access*, vol. 6, pp. 1832–1843, 2017.

[5] K. Movassagh, A. Raihan, B. Balasingam, and K. Pattipati, "A critical look at coulomb counting approach for state of charge estimation in batteries," *Energies*, vol. 14, no. 14, 2021, Art. no. 4074.

[6] X. Bian, Z. Wei, J. He, F. Yan, and L. Liu, "A two-step parameter optimization method for low-order model-based state-of-charge estimation," *IEEE Trans. Transport. Electrific.*, vol. 7, no. 2, pp. 399–409, Jun. 2021.

[7] A. Bavand, S. A. Khajehoddin, M. Ardakani, and A. Tabesh, "Online estimations of li-ion battery SOC and SOH applicable to partial charge/discharge," *IEEE Trans. Transport. Electrific.*, vol. 8, no. 3, pp. 3673–3685, Sep. 2022.

[8] Y. Fan, J. Wu, Z. Chen, H. Wu, J. Huang, and B. Liu, "Data-driven state-of-charge estimation of lithium-ion batteries," in *Proc. 8th Int. Conf. Power Electron. Syst. Appl.*, 2020, pp. 1–5.

[9] Z. Du, L. Zuo, J. Li, Y. Liu, and H. T. Shen, "Data-driven estimation of remaining useful lifetime and state of charge for lithium-ion battery," *IEEE Trans. Transport. Electrific.*, vol. 8, no. 1, pp. 356–367, Mar. 2022.

[10] C. Li, F. Xiao, Y. Fan, G. Yang, and W. Zhang, "A recurrent neural network with long short-term memory for state of charge estimation of lithium-ion batteries," in *Proc. IEEE 8th Joint Int. Inf. Technol. Artif. Intell. Conf.*, 2019, pp. 1712–1716.

[11] X. Shu, G. Li, Y. Zhang, S. Shen, Z. Chen, and Y. Liu, "Stage of charge estimation of lithium-ion battery packs based on improved cubature kalman filter with long short-term memory model," *IEEE Trans. Transport. Electrific.*, vol. 7, no. 3, pp. 1271–1284, Sep. 2020.

[12] H. S. Bhattacharyya, A. Yadav, A. B. Choudhury, and C. K. Chanda, "Convolution neural network-based soc estimation of li-ion battery in ev applications," in *Proc. 5th Int. Conf. Elect., Electron., Commun., Comput. Technol. Optim. Techn.*, 2021, pp. 587–592.

[13] A. Herle, J. Channegowda, and D. Prabhu, "A temporal convolution network approach to state-of-charge estimation in li-ion batteries," in *Proc. IEEE 17th India Council Int. Conf.*, 2020, pp. 1–6.

[14] G. Javid, M. Basset, and D. O. Abdeslam, "Adaptive online gated recurrent unit for lithium-ion battery SOC estimation," in *Proc. 46th Annu. Conf. IEEE Ind. Electron. Soc.*, 2020, pp. 3583–3587.

[15] J. Li, K. Lu, Z. Huang, L. Zhu, and H. T. Shen, "Transfer independently together: A generalized framework for domain adaptation," *IEEE Trans. Cybern.*, vol. 49, no. 6, pp. 2144–2155, Jun. 2019.

[16] J. Li, M. Jing, H. Su, K. Lu, L. Zhu, and H. T. Shen, "Faster domain adaptation networks," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 12, pp. 5770–5783, Dec. 2022.

[17] J. Li, K. Lu, Z. Huang, L. Zhu, and H. T. Shen, "Heterogeneous domain adaptation through progressive alignment," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 5, pp. 1381–1391, May 2019.

[18] K. M. Borgwardt, A. Gretton, M. J. Rasch, H.-P. Kriegel, B. Schölkopf, and A. J. Smola, "Integrating structured biological data by Kernel maximum mean discrepancy," *Bioinformatics*, vol. 22, no. 14, pp. e49–e57, 2006.

[19] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3723–3732.

[20] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7167–7176.

[21] J. Li, E. Chen, Z. Ding, L. Zhu, K. Lu, and H. T. Shen, "Maximum density divergence for domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 3918–3930, Nov. 2021.

