



Short-term aggregate electric vehicle charging load forecasting in diverse conditions with minimal data using transfer and meta-learning

Shashank Narayana Gowda¹ · Keshav Nath² · Chen Zhang¹ · Rohan Gowda³ · Rajit Gadh¹

Received: 16 October 2023 / Accepted: 12 April 2024
© The Author(s) 2024

Abstract

The proliferation of electric vehicles (EVs) necessitates accurate EV charging load forecasting for demand-side management and electric-grid planning. Conventional machine learning-based load forecasting methods like long short-term memory (LSTM) neural networks rely on large amounts of historical data, which can be resource-intensive and time-consuming to collect. In this study, we employ Transfer Learning (TL) and Model-Agnostic Meta-Learning (MAML) for short term EV charging load forecasting. These methods involve pre-training a base model on a larger comprehensive EV charging dataset followed by fine-tuning using a few days' worth of EV charging data in our target location. We find that the performance of both the TL and MAML models outperform traditional LSTM models and other classic machine learning models in the context of forecast accuracy when working in three different settings with limited data, load variance, and diverse geographical locations. The error metrics from TL and MAML are up to 24% and 61% lower than deep learning and classic machine learning models respectively.

Keywords Forecasting · Transfer learning · Meta-learning · Machine learning · Electric vehicles · Electric load

1 Introduction

The increasing incorporation of renewable energy sources (RES) and electric vehicles (EVs) introduces complexities in achieving coordination between electricity supply and demand due to their inherent uncertainties. Consequently, accurate forecasting of electricity generation and loads assumes pivotal importance in sustaining a stable smart grid, enabling the precise matching of supply and demand across various time scales [1]. On the supply side, wind and solar energy sources exhibit intermittency, relying on external weather variables outside human control.

Extended author information available on the last page of the article

Simultaneously, the forecasting of EV charging loads becomes a necessity for long-term infrastructure planning and medium-to-short-term demand-side management. These efforts mitigate peak loads, reduce costs, and minimize the curtailment of RES [2]. Furthermore, harnessing demand-side flexibility has the potential to yield cost savings in both planning and operational aspects [3].

Traditionally, EV charging load forecasting is treated as a time series problem and involves the application of statistical and machine learning approaches [2]. Among statistical methods, autoregressive integrated moving average (ARIMA) and exponential smoothing (ETS) are popular due to their simplicity, robustness, efficiency, and accuracy [4]. Machine learning techniques, such as Long Short-Term Memory (LSTM) networks, a type of recurrent neural network (RNN) that leverages both current and stored information in memory, are widely used for time series forecasting, at times surpassing the performance of statistical methods [4]. To enhance the predictive accuracy further, the authors in [2] adopted ensemble learning techniques that combine the strengths of multiple machine learning models, effectively synergizing their capabilities to enhance the precision of EV charging load forecasting.

However, neural network (NN)-based forecasting methods exhibit suboptimal performance in scenarios characterized by limited training data. This limitation often arises when data collection proves challenging, expensive, subject to strict regulations, or when dealing with newly established setups [5]. One approach to address this challenge is transfer learning (TL), which leverages pre-trained forecasting models developed using extensive and diverse datasets as initial models. These pre-trained models can then undergo fine-tuning using a limited dataset specific to the problem at hand [6]. [7] proposed the use of transfer learning using deep generative models for cold-start forecasting of residential EV charging behavior. Generative models were used to augment the small samples in the target dataset before fine-tuning. A hybrid LSTM transfer learning model was used in [8] to predict EV charging demand and network voltage profiles with limited data within a single city.

Another promising algorithm for mitigating the data scarcity issue is model-agnostic meta-learning (MAML). MAML is capable of training on a variety of learning tasks, enabling it to adeptly tackle new learning tasks with a limited number of available training samples [9]. [10] introduced a transferable MAML model tailored for short-term load forecasting in individual Australian households. [5] harnessed transfer and meta-learning techniques to predict personalized residential loads in Korea and substation loads in Portugal but ignored load patterns during weekends and holidays. Both [10] and [5] performed univariate load forecasting where the only feature used to predict load was historical load. [11] employed a meta-learning model to forecast load profiles at the transmission level, utilizing open-access data from Elia, a Belgian transmission system operator. In [11], the forecast was performed to find the future values in the same source dataset, meanwhile [10] and [5] forecasted loads of similar scale in the same country.

In this study, we employ TL and MAML techniques for short-term aggregate EV charging load forecasting. Our investigation spans two distinct geographical areas: the California Institute of Technology (CalTech) campus [12] and the Trondheim region in Norway [13]. We accomplish this by leveraging a comprehensive EV charging load dataset sourced from Boulder, Colorado for pre-training [2]. These

datasets encompass varying scales, including campus-level, housing initiative-level, and city-level data. Moreover, they originate from different locations, namely Pasadena, CA, USA; Trondheim, Norway; and Boulder, CO, USA. Additionally, we work with aggregated EV charging data in both shared charger and private charger settings. To assess the effectiveness of our approach, we evaluate the performance of different TL, MAML, and traditional LSTM network variants on these diverse datasets at the time scales of 10–20 days. The models only get to be fine-tuned on 10 or 20 days' worth of data to predict the next 5–10 days on each of the investigated datasets. It is worth noting that, to the best of our knowledge, our study represents the first of its kind in this context. We compare our results to traditional LSTM deep learning networks and to other classic machine learning models.

The key contributions of our paper can be summarized as follows:

- We use TL and MAML training to create short-term load forecasting models;
- Additional temporal features such as seasons, weekends and holidays are incorporated to create richer data;
- Utilize adjacent time windows as query and support sets for MAML;
- Models utilize small amounts (10 or 20 days) of fine-tuning data;
- We ensure the robustness of models in making predictions across varying scales, geographies and demographics by utilizing multiple datasets;

The subsequent sections of this paper are structured as follows:

Methods: This section provides a comprehensive overview of the diverse models employed for short-term EV charging load forecasting. Additionally, it outlines the particulars of the datasets, and delves into the data pre-processing procedures, model architectures, hyperparameters, and details concerning the experimental configuration.

Results: In this section, we present the outcomes of our study and highlight the noteworthy observations derived from the experiments.

Discussion: The concluding section summarizes our findings and provides prospective directions for future research endeavors.

