A Privacy-Preserving Prediction Model for Individualized Electric Vehicle Energy Consumption

Xudong Hu and Biplab Sikdar, *Senior Member, IEEE,*

***Abstract*—Electric vehicles (EVs) have been gaining popularity in recent years but range anxiety of drivers and the ability to predict the energy consumption of EVs remains an important problem. While machine learning offers promising solutions for energy consumption prediction, it also introduces privacy challenges, especially when handling sensitive user data. As data-driven models become ubiquitous, ensuring the privacy and security of user information is paramount. This paper not only presents an innovative approach for EV energy prediction but also emphasizes the importance of privacy considerations in machine learning applications. We use a system that integrates key parameters such as ambient temperature, road gradient, and vehicle load to simulate real-world EV usage. By utilizing an innovative transformation layer that enables minimal low-level feature sharing and maintains maximum independence between groups, the system is designed to produce multiple simultaneous predictions for individual EVs while protecting privacy. Empirical results validate the system's ability to concurrently generate accurate predictions, outperforming conventional single-output models. Additionally, it provides a granular accuracy analysis across diverse EV models. We advocate for a balanced approach, harnessing data's potential while upholding stringent privacy standards and our experimental results show that the proposed model is robust against various attacks that seek to compromise user privacy.**

***Index Terms*—Electrical Vehicle, Energy Consumption prediction, Parallel processing, Time series data, Inference attack, Predictive model**

I. Introduction

THE combustion of fossil fuels which are predominantly used in transportation accounts for a substantial portion of energy-related CO_2 emissions. The International Energy Agency (IEA) highlights that road transport alone contributes to three-quarters of these emissions [1]. Given this substantial footprint, there is an urgent need to innovate and adopt advanced vehicle and fuel technologies. Electrification, particularly in the form of electric vehicles (EVs), is a promising solution, offering potential environmental benefits by significantly reducing CO_2 emissions from road transport. However, the adoption rate of EVs has been sluggish, primarily due to drivers' range anxiety, stemming from uncertainties in the remaining driving range and an underdeveloped public charging infrastructure [2].

To address the prevalent concern of range anxiety, a predictive model that offers a nuanced understanding of energy consumption is paramount [3]. Such a model serves a dual purpose: from the service provider's perspective, it facilitates the strategic placement and planning of charging infrastructure, ensuring that EV users have access to charging facilities when and where they need them. This not only optimizes the utility's resources but also enhances the overall user experience. On the other hand, from the customer's standpoint, a reliable predictive model can assuage range anxiety by offering accurate forecasts of energy consumption based on various driving conditions and patterns. A comprehensive grasp of EV performance, derived from such models, can bolster the confidence of potential EV adopters, nudging them towards making the switch. Furthermore, with an accurate prediction model in place, drivers can plan their routes more efficiently, ensuring that they have sufficient charge for their journeys and know where to recharge when necessary.

Building on this notion of predictive modeling, the state of charge displayed on vehicle dashboards provides a measure of the remaining energy, but the true challenge lies in accurately predicting energy consumption. With advancements in technology, both traditional statistical models and machine learning or neural networks have been employed to enhance energy consumption prediction [4]. However, a critical limitation of the current methodologies is their propensity to generate singular predictions that predominantly reflect the average tendencies of the training dataset, thereby inadvertently marginalizing the idiosyncratic behaviors of specific subgroups, such as distinct car models. This generalization bias dilutes the nuances of individual patterns during the training phase. Moreover, as machine learning models become integral to such predictions, concerns about privacy arise. Inference attacks, where malicious entities deduce sensitive information from model outputs, and the potential misuse of manually inserted information to build these systems, underscore the need for privacy considerations in machine learning studies [5].

Conventional methods often neglect the subtle behavioral variations among individuals or groups, and they typically lack a comprehensive analysis addressing privacy concerns [4]. To bridge this gap, this paper introduces an innovative approach that encompasses the following four key aspects:

– *Individualized prediction:* By focusing on the behavioral

Manuscript is created and submitted on 11th February 2024; This research was supported in part by Singapore Ministry of Education Academic Research Fund Tier 1 Grants R-263-000-E78-114 and R-263-001- E78-114.; Corresponding author: Biplab Sikdar.