[22] J. Li, Z. Du, L. Zhu, Z. Ding, K. Lu, and H. T. Shen, "Divergence-agnostic unsupervised domain adaptation by adversarial attacks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8196–8211, Nov. 2022.

[23] L. Shen, J. Li, J. Liu, L. Zhu, and H. T. Shen, "Temperature adaptive transfer network for cross-domain state-of-charge estimation of li-ion batteries," *IEEE Trans. Power Electron.*, vol. 38, no. 3, pp. 3857–3869, Mar. 2023.

[24] Y. Qin, S. Adams, and C. Yuen, "Transfer learning-based state of charge estimation for lithium-ion battery at varying ambient temperatures," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7304–7315, Nov. 2021.

[25] K. S. Ng, C.-S. Moo, Y.-P. Chen, and Y.-C. Hsieh, "Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries," *Appl. Energy*, vol. 86, no. 9, pp. 1506–1511, 2009.

[26] P. Ningrum, N. A. Windarko, and S. Suhariningsih, "Estimation of state of charge (SoC) using modified coulomb counting method with open circuit compensation for battery management system (BMS)," *J. Adv. Res. Elect. Eng.*, vol. 5, no. 1, pp. 15–20, 2021.

[27] Z. Ren, C. Du, Z. Wu, J. Shao, and W. Deng, "A comparative study of the influence of different open circuit voltage tests on model-based state of charge estimation for lithium-ion batteries," *Int. J. Energy Res.*, vol. 45, no. 9, pp. 13692–13711, 2021.

[28] A. Gismero, E. Schaltz, and D.-I. Stroe, "Recursive state of charge and state of health estimation method for lithium-ion batteries based on coulomb counting and open circuit voltage," *Energies*, vol. 13, no. 7, 2020, Art. no. 1811.

[29] C. Unterrieder, M. Lunglmayr, S. Marsili, and M. Huemer, "Battery state-of-charge estimation prototype using EMF voltage prediction," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2014, pp. 622–625.

[30] W.-Y. Kim, P.-Y. Lee, J. Kim, and K.-S. Kim, "State of charge and equivalent internal resistance estimation for a multi-cell application based on cell-difference-model," in *Proc. IEEE Energy Convers. Congr. Expo.*, 2019, pp. 2664–2668.

[31] J. Zhang et al., "Variable-order equivalent circuit modeling and state of charge estimation of lithium-ion battery based on electrochemical impedance spectroscopy," *Energies*, vol. 14, no. 3, 2021, Art. no. 769.

[32] P. Shrivastava, T. K. Soon, M. Y. I. B. Idris, and S. Mekhilef, "Overview of model-based online state-of-charge estimation using Kalman filter family for lithium-ion batteries," *Renewable Sustain. Energy Rev.*, vol. 113, 2019, Art. no. 109233.

[33] Y. Tian, R. Lai, X. Li, L. Xiang, and J. Tian, "A combined method for state-of-charge estimation for lithium-ion batteries using a long short-term memory network and an adaptive cubature Kalman filter," *Appl. Energy*, vol. 265, 2020, Art. no. 114789.

[34] F. Naseri, E. Schaltz, D.-I. Stroe, A. Gismero, and E. Farjah, "An enhanced equivalent circuit model with real-time parameter identification for battery state-of-charge estimation," *IEEE Trans. Ind. Electron.*, vol. 69, no. 4, pp. 3743–3751, Apr. 2022.

[35] G. Fan, X. Li, and R. Zhang, "Global sensitivity analysis on temperature-dependent parameters of a reduced-order electrochemical model and robust state-of-charge estimation at different temperatures," *Energy*, vol. 223, 2021, Art. no. 120024.

[36] X. Lin, Y. Tang, J. Ren, and Y. Wei, "State of charge estimation with the adaptive unscented Kalman filter based on an accurate equivalent circuit model," *J. Energy Storage*, vol. 41, 2021, Art. no. 102840.

[37] E. Galiounas, T. G. Tranter, R. E. Owen, J. B. Robinson, P. R. Shearing, and D. J. Brett, "Battery state-of-charge estimation using machine learning analysis of ultrasonic signatures," *Energy AI*, vol. 10, 2022, Art. no. 100188.