2 Methods

2.1 Datasets

We utilized three distinct publicly available datasets:

2.1.1 Colorado dataset

The dataset from Colorado [14] used in the work of [2], encompasses records of individual EV charging transactions recorded at city-owned shared charging stations located in Boulder, Colorado, USA, covering a time frame extending from January

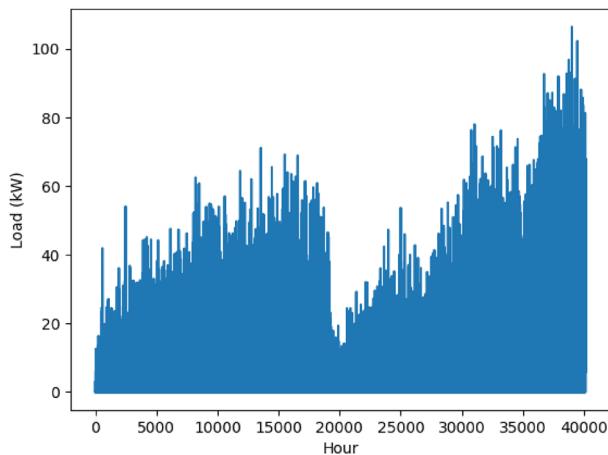


Fig. 1 Aggregate EV charging load distribution in the Colorado dataset

1, 2018, to July 31, 2022. The aggregated EV charging load data is visually represented in Fig. 1.

2.1.2 ACN dataset

The dataset from the Adaptive Charging Network (ACN), as detailed in [12], comprises records of EV charging transactions logged at the CalTech and JPL campuses. Our particular focus centers on the CalTech data, primarily because it exhibits lower EV penetration and a more sparse nature, as visually depicted in Fig. 2. This dataset encompasses a timeframe ranging from April 4, 2019, to September 16, 2021.

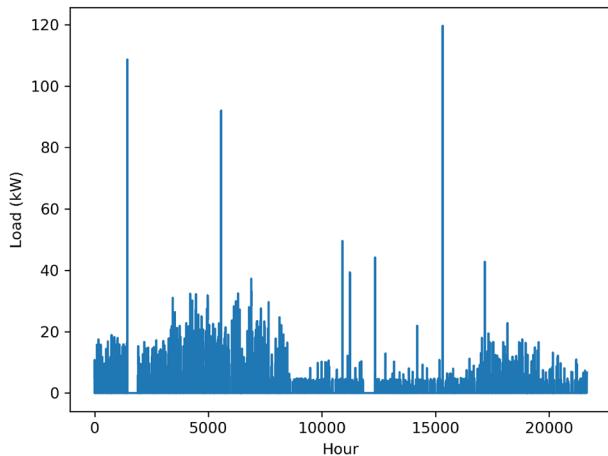


Fig. 2 Aggregate EV charging load distribution in the ACN dataset

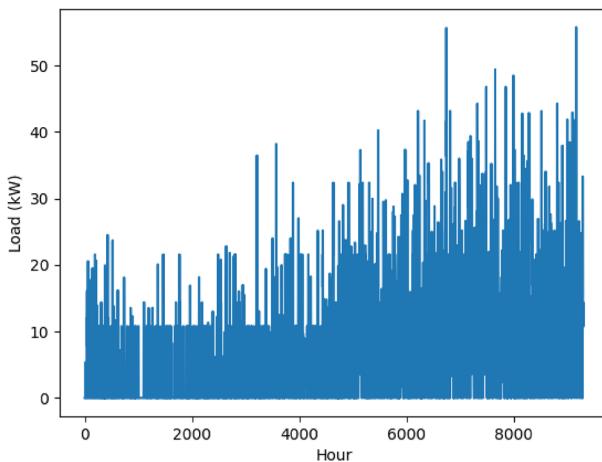


Fig. 3 Aggregate EV charging load distribution from shared chargers in the Norway dataset

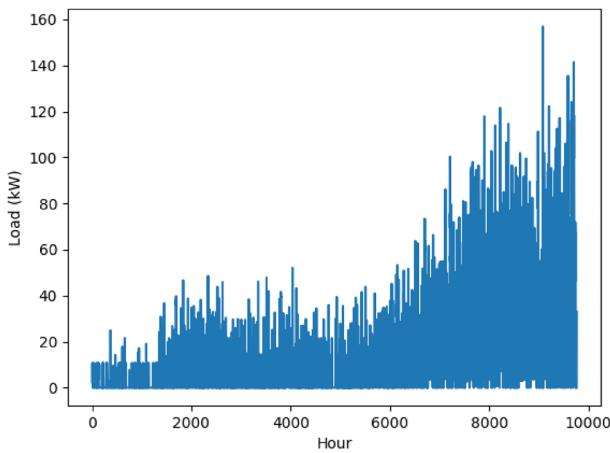


Fig. 4 Aggregate EV charging load distribution from private chargers in the Norway dataset

2.1.3 Norway dataset

The Norway dataset [15] used in [13] contains hourly charging load data from charging stations within a residential block in the Trondheim region of Norway. It consists of two parts—aggregate shared charger data and aggregate private charger data visualized in Figs. 3 and 4 respectively. Both these datasets exhibit an irregular and challenging data distribution because chargers were added in each of these settings over time. Additionally, we can see that the magnitude of the aggregated private charger data is about three times the aggregated shared charger data. The aggregate shared charger data was collected from January 10, 2019 until January

31, 2020 while the aggregate private charger was collected from December 21, 2018 until January 31, 2020.

2.2 Data pre-processing

We transformed both the Colorado and ACN datasets, originally containing charging transactions, into an hourly load format. Following this, all four datasets (Colorado, ACN, Norway-shared, and Norway-private) underwent an extensive pre-processing procedure, incorporating a comprehensive set of 10 features encompassing hour, quarter-of-day, day, day-of-week, week-of-year, month, quarter-of-year, season, and load (kW).

The data was then structured into input–output pairs, with the input comprising a sequence of 24 consecutive hours as features, with the load of the 25th hour serving as the target variable. $D = (x_t, y_t), t \in T$ with $x_t \in \mathbb{R}^{10 \times 24}$ and $y_t \in \mathbb{R}$. Given the substantial size and density of the Colorado dataset, we opted to employ it for the pre-training phase of both the MAML and TL models. Subsequently, these pre-trained models underwent a fine-tuning process utilizing subsets of the ACN and both the Norway datasets. The fine-tuning was conducted using 10-day and 20-day subsets, requiring the models to make hourly load forecasts for the subsequent 10 and 20 days, respectively. This approach compelled the models to adapt to very limited historical data when making predictions about future trends.