Xudong Hu is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117583 (e-mail: e0459193@u.nus.edu).

Biplab Sikdar is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117583 (e-mail: bsikdar@nus.edu.sg).

differences among subgroups, our system achieves enhanced performance.
- *Multiple-output configuration:* The novel implementation of the transformation layer empowers the system to predict for multiple individuals simultaneously within a single system.
- *Robustness:* The system's resilience is evident through its consistent performance, even under conditions with noise and fast gradient sign method (FGSM) attacks.
- *Privacy safeguard:* By evaluating the system's performance against three distinct inference attacks, we demonstrate its capability to address privacy concerns effectively.

This paper is structured as follows: After this introduction, Section II provides an overview of current research in the field. Section III details our proposed approach, encompassing data collection, preprocessing, and machine learning model development. Section IV discusses the results, and the paper concludes with key findings and implications for future research in Section V.

II. Related works

Predictive analytics for EV energy consumption has witnessed substantial explorations and advancements in recent years. These endeavors predominantly revolve around three central methodologies: analytical models, statistical models, and machine learning (ML) models, each with its unique attributes and applications.

Analytical models are grounded in the principles of physics, employing a series of equations that encapsulate the fundamental dynamics of vehicle motion. These models calculate the net force propelling the vehicle or the power requisite for such force, subsequently deriving the energy consumption through established relationships with force [6], [7] or power [8], [9]. These equations typically involve variables such as the net force F essential for vehicle motion, distance traveled d, power required P, and operation time t.

In contrast, statistical models leverage empirical data to ascertain the mathematical correlations between vehicle energy consumption and various influential factors. These models are inherently reliant on data and necessitate the preliminary definition of relationships between predictor and response variables [10]. Techniques such as regression analysis are commonplace, with prior research employing linear regression models for EV energy consumption predictions [11], [12]. Additional studies have delved into the effects of specific factors like temperature [13] and road gradient [14] on energy consumption.

Machine learning models, particularly those based on neural networks, have gained traction with the advent of enhanced computational capabilities. Research in this domain has experimented with Artificial Neural Network (ANN) architectures, incorporating inputs like vehicle speed, acceleration, jerk, and road information to model driver behavior and predict energy consumption [15], [16]. The convolutional neural network (CNN), another prominent approach, has been utilized for real-time energy consumption predictions [3], [4]. However, these studies often limit their scope to specific EV models or a narrow set of features, overlooking the multifaceted nature of factors influencing energy consumption, as highlighted by [17]. Moreover, the Graph Convolutional Network (GCN) [18], despite being a promising model for prediction tasks, demands high-quality training data and is prone to the over-smoothing issue as the network depth increases, which can diminish the model's predictive performance.

Despite the significant strides in prediction accuracy and model sophistication, a conspicuous gap persists in the realm of privacy preservation within these predictive models [3], [4], [18], [19]. The contemporary academic discourse has not sufficiently addressed the integration of robust privacy measures in EV energy consumption prediction models. As user data is harnessed and employed for model training, it inadvertently creates a potential for privacy breaches. The sanctity of user data is paramount, especially given the intimate nature of the information that can be inferred from driving patterns and energy consumption behaviors. The significance of privacy-preserving mechanisms has been established in related domains, such as studies spotlighting inference attacks that endeavor to extract sensitive information from model predictions, pose significant threats to user privacy [20]. Additionally, the specter of adversarial attacks introduces a further layer of security concerns. Such attacks, designed to subtly distort model outputs by introducing minuscule perturbations to the input, are exemplified by techniques like the (FGSM) [21]. While these studies provide valuable insights into potential vulnerabilities and defense mechanisms, their methodologies and findings are not readily translatable to the context of EV energy consumption predictions. This underscores the need for development of privacy-preserving mechanisms tailored to the intricacies of EV energy consumption predictive models, and this paper seeks to address this problem.

Extending our preliminary work published in [19], this paper improves the model prediction accuracy and addresses the over-fitting problem. Besides, we also employ three different inference attacks as evaluation metrics to show the model's robustness in protecting user privacy.