[38] M. A. Hannan, D. N. How, M. B. Mansor, M. S. H. Lipu, P. J. Ker, and K. M. Muttaqi, "State-of-charge estimation of li-ion battery using gated recurrent unit with one-cycle learning rate policy," *IEEE Trans. Ind. Appl.*, vol. 57, no. 3, pp. 2964–2971, May 2021.

[39] J. Li, W. Ziehm, J. Kimball, R. Landers, and J. Park, "Physical-based training data collection approach for data-driven lithium-ion battery state-of-charge prediction," *Energy AI*, vol. 5, 2021, Art. no. 100094.

[40] M. A. Hannan et al., "SOC estimation of li-ion batteries with learning rate-optimized deep fully convolutional network," *IEEE Trans. Power Electron.*, vol. 36, no. 7, pp. 7349–7353, Jul. 2021.

[41] S. Shen, M. Sadoughi, M. Li, Z. Wang, and C. Hu, "Deep convolutional neural networks with ensemble learning and transfer learning for capacity estimation of lithium-ion batteries," *Appl. Energy*, vol. 260, 2020, Art. no. 114296.

[42] C. Vidal, P. Kollmeyer, E. Chemali, and A. Emadi, "Li-ion battery state of charge estimation using long short-term memory recurrent neural network with transfer learning," in *Proc. IEEE Transp. Electrific. Conf. Expo*, 2019, pp. 1–6.

[43] Y. Liu, J. Li, G. Zhang, B. Hua, and N. Xiong, "State of charge estimation of lithium-ion batteries based on temporal convolutional network and transfer learning," *IEEE Access*, vol. 9, pp. 34177–34187, 2021.

[44] C. Bian, S. Yang, and Q. Miao, "Cross-domain state-of-charge estimation of li-ion batteries based on deep transfer neural network with multiscale distribution adaptation," *IEEE Trans. Transport. Electrific.*, vol. 7, no. 3, pp. 1260–1270, Sep. 2021.

[45] W. M. Kouw and M. Loog, "A review of domain adaptation without target labels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 3, pp. 766–785, 2019.

[46] P. Baldi and P. J. Sadowski, "Understanding dropout," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2814–2822.

[47] J. Si, S. L. Harris, and E. Yfantis, "A dynamic ReLU on neural network," in *Proc. IEEE 13th Dallas Circuits Syst. Conf.*, 2018, pp. 1–6.

[48] J. A. Kumar and S. Abirami, "Ensemble application of bidirectional LSTM and GRU for aspect category detection with imbalanced data," *Neural Comput. Appl.*, vol. 33, no. 21, pp. 14603–14621, 2021.

[49] P. Kollmeyer, "Panasonic 18650PF li-ion battery data," *Mendeley Data*, vol. 1, no. 2018, 2018, doi: 10.17632/wykht8y7tg.1.

[50] M. Naguib, P. Kollmeyer, and M. Skells, "Lg 18650hg2 li-ion battery data," *Mendeley Data*, vol. 1, 2020, doi: 10.17632/b5mj79w5w9.2.

[51] Center for advanced life cycle engineering battery research group, 2012. [Online]. Available: https://web.calce.umd.edu/batteries/data.htm

[52] I. Oyewole, A. Chehade, and Y. Kim, "A controllable deep transfer learning network with multiple domain adaptation for battery state-of-charge estimation," *Appl. Energy*, vol. 312, 2022, Art. no. 118726.

[53] P. Kollmeyer, C. Vidal, M. Naguib, and M. Skells, "LG 18650HG2 li-ion battery data and example deep neural network XEV SOC estimator script," *Mendeley Data*, vol. 3, 2020, doi: 10.17632/cp3473x7xv.3.

[54] F. Sun, X. Hu, Y. Zou, and S. Li, "Adaptive unscented Kalman filtering for state of charge estimation of a lithium-ion battery for electric vehicles," *Energy*, vol. 36, no. 5, pp. 3531–3540, 2011.

[55] I. Babaeiyazdi, A. Rezaei-Zare, and S. Shokrzadeh, "State of charge prediction of EV li-ion batteries using EIS: A machine learning approach," *Energy*, vol. 223, 2021, Art. no. 120116.