2.3 Models

2.3.1 Neural networks

In the context of forecasting, which falls under supervised learning, a neural network (NN) can be represented as $z = f_\theta(x)$, where x represents the input, z denotes the output, and $\theta \in \mathbb{R}^n$ encompasses parameters distributed across multiple layers in the network. Typically, θ comprises weights W that signify the strength between neuronal connections in adjacent layers, as well as biases b within each layer.

These parameters are subject to optimization for a specific task $\mathcal{T} = (D, L)$, minimizing a loss function L over a designated subset of data $D = (x_t, y_t) | t \in T$, employed for training $D_{train} = (x_t, y_t) | t \in T_{train}$. x_t represents the input data and y_t stands for the corresponding labels. T encompasses the total number of samples within the dataset, and T_{train} denotes the portion of samples reserved for training. The loss function L leverages a distance metric $d()$ to capture the dissimilarity between the model outputs z and the ground truth labels y .

$$L(\theta) = \sum_{t \in T} d(z_t, y_t) = \sum_{t \in T} d(f_\theta(x_t), y_t) \quad (1)$$

For regression problems, L1 loss and mean squared error (MSE) loss are the prevalent loss functions as shown in Eqs. (2) and (3). The MSE loss is used when we want to penalize larger errors more heavily, while the L1 loss is used in the presence of outliers in the data and when the distribution of errors is not necessarily normal.

$$L_{L1}(\theta) = \frac{1}{T} \sum_{t \in T} |f_\theta(x_t) - y_t| \quad (2)$$

$$L_{MSE}(\theta) = \frac{1}{T} \sum_{t \in T} (f_\theta(x_t) - y_t)^2 \quad (3)$$

The optimal parameters θ^* are achieved through the iterative application of the back-propagation algorithm, as depicted in Eq. (4), that minimizes the loss function across the training data. Here, α represents the learning rate, and $\nabla_\theta L()$ denotes the gradient of the loss function.

$$\theta \leftarrow \theta - \alpha \nabla_\theta L(\theta) \quad (4)$$

2.3.2 LSTM

Long Short-Term Memory (LSTM) networks, an extension of NNs are designed to handle long-term dependencies [4]. The fundamental components of LSTM units encompass a cell, an input gate, an output gate, and a forget gate, each playing a distinct role in information processing [16]. In the description below, the symbols W denote weights, while b represents biases. x_t represents the current input and h_{t-1} is the historical output. A visual representation of the various LSTM network components is provided in Fig. 5.

The functioning of an LSTM network can be explained through its individual components:

Forget Gate (f_t): The forget gate determines the information discarded. It operates on x_t and h_{t-1} using weights $W_{f,x}$ and $W_{f,h}$, followed by the application of the sigmoid function $\sigma()$ which yields an output in the range (0,1).

$$f_t = \sigma(W_{f,x}x_t + W_{f,h}h_{t-1} + b_f) \quad (5)$$

Input Gate (i_t): The input gate determines what information should be retained in memory. Weights $W_{i,x}$ and $W_{i,h}$ associated with the x_t and h_{t-1} , facilitate this decision.

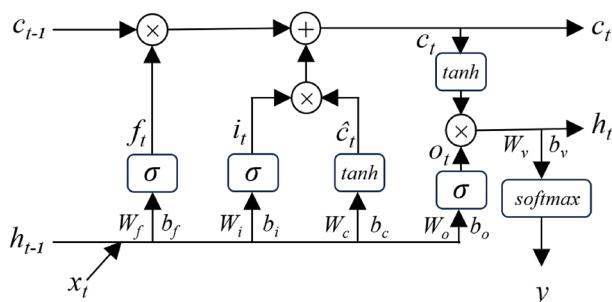


Fig. 5 Structure of an LSTM network highlighting the various gates, weights, biases and operations

$$i_t = \sigma(W_{i,x}x_t + W_{i,h}h_{t-1} + b_i) \quad (6)$$

Current Cell State Estimate (\hat{C}_t): A hyperbolic tangent function $\tanh()$, which generates outputs within the range (-1,1), is employed to compute the estimate of the current cell state. This estimation relies on parameters $W_{c,x}$, $W_{c,h}$, and b_c .

$$\hat{C}_t = \tanh(W_{c,x}x_t + W_{c,h}h_{t-1} + b_c) \quad (7)$$

Current Cell State (C_t): C_t is updated using the previous cell state C_{t-1} , \hat{C}_t , and the outputs of f_t and i_t ,

$$C_t = f_t \times C_{t-1} + i_t \times \hat{C}_t \quad (8)$$

Output Gate (o_t): The output gate interacts with x_t and h_{t-1} through weights $W_{o,x}$ and $W_{o,h}$,

$$o_t = \sigma(W_{o,x}x_t + W_{o,h}h_{t-1} + b_o) \quad (9)$$

Current Output (h_t): h_t is generated by combining the results of the o_t and C_t ,

$$h_t = o_t \times \tanh(C_t) \quad (10)$$

Final Output (y): The final output y is obtained by applying a softmax function to the product of weights W_v and h_t ,

$$y = \text{softmax}(W_v h_t + b_v) \quad (11)$$

LSTM models require access to substantial amounts of historical data to generate meaningful results. This can pose a challenge when dealing with newer or smaller hubs with limited data availability.

2.3.3 Transfer learning

Transfer Learning (TL) is designed to address challenges arising from limited training data. The TL process begins by initially pre-training models on a larger, separate dataset. These models are then fine-tuned through training on the target dataset, allowing them to extract general patterns from the larger dataset and adapt this knowledge to the smaller, intended dataset, as outlined in [6]. TL is employed to enhance learning in a target task, denoted as $\mathcal{T}_T = (D_T, L_T)$, by leveraging knowledge from a different yet related task, $\mathcal{T}_S = (D_S, L_S)$.

The optimal parameters θ_S , obtained from a NN trained on $D_{S,\text{train}}$ and validated on $D_{S,\text{test}}$, serve as the initial weights and biases for the NN model intended for the target task \mathcal{T}_T as shown in Algorithm 1. Subsequently, the TL model undergoes fine-tuning by training on $D_{T,\text{train}}$ and is evaluated on $D_{T,\text{test}}$ to obtain optimal parameters θ_T specific to the target task \mathcal{T}_T .