III. Methodology

A. Dataset

To generate a dataset that is representative of diverse driving patterns, this study employs the emobpy simulation model [22]. emobpy, an open-source framework designed for the generation of time-series data for battery electric vehicles (BEVs) using Python, offers a comprehensive range of adjustable parameters, including driver behavior and vehicle models. The model facilitates the simulation of intricate driving profiles by allowing the user to establish a probability map for various factors such as $departure\ and\ destination$, $distance\ and\ duration$, and $daily\ trip\ frequency$. Notably, the configuration possibilities within emobpy enable the close emulation of German driving behaviors. This is achieved by aligning the cumulative distribution of trips and distances

TABLE I: Driver category and rule setting

Condition	Full-time commuters		Part-time commuters		Non-commuters	
	Weekday	Weekend	Weekday	Weekend	Weekday	Weekend
Category probability	0.4	0.4	0.3	0.3	0.3	0.3
Minimum Number of trips	1	1	1	1	1	1
Last trip destination	Home	Home	Home	Home	Home	Home
Minimum time at home	9	6	9	6	9	6
Trip to work	At least 1	Based on need	At least 1	Based on need	N.A.	N.A.
Minimum time at workplace	7	3	3.5	3	N.A.	N.A.
Maximum Time at workplace	8	4	4	4	N.A.	N.A.
Minimum state duration at workplace	3.5	3	3.5	3	N.A.	N.A.
Minimum state duration except for workplace	0.25	0.25	0.25	0.25	0.25	0.25

with the foundational statistics of German mobility [23]. Furthermore, emobpy categorizes drivers into three distinct groups based on their trip patterns during weekends and weekdays, as detailed in Table I. Among these categories, full-time and part-time commuters exhibit more consistent daily trips relative to non-commuters, primarily due to their regular commuting to workplaces. In terms of vehicle selection, emobpy encompasses an array of pre-defined EV models, each characterized by specific parameters such as battery capacity, motor type, and torque. For the purposes of this study, a deliberate selection approach was adopted, wherein four EV models were uniformly chosen to concentrate on the analysis of subgroup characteristics. During the data generation phase, a driver category is selected based on the probabilities outlined in Table I, following which one of the four predetermined EV models is randomly allocated to the selected driver category. The simulation parameters were set to span one year with a time resolution of 15 minutes, resulting in each driver having 35,040 timestamped records. In total, the simulation yielded 200 unique driver records, cumulatively comprising approximately seven million timestamped data points, as delineated in Algorithm 1. These records were utilized in the training phase. For evaluation, an additional set of 200 driver records with the same length and resolution was generated, ensuring a robust and comprehensive assessment framework.

Algorithm 1: Dataset Generation Algorithm

$Init\ Categories,\ vehicle\ brand\ ,and\ mean\ passenger\ numbers\ and\ then\ assign\ probability$

$//\ 'CFT':\ \ full-time\ \ commuter$
$//\ 'CPT':\ \ part-time\ \ commuter$
$//\ 'NFT':\ \ non-commuter$
$Cat_values \leftarrow ['CFT': 0.4, 'CPT': 0.3, 'NFT': 0.3]$

$//\ \ Distribution: 0.4, 0.3, 0.2, 0.1$
$mean\_passenger\_number \leftarrow [1.5, 2, 2.5, 3]$

$//\ \ vehicle\_selection\ \ and\ \ distribution$
$Volkswagen\ \ \leftarrow ID.3:\ 0.25$
$BMW \leftarrow i3s\ \ Edition\ \ RoadStyle\ 42\ kWh:\ 0.25$
$Audi\ \leftarrow e-tron\ \ Sportback\ 55\ \ quattro:\ 0.25$
$Tesla\ \leftarrow Model\ X\ \ Long\ \ Range\ (SR):\ 0.25$

for $counter\ = 0,\ counter < 200,\ counter++$ **do**
- Pick driver category vehicle_model
- set rules of category based on Table I
- set total hours = 8760
- set time step = 0.25
- set Trips per day
- set Distance and duration
- Generate the travel summary profile
- set vehicle_model
- set mean_passenger_number
- set the rest of the parameter
- Generate the Consumption time_series file
- Combine data to form dataset