It is essential to acknowledge a primary limitation of TL models. Fine-tuning may not fully mitigate the disparities in datasets and trends, potentially resulting in sub-optimal performance when transitioning from the source to the target task.

Algorithm 1 Transfer Learning (TL)

Require: $\mathcal{T}_S = (D_S, L_S)$: Source task
Require: α : Step size hyperparameter

- 1: Randomly initialize θ
- 2: **while** not done **do**
- 3: Batch load data (x_i, y_i) from $D_{S,train}$
- 4: Evaluate $\nabla_\theta L_{\mathcal{T}_i}(f_\theta)$ using 2 or 3
- 5: Update $\theta \leftarrow \theta - \alpha \nabla_\theta L_{\mathcal{T}_i}(f_\theta)$
- 6: **end while**
- 7: Use optimal θ_S as initial parameters for fine-tuning

2.3.4 Model-agnostic meta-learning

Model-Agnostic Meta-Learning (MAML) represents a recent advancement within the realm of Deep Learning (DL) that offers a solution to the limitations of conventional DL and TL, as introduced in [9]. In the context of meta-learning, specifically MAML, a large dataset is initially divided into numerous meta-tasks, each comprising only a limited number of data points. The model is subsequently exposed to these tasks, essentially becoming proficient in "learning to learn". This means that instead of directly updating its parameters based on an extensive external dataset, the model focuses on acquiring the ability to efficiently adjust its parameters in order to maximize its performance when faced with new tasks.

In the case of time series data, following the generation of features and time windows as outlined in Sect. 2.2, the dataset $D = (x_n, y_n)$ is partitioned into N tasks denoted as $\mathcal{T}_i, i \in (1, N)$ that each contain the same number of time windows $2l$. $2l$ consecutive time windows with a random starting point are selected for each task. Alternate time windows in each task \mathcal{T}_i are segregated into exactly l support sets trained in an inner loop and l query sets trained in the outer loop. This is because alternate time windows are temporally correlated [17].

The NN model utilized for MAML is initialized with random parameters θ . The time windows $(x_{i,k}, y_{i,k}), k \in (1, l)$ in the support set of each task \mathcal{T}_i^{sup} are used to iteratively update temporary parameters θ'_i by using θ and the gradient of the loss function as expressed in Eq. (12) for a number of adaptation steps set by the user. α is the inner loop learning rate.

$$\theta'_i = \theta - \alpha \nabla_\theta L_{\mathcal{T}_i^{sup}}(f_\theta) \quad (12)$$

Following this parameter update, the meta-learning rate β , parameters θ'_i and their gradients as well as time windows $(x_{i,j}, y_{i,j}), j \in (1, l)$ from the query set of each task \mathcal{T}_i^{que} are employed to adjust θ , as demonstrated in Eq. (13).

$$\theta \leftarrow \theta - \beta \sum_{i=1}^N \nabla_\theta L_{\mathcal{T}_i^{que}}(f_{\theta'_i}) \quad (13)$$

The model, now meta-trained, can be fine-tuned on smaller datasets, allowing it to adapt quickly and learn effectively without being hampered by data scarcity or external dataset influences. The entire process is shown in Algorithm 2.

Algorithm 2 Model-Agnostic Meta-Learning (MAML)

Require: Tasks $\mathcal{T}_i, i \in (1, N)$: Derived from dataset D
Require: α, β : Step size hyperparameters

- 1: Randomly initialize θ
- 2: **while** not done **do**
- 3: Sample batch of tasks \mathcal{T}_i
- 4: **for** all \mathcal{T}_i **do**
- 5: Sample data $(x_{i,k}, y_{i,k}), k \in (1, l)$ from support set \mathcal{T}_i^{sup}
- 6: Evaluate $\nabla_\theta L_{\mathcal{T}_i}(f_\theta)$ using equation 2 or 3
- 7: Compute temporary parameters with gradient descent using equation 12
- 8: Sample data $(x_{i,j}, y_{i,j}), j \in (1, l)$ from query set \mathcal{T}_i^{que} for meta-update
- 9: **end for**
- 10: Update θ using equation 13
- 11: **end while**
- 12: Use θ as initial parameters for fine-tuning

2.3.5 Classic machine learning

In order to justify the use of complex deep learning experiments, it is essential to compare performances with classical machine learning (CML) models. We utilize two algorithms, namely Random Forests (RF) and K-Nearest-Neighbors (KNN) due to their popularity, simplicity, and known efficacy at regression problems [18]. Other popular models such as Support Vector Machines were also included in experimentation, but their performances were consistently poor and are thus not included in the results.

The basic unit of a RF is a decision tree. A Decision Tree (DT) is a machine learning model that splits features into branches of a tree. The tree starts with a root node, representing a decision or a test on a specific feature. From there, it branches into child nodes, each corresponding to different outcomes of that decision. This branching continues until the final decisions are reached, expressed as leaf nodes. The splitting criterion, in our case, was MSE as described in Eq. 3.

RF models are a collection of DTs, each trained on a different randomly divided subset of the data. Each DT then predicts the outcome, and the final prediction is based on the average prediction of the ensemble of DTs. This ensemble approach helps to reduce overfitting.

In KNN regression, each data point is positioned in a multidimensional space according to its features. KNN locates the k-nearest instances in this space to the input data point in question via euclidean distance. The predicted value is calculated as the average of the target values of these neighbors as shown in Eq. 14. We selected k as 5.

$$\hat{y}(x) = \frac{1}{k} \sum_{i=1}^k y_i \quad (14)$$

Here, $\hat{y}(x)$ is the predicted value for the input x , k is the number of nearest neighbors considered and y_i are the target values of the k nearest neighbors to the input x .

2.3.6 Experimental setup

In our experimental setup, we employ LSTM networks as the foundational architecture. We configure an LSTM network comprising of two LSTM layers, each housing 48 neurons. This is followed by a Dense layer with 48 neurons and culminating in a single-neuron output layer. To optimize the network, we employ an RMSprop optimizer with a learning rate (α) of 0.001. Our experiments are conducted using Pytorch and the learn2learn library [19] in Python, hosted on Google Colab, which provides ample resources including 12.7 GB of RAM and an NVIDIA Tesla T4 GPU. Multiple variations of the above network were trained and tested, including number of layers, number of neurons in each layer, number of LSTM layers, bidirectionality, and learning rate. The settings mentioned were selected because they provided the best results.

For our meta-learning approach, we commence by partitioning the Colorado dataset into 500 tasks (N), with each task encompassing 200 consecutive data points (2 l). The number of tasks and data points per task were selected on the basis of performance by testing each of them in steps of 50 starting from 50 until 500. The LSTM network is encapsulated within a MAML module with a learning rate (β) of 0.05. We create two distinct models, one trained with L1 loss and the other with MSE loss. Our training process consists of 50 iterations. Within each iteration, the model encounters 32 tasks, where each task is further divided into query and support sets. Specifically, every alternate 24-hour input time window serves as a support set, while its subsequent time window serves as the query set. The model adapts to the support sets in the inner loop and evaluates its performance on the query sets, iteratively updating its parameters to enhance its performance across tasks. In each iteration, a new copy of the MAML model is employed. This practice involves cloning the original model before adaptation, effectively isolating changes made during adaptation to a specific task ensuring proper generalization.

Upon completing the meta-learning phase, we create fresh LSTM models, initializing them with the same parameters θ as the meta-trained LSTM. These models are then subjected to fine-tuning on the remaining datasets (ACN, Norway-shared, and Norway-private). We select 20-day and 40-day subsets from these datasets, splitting them evenly into training and testing sets. Fine-tuning is carried out over 50 iterations, employing L1 loss as the objective function.

In contrast, our TL approach involves training LSTM models with the previously described architecture directly on the entire Colorado dataset. Two separate models are trained—one with L1 loss and the other with MSE loss—for 50 iterations. Subsequently, these models are fine-tuned on the ACN and Norway datasets in a manner consistent with the process outlined above.

For traditional DL, where no pretraining is involved, models are directly trained on the first halves of the selected subsets for 50 iterations using L1 loss and then evaluated on the remaining halves. The CML models are also directly trained on the same portions of the data after flattening the input vector. The evaluation metrics include mean absolute error (MAE), root mean squared error (RMSE), and coefficient of determination (R²) scores. To ensure robustness, each experiment is conducted five times, and the average values of all metrics are reported.

3 Results

This section is divided into 3 subsections pertaining to each of the datasets under investigation. Tables 1, 2 and 3 provide the metrics for the seven EV charging load forecasting models : TL with MSE and L1 loss, MAML with MSE and L1 loss, DL with L1 loss, RF and KNN.

3.1 ACN dataset

Analyzing the ACN dataset results presented in Table 1, it becomes evident that TL L1 exhibits the most favorable performance, particularly in terms of MAE and RMSE. Notably, TL L1's RMSE score is surpassed only by TL MSE in two subsets of the ACN dataset considered. In contrast, DL consistently lags behind the performance of competing models across all metrics. With the exception of the first and third subsets wherein RF performs competitively, CML models are well-beaten by the deep learning models.

Furthermore, when comparing L1 variants to their MSE counterparts within the MAML and TL models, it becomes apparent that L1 variants outshine their MSE counterparts. Additionally, examining the R-squared (R²) scores reveals that TL models excel in capturing and adhering to the underlying trends in the data.

Analyzing the trends depicted in Figs. 6 and 7, we can evaluate how well each model aligns with the actual hourly load patterns. It is evident that, in general, models tend to exhibit suboptimal performance during peak load periods, occasionally either overestimating or underestimating the peak values. Interestingly, models display a higher level of accuracy in predicting periods of zero load, with some exceptions noted in the case of DL and MAML L1, as illustrated in Fig. 6. From both figures, we see that the CML models perform very poorly during zero-load hours. While they are able to understand the general trend, they predict the trend to continue even when there is no actual load.

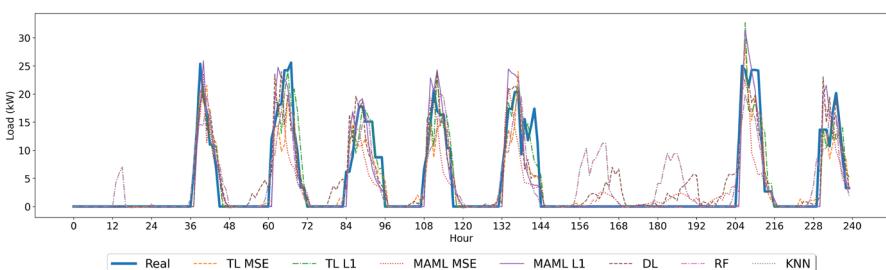
3.1.1 Norway shared charger dataset

Examining the Norway shared charger data presented in Table 2, it is clear that MAML demonstrates the most robust performance, specifically excelling in terms of MAE and RMSE. Interestingly, in this dataset, DL manages to outperform TL models, yet it still falls short when compared to the performance achieved by MAML

Table 1 Mean MAE, RMSE and R2 scores of the forecasting models on the ACN dataset

Specification	Subset	Model	Mean scores		
			MAE	RMSE	R2
18th September 2019, 10 Days		TL MSE	1.48	2.97	0.83
		TL L1	1.40	3.03	0.82
		MAML MSE	1.58	3.17	0.81
		MAML L1	1.53	3.19	0.81
		DL	1.95	3.53	0.75
		RF	1.51	2.99	0.74
		KNN	2.27	3.93	0.54
9th November 2020, 10 Days		TL MSE	0.18	0.76	0.72
		TL L1	0.14	0.70	0.76
		MAML MSE	0.20	0.78	0.70
		MAML L1	0.18	0.79	0.69
		DL	0.19	0.78	0.70
		RF	0.34	0.93	0.16
		KNN	0.5	1.27	0.01
18th September 2019, 20 Days		TL MSE	1.52	3.42	0.68
		TL L1	1.37	3.37	0.69
		MAML MSE	1.47	3.38	0.69
		MAML L1	1.47	3.38	0.69
		DL	1.46	3.39	0.69
		RF	1.51	3.21	0.68
		KNN	1.93	3.79	0.54
9th November 2020, 20 Days		TL MSE	0.30	0.96	0.72
		TL L1	0.29	1.00	0.70
		MAML MSE	0.34	1.05	0.67
		MAML L1	0.31	1.03	0.68
		DL	0.33	1.03	0.67
		RF	0.39	1.10	0.52
		KNN	0.8	1.60	0.09