end

B. Features and transformation and normalization

Contrary to studies that narrowly concentrate on engine efficiency, such as [3], real-world battery consumption in electric vehicles is influenced by a multitude of factors. Skuza et al. [17] underscore the significance of various elements including wind speed, ambient temperature, road inclination, and vehicle load, among others, in the context of battery usage. In light of these insights, our study incorporates a selection of pertinent features available within the simulation tool, as enumerated in Table II. In a novel approach, Wang et al. [24] introduced an encoding technique to transform time series data into image-like representations, termed Gramian Angular Fields (GAF). This methodology commences with the normalization of the time series data F = $f_1, f_2, \cdots, f_n$ to a range of [-1, 1] using the equation:

$$\widetilde{f}_i = \frac{(f_i - \max(F)) + (f_i - \min(F))}{\max(F) - \min(F)}. \tag{1}$$

Subsequent to feature scaling, the data points are transposed into polar coordinates, incorporating both value and time indices, as expressed by:

TABLE II: Selected features and their description

Feature name	Description
Number of passengers	Number of passengers to obtain vehicle loading
Vehicle speed	Speed at different timestamp
Driving Cycle	Worldwide Harmonized Light Vehicles Test Cycle (WLTC) or Environmental Protection Agency (EPA)
Road gradient	Describe the slope in radians
Road type	Build in road type to obtain rolling resistance coefficient
Temperature	Ambient temperature in Kelvin
Wind speed	Local Wind speed
Weekend	Weekend or weekday indicator
Category	Indicate if driver is full-time/part-time/non-commuters

$$\begin{cases} \phi = \arccos(\widetilde{f}_i), -1 \leq \widetilde{f}_i \leq 1, \widetilde{f}_i \in \widetilde{F}_i \\ r = \frac{t_i}{N}, t_i \in N \end{cases} . \tag{2}$$

Here, t_i are the individual time stamps, while N signifies the total duration encompassed by the polar coordinate system. The GAF matrix is subsequently construed as the inner product within the time series feature space, formulated as:

$$\widetilde{F}' \cdot \widetilde{F} - \sqrt{I - \widetilde{F}^2}' \cdot \sqrt{I - \widetilde{F}^2} \quad . \tag{3}$$

In this context, I represents the unit vector. The outcome of this process is an $N \times N$ square matrix, where N corresponds to the number of time stamps, effectively encapsulating the temporal dynamics amongst them.

C. Architecture

A main objective of our work is to obtain predictions for multiple individuals or groups of individuals simultaneously, using a single prediction model. Thus, we propose a transformation layer with two characteristics: Linear transformation and maximum independence between the individuals or groups. This layer preserves the intrinsic properties of the dataset through its linear nature. For data x passing through a feed-forward hidden layer, the output y is formulated as:

$$y = \sigma(w \cdot x + b)$$

where σ denotes the activation function, w represents the weights, and b is the bias.
With our transformation, the modified output is computed as:

$$x' = Ax + c$$

$$y' = \sigma(w \cdot (Ax + c) + b)$$

where A is the transformation matrix, and c is the bias vector. This transformation achieves our objective with new weights and bias as:

$$y' = \sigma(w' \cdot x + b')$$

where $w' = wA$ and $b' = w \cdot c + b$.
This layer also ensures maximum independence by integrating data from different groups within various domains. Similar to signal processing, where merging data in the frequency domain maintains content independence during time domain transmission, we adopt the 2D-Discrete Cosine Transformation (2D-DCT) inspired by [25]. This method ensures feature independence during the parallel processing of multiple inputs. The 2D-DCT is expressed as:

Algorithm 2: Transformation Algorithm

input : $Individual\ driving\ record$
output: $Transformed\ data\ record$

$Load\ dataset\ from\ the\ path$
$Drop\ unwanted\ columns$
$Create\ dictionary\ in\ the\ range\ (0, N-1)\ based\ on\ vehicle\_selection$

$Rx\_i \leftarrow 9\ features\ from\ Table$ II
$Ry\_i \leftarrow Time\ series\ consumption$
$//\ \ i\ \ \leftarrow dict\{vehicle\_model\}$