The values in bold and italics represent the best metrics achieved by the models for the subset of data in consideration

**Fig. 6** Results from the forecasting models compared to the true load value on a subset of the ACN dataset starting from September 18, 2019 for 10 days

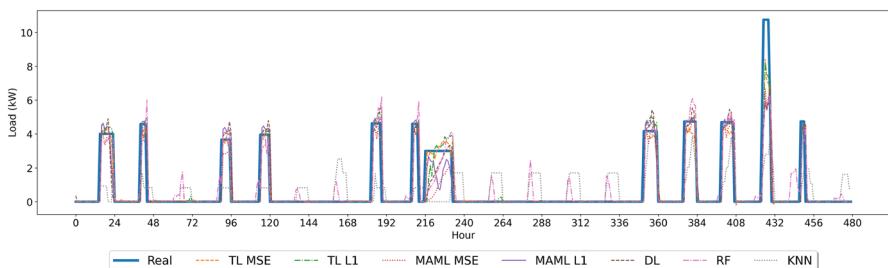


Fig. 7 Results from the forecasting models compared to the true load value on a subset of the ACN dataset starting from November 9, 2020 for 20 days

Table 2 Mean MAE, RMSE and R2 scores of the forecasting models on the Norway shared charger dataset

Specification	Model	Mean scores		
		MAE	RMSE	R2
21st February 2019, 10 Days	TL MSE	0.96	2.38	0.63
	TL L1	0.91	2.39	0.63
	MAML MSE	0.91	2.38	0.63
	MAML L1	0.89	2.32	0.65
	DL	0.92	2.40	0.62
	RF	1.18	2.48	0.32
	KNN	2.28	4.3	-0.32
	TL MSE	3.77	5.64	0.59
	TL L1	3.66	5.50	0.61
17th September 2019, 10 Days	MAML MSE	3.30	5.06	0.67
	MAML L1	3.47	5.29	0.64
	DL	3.60	5.55	0.61
	RF	3.51	4.94	0.46
	KNN	5.88	8.36	-1.84
	TL MSE	1.01	2.43	0.59
	TL L1	1.03	2.44	0.59
21st February 2019, 20 Days	MAML MSE	0.93	2.35	0.62
	MAML L1	0.88	2.25	0.65
	DL	0.97	2.41	0.60
	RF	1.24	2.31	0.49
	KNN	2.17	3.83	-0.23
	TL MSE	3.70	5.64	0.65
	TL L1	3.38	5.50	0.67
17th September 2019, 20 Days	MAML MSE	3.23	5.38	0.68
	MAML L1	3.38	5.52	0.67
	DL	3.33	5.45	0.67
	RF	4.29	5.68	0.33
	KNN	5.93	8.18	-1.03

The values in bold and italics represent the best metrics achieved by the models for the subset of data in consideration

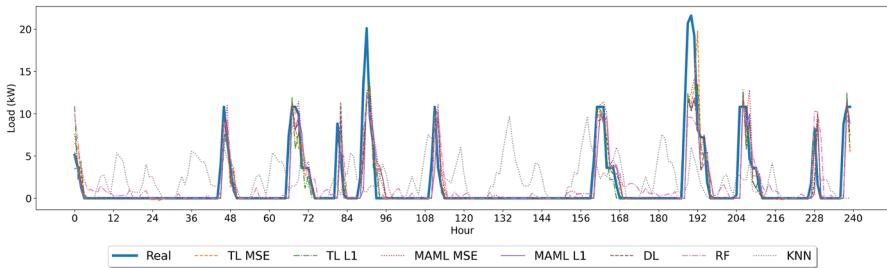


Fig. 8 Results from the forecasting models compared to the true load value on a subset of the Norway shared charger dataset starting from February 21, 2019 for 10 days

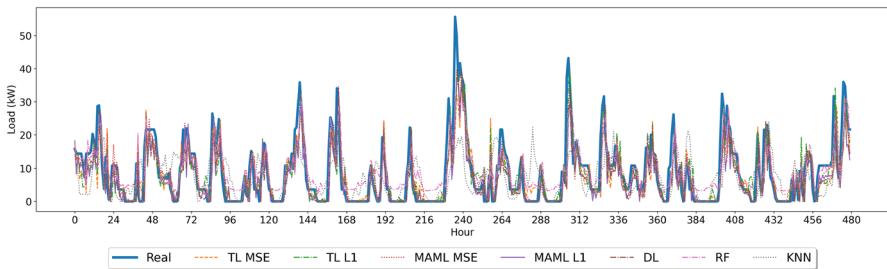


Fig. 9 Results from the forecasting models compared to the true load value on a subset of the Norway shared charger dataset starting from September 17, 2019 for 20 days

models. CML models expectedly underperform, with KNN showing negative R² scores. In the second subset of the dataset, the MAE exhibited by DL is 10% higher than that of MAML MSE but 5% lower than that of TL MSE. Additionally, the R² scores affirm that MAML models tend to make predictions that closely align with the actual data trends, demonstrating their superior predictive capability.

From Figs. 8 and 9, we can assess how each model fits the real hourly load trend. We can see that the models under-perform at the two peaks greater than 15 kW in Fig. 8, and the CML models continue to predict loads even when there are none, as discussed previously. Additionally, Fig. 8 displays an interesting phenomenon—models appear to arrive with a delay of one timestep with respect to the peaks, suggesting that they are under-confident in predicting the starting edges of the peak periods.

3.1.2 Norway private charger dataset

From the data outlined in Table 3, it becomes evident that TL L1 demonstrates the most robust performance across all subsets examined within the Norway private charger dataset. Conversely, DL exhibits relatively poorer performance in most instances.

Table 3 Mean MAE, RMSE, and R2 scores of the forecasting models on the Norway private charger dataset

Specification	Model	Mean scores		
		MAE	RMSE	R2
21st February 2019, 10 Days	TL MSE	2.8	4.95	0.51
	TL L1	2.61	4.93	0.58
	MAML MSE	3.1	6.35	0.2
	MAML L1	2.98	6.18	0.24
	DL	3.12	6.72	0.18
	RF	2.80	5.60	0.14
	KNN	3.66	7.28	-0.10
	TL MSE	5.01	7.73	0.59
17th September 2019, 10 Days	TL L1	4.75	6.96	0.69
	MAML MSE	6.18	9.23	0.46
	MAML L1	5.74	8.18	0.58
	DL	6.24	9.34	0.45
	RF	4.85	6.65	0.61
	KNN	6.24	9.34	0.45
	TL MSE	3.10	5.65	0.58
	TL L1	2.98	5.64	0.58
21st February 2019, 20 Days	MAML MSE	3.92	7.01	0.35
	MAML L1	3.39	6.38	0.45
	DL	3.59	6.66	0.41
	RF	3.23	6.28	0.43
	KNN	5.17	9.01	0.12
	TL MSE	5.56	8.95	0.69
	TL L1	5.39	8.28	0.74
	MAML MSE	5.61	8.45	0.72
17th September 2019, 20 Days	MAML L1	5.39	8.35	0.73
	DL	5.70	8.93	0.69
	RF	5.50	8.57	0.46
	KNN	7.36	10.21	-0.31

The values in bold and italics represent the best metrics achieved by the models for the subset of data in consideration

For example, consider the MAE scores: TL L1 achieves scores that are 20% and 31% lower than those attained by DL for the first and second subsets within the dataset. Moreover, TL L1 outperforms MAML L1 (the better-performing MAML) by approximately 15% and 21% for the same subsets. CML models expectedly underperform again. Additionally, it's worth noting that both MAE and RMSE values for all models within the Norway private charger dataset are higher when compared to those of the Norway shared charger dataset and the ACN dataset. This discrepancy might be attributed to the Colorado dataset containing charging transactions acquired from shared chargers that might exhibit similar patterns to Norway shared

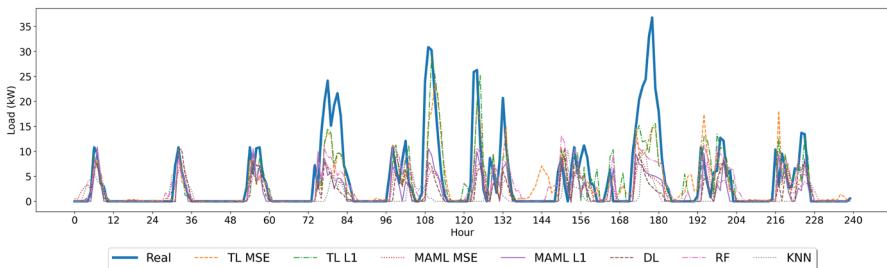


Fig. 10 Results from the forecasting models compared to the true load value on a subset of the Norway private charger dataset starting from February 21, 2019 for 10 days

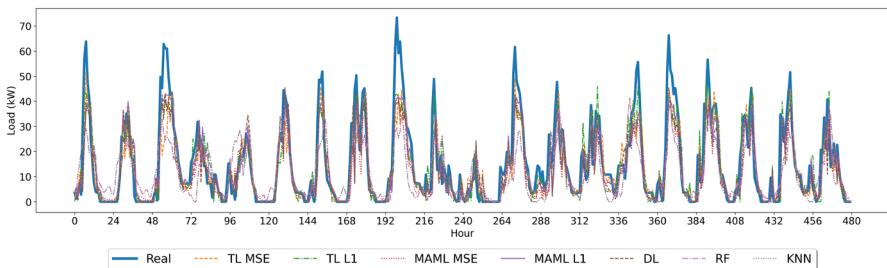


Fig. 11 Results from the forecasting models compared to the true load value on a subset of the Norway private charger dataset starting from September 17, 2019 for 20 days

charger dataset and the ACN dataset (shared chargers on campus). Achieving an optimal balance between private and shared charging data during pre-training may be necessary to enhance the final results.

Analyzing Figs. 10 and 11, we can evaluate how well each model aligns with the actual hourly load trends. Notably, all models, with the exception of TL L1 and TL MSE, exhibit significant underperformance. TL models struggle to accurately predict peaks exceeding 15 kW as seen in Fig. 10. However, Fig. 11 shows a relatively improved performance, though the models still face challenges in accurately predicting peaks exceeding 50 kW.

4 Discussion

Our research demonstrates the effectiveness of MAML and TL models in short-term load forecasting, particularly in scenarios with limited data availability. These models exhibit the capability to forecast accurately across diverse geographical regions, scales, and data distributions by employing a pre-training strategy on a more extensive dataset followed by fine-tuning on the smaller target dataset. Notably, both MAML and TL models exhibit competitive performance, with MAML excelling for the Norway shared charge dataset and TL performing exceptionally well for the ACN dataset as well as the Norway private charger dataset. MAML seems to perform

best when the target dataset closely matches the setting of the source dataset but this does not seem to matter too much for TL. In this work, the Colorado dataset was the only dataset used for pre-training and the data is obtained from shared chargers dispersed in Colorado which is similar to the Norway shared charger dataset with data from shared chargers in Trondheim. Achieving an optimal balance between private and shared charging data during pre-training may be necessary to enhance forecasting metrics. Across all three evaluation metrics, both MAML and TL models consistently outperform traditional Deep Learning (DL) models. Classical Machine Learning (CML) models are clearly not adequate for the complexity of the task. It's worth noting that while the differences in scores are generally significant, there may be instances where DL models outperform specific MAML and TL configurations.

We also see that models tend to exhibit suboptimal performance during peak load periods, occasionally either overestimating or underestimating the peak values. However, most models display a higher level of accuracy in predicting periods of zero load except CML models. While CML models are able to understand the general trend, they predict the trend to continue even when there is no actual load. It is also worth noting that sometimes the models appear to perform well after a delay of one timestep when the load jumps from zero to a higher value. This suggests that the models are under-confident in predicting the starting edges of the peak periods.

A few variations of the models were tested before arriving at the final models. We tried pre-training the models for MAML and TL on the same timespan as the test data i.e. if the test data was October to November 2019 then we pre-trained the models of data only from August to September 2019. The results were poor compared to the results discussed above. Additionally, multiple variations of the models were tested by tuning various hyperparameters before settling on the model architectures utilized for forecasting.

4.1 Future research

The addition of more diverse data in the source dataset for pre-training can potentially improve forecasting accuracy. Another avenue worth exploring is the possibility of predicting individual user-specific charging loads rather than aggregate loads with aggregate load pre-training, potentially protecting privacy and enhancing forecasting precision. Additionally, the adoption of ensemble models (combination of TL and MAML) could be investigated as a means to further improve forecasting metrics.