$//\ General\ process$
$Rx\_i \leftarrow Rx\_i.reshape(-1, 48, 9)$
$Rx\_i \leftarrow add\ 2\ pad\ at\ bottom$
$Rx\_i \leftarrow transpose(0, 2, 1)$
$Rx \leftarrow concatenate\ Rx\_i\ on\ axis\ 2$

$Ry\_i \leftarrow Ry\_i.reshape(-1, 48, 1)$
$Ry\_i \leftarrow add\ 2\ pad\ at\ bottom$
$Ry\_i \leftarrow transpose(0, 2, 1)$
$Ry \leftarrow concatenate\ Ry\_i\ on\ axis\ 2$

$//\ transform\ and\ create\ dataset$
for $rows\ in\ Rx$ **do**
  $Create\ input\ matrix\ S\ //\ \ 100 \times 100$
  for $model\ in\ models$ **do**
    $Get\ N\ based\ on\ model\ dictionary$
    $i \leftarrow N * 50,\ j \leftarrow 50 * (N + 1)$
    $r \leftarrow int(N/2) * 50,\ l \leftarrow 50 * (N\%2 + 1)$
    $S\_temp \leftarrow Rx[:, i : j]$
    $S\_temp\ \leftarrow normalize\ from\ Eq.1$
    $S\_temp\ \leftarrow GAF\ representation\ from\ Eq.3$
    $S\_temp\ \leftarrow 2D - DCT\ from\ Eq.4$
    $S[:, r : r + 50, l : l + 50] \leftarrow S\_temp$
  end
  $S \leftarrow Invers\ 2D - DCT$
end

$$X_{k1,k2} = \sum_{n1=0}^{N_1-1} \sum_{n2=0}^{N_2-1} x_{n1,n2} \cos\left[\frac{\pi}{N_1}\left(n_1+\frac{1}{2}\right)k_1\right] \cos\left[\frac{\pi}{N_2}\left(n_2+\frac{1}{2}\right)k_2\right] \quad (4)$$

$$k = 0, ..., N-1.$$

This transformation also retains the property that convolution prior to transformation is equivalent to multiplication post-transformation:

$$\{g * h\}(X) = T^{-1}\{G \cdot H\}. \quad (5)$$

Here, $*$ denotes the convolution operation, and $\cdot$ indicates point-wise multiplication. The transformation layer processes the time-series data as depicted in Algorithm 2. During this phase, data is converted into an image representation format of (100×100). For the multiple-output implementation, rather than expanding the input matrix size to (200×200)—a process that demands additional computational resources—the input matrix is segmented into four 50×50 regions. This approach effectively balances efficiency with computational resource demands. Each region's output is the convolutional result of the transformed driver's data with a mask matrix (50×50, with a value of 1). Beyond ensuring data independence across different subgroups, this transformation augments privacy safeguards. The mask matrix serves a dual purpose: it not only aids in the convolution process, but also introduces an element of noise and low level feature sharing between the groups, bolstering the robustness of privacy protection. The comprehensive transformation procedure, encompassing steps like normalization, GAF matrix conversion, and 2D-DCT transformation preceding the convolutional layer, is graphically represented in Figure 1.

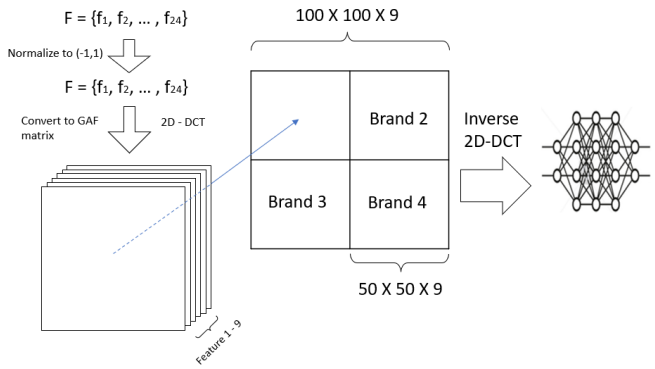


Fig. 1: Transformation illustration of one subgroup.