In the realm of medium and heavy-duty electric vehicles EVs, MAML and TL find valuable applications beyond forecasting aggregate loads. They prove beneficial for predictive maintenance, onboard diagnostics, and adapting to various routes, drivers, and weather conditions. Predictive maintenance models have the capacity to adjust to the unique usage patterns and wear and tear experienced by each individual truck. MAML aids in the generation of optimal driving decisions, facilitating rapid adaptation to diverse driving conditions, routes, and cargo loads.

Mandates in specific regions, such as California, may offer access to extensive early-stage data, which can be harnessed for pre-training. Subsequently, these

models can undergo fine-tuning for future adopters. This process contributes to the planning of advanced high-power charging infrastructure and enhances driving and energy efficiency.

Acknowledgements We are grateful to the sponsorship received from the following Grants 69763, 77739, and 45779 to the department of Mechanical and Aerospace Engineering, UCLA.

Author Contributions SN: conceptualization, investigation, formal analysis, methodology, data curation, software, validation, visualization, writing—original draft preparation, KN: conceptualization, investigation, methodology, data curation, software, validation, visualization, writing—review and editing, CZ: conceptualization, investigation, project administration, writing—review and editing, RG (Rohan Gowda): software, validation, visualization, writing—review and editing, RG (Rajit Gadh): supervision, project administration, funding acquisition, writing—review and editing.

Availability of data and materials The publicly available data sources have been cited.

Declarations

Conflict of interest The authors declare that there is no potential conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Balasubramanian, S., Balachandra, P.: Effectiveness of demand response in achieving supply-demand matching in a renewables dominated electricity system: A modelling approach. *Renew. Sustain. Energy Rev.* **147**, 111245 (2021)
2. Huang, X., Wu, D., Boulet, B.: Ensemble learning for charging load forecasting of electric vehicle charging stations. In: 2020 IEEE Electric Power and Energy Conference (EPEC), pp. 1–5 (2020). IEEE
3. Sadeghianpourhamami, N., Refa, N., Strobbe, M., Develder, C.: Quantitive analysis of electric vehicle flexibility: a data-driven approach. *Int. J. Electr. Power Energy Syst.* **95**, 451–462 (2018)
4. Hewamalage, H., Bergmeir, C., Bandara, K.: Recurrent neural networks for time series forecasting: current status and future directions. *Int. J. Forecast.* **37**(1), 388–427 (2021)
5. Lee, E., Rhee, W.: Individualized short-term electric load forecasting with deep neural network based transfer learning and meta learning. *IEEE Access* **9**, 15413–15425 (2021)
6. Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. *J. Big Data* **3**(1), 1–40 (2016)
7. Forootani, A., Rastegar, M., Zareipour, H.: Transfer learning-based framework enhanced by deep generative model for cold-start forecasting of residential ev charging behavior. *IEEE Trans. Intell. Veh.* (2023)
8. Banda, P., Bhuiyan, M.A., Hasan, K.N., Zhang, K.: Assessment of hybrid transfer learning method for forecasting ev profile and system voltage using limited ev charging data. *Sustain. Energy Grids Netw.* **36**, 101191 (2023)
9. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning, pp. 1126–1135 (2017). PMLR

10. He, Y., Luo, F., Ranzi, G.: Transferrable model-agnostic meta-learning for short-term household load forecasting with limited training data. *IEEE Trans. Power Syst.* **37**(4), 3177–3180 (2022)
11. Xiao, X., Mo, H., Zhang, Y., Shan, G.: Meta-ann—a dynamic artificial neural network refined by meta-learning for short-term load forecasting. *Energy* **246**, 123418 (2022)
12. Lee, Z.J., Li, T., Low, S.H.: Acn-data: analysis and applications of an open ev charging dataset. In: Proceedings of the Tenth ACM International Conference on Future Energy Systems, pp. 139–149 (2019)
13. Sørensen, Å.L., Lindberg, K.B., Sartori, I., Andresen, I.: Analysis of residential ev energy flexibility potential based on real-world charging reports and smart meter data. *Energy Build.* **241**, 110923 (2021)
14. Boulder Open Data, C.: Electric vehicle charging station data. https://open-data.bouldercolorado.gov/datasets/95992b3938be4622b07f0b05eba95d4c_0/explore
15. Sørensen, Å.L., Lindberg, K.B., Sartori, I., Andresen, I.: Residential electric vehicle charging datasets from apartment buildings. *Data Brief* **36**, 107105 (2021)
16. Dong, Y., Chen, Y., Zhao, X., Huang, X.: Short-term load forecasting with distributed long short-term memory. In: 2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), pp. 1–5 (2023). IEEE
17. Arango, S.P., Heinrich, F., Madhusudhanan, K., Schmidt-Thieme, L.: Multimodal meta-learning for time series regression. In: Advanced Analytics and Learning on Temporal Data: 6th ECML PKDD Workshop, AALTD 2021, Bilbao, Spain, September 13, 2021, Revised Selected Papers 6, pp. 123–138 (2021). Springer
18. Fernández-Delgado, M., Sírsat, M.S., Cernadas, E., Alawadi, S., Barro, S., Frerero-Bande, M.: An extensive experimental survey of regression methods. *Neural Netw.* **111**, 11–34 (2019)
19. Arnold, S.M., Mahajan, P., Datta, D., Bunner, I., Zarkias, K.S.: learn2learn: a library for meta-learning research (2020). arXiv preprint [arXiv:2008.12284](https://arxiv.org/abs/2008.12284)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Shashank Narayana Gowda¹  · Keshav Nath² · Chen Zhang¹ · Rohan Gowda³ ·
Rajit Gadh¹

 Shashank Narayana Gowda
shashank07@ucla.edu

Keshav Nath
nathkeshav14@gmail.com

Chen Zhang
chenzhang96@g.ucla.edu

Rohan Gowda
rsg3418@g.coppellisd.com

Rajit Gadh
gadh@ucla.edu

¹ Smart Grid Energy Research Center, University of California Los Angeles, Los Angeles, CA 90095, USA

² Department of Mechanical Engineering, Delhi Technological University, New Delhi, Delhi 110042, India

³ Coppell High School, Coppell, TX 75019, USA