Convolutional layers excel at discerning image characteristics [26], [27], and hence, are our choice of image encoding strategy for optimal compatibility. The remaining system comprises of convolutional layers, Rectified Linear Units (ReLU), max-pooling layers, and fully connected layers. The final fully connected layer, X_{out}, varies depending on the output format:

- *Single-output model:* $X_{out} = 1$ for period consumption prediction; $X_{out} = 100$ for timestamp consumption prediction.
- *Multiple-output model:* $X_{out} = 4$ for period consumption prediction; $X_{out} = 200$ for timestamp consumption prediction.

To mitigate the issue of dead neurons, Leaky ReLU is utilized in place of the standard ReLU function. At last, given that energy consumption is intrinsically linked to the frequency of trips, the consumption metric defaults to 0 in the absence of any travel activity by the driver. Consequently, an output value of 0 is often anticipated. Furthermore, recognizing the significant correlation between consumption and trip frequency, and the prevalence of zero values in the anticipated output, we've instituted a cumulative sum at the output for an alternative, more convergent evaluation. This cumulative sum is computed as follows:

$$\widetilde{Y}_{i+1} = Y_{i+1} + Y_i. \quad (6)$$

A comprehensive depiction of the system architecture is delineated in Table III.

TABLE III: Architecture of overall system

Layer	Filter	Kernel size	Stride
Transformation	-	-	-
Convolution	32	(7,7)	(2,2)
leakyrelu	-	-	-
Convolution	64	(5,5)	(1,1)
leakyrelu	-	-	-
pooling	-	(2,2)	(2,2)
Convolution	128	(5,5)	(1,1)
leakyrelu	-	-	-
Convolution	128	(3,3)	(1,1)
leakyrelu	-	-	-
pooling	-	(2,2)	(2,2)
Convolution	64	(3,3)	(1,1)
leakyrelu	-	-	-
AdaptiveAvgPool	-	-	-
Dropout(0.5)	-	-	-
dense (4000)	-	-	-
dense (1000)	-	-	-
dense (200)*	-	-	-

D. Objective Functions and Optimizer

For the evaluation of regression tasks, Mean Absolute Error (MAE) and Mean Squared Error (MSE) are frequently utilized metrics, defined in Equations (7) and (8), respectively:

$$MAE\ Loss = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right|, \quad (7)$$

$$MSE\ Loss = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2. \quad (8)$$

In these equations, $\hat{Y}_i$ represents the prediction output from the neural network for the i-th day, while Y_i stands for the

[1] $*$ Output layer X_{out} size varies according to the model configuration.

actual consumption. Our approach involves employing both of these objective functions across various training sessions to discern the most advantageous decision-making criterion.

Regarding the optimization process, we have chosen the Adam optimizer [28] for its efficacy in updating neural network weights during the backpropagation phase of training. Complementing this, we implement an adaptive learning rate strategy, initiating with a rate of 0.001, to further refine our model's performance.

E. Evaluation

The evaluation of the proposed method encompasses several dimensions, including the performance metrics outlined by [19], system resilience in the face of additive Gaussian noise and adversarial attacks, and the preservation of privacy against various inference attacks. A total of 10 models have been developed to facilitate a thorough analysis of the method's efficacy under diverse conditions. These models comprise of eight timestamp consumption models, derived from permutations of objective functions, the application of cumulative sums, and the choice of multiple-output configurations. Additionally, two models are dedicated to predicting period sum for single-output scenarios, contingent on the selection of the objective function. The diverse scenarios and corresponding evaluation metrics are detailed in Table IV. MAE (Equation (7) and Root Mean Square Error (RMSE) offer direct assessments of the discrepancy between predicted and actual energy consumption values. To refine our model's predictive accuracy—particularly in situations where the absolute differences may not fully capture prediction dynamics due to large prediction magnitudes—we have incorporated an evaluation using the Mean Absolute Percentage Error (MAPE). The MAPE assesses accuracy based on proportional differences, offering a more nuanced view. Additionally, the R^2 score is utilized to gauge the extent to which variance in the dependent variable is explained by the model. These metrics are defined as follows:

$$RMSE\ Loss = \sqrt{\frac{\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2}{n}}, \tag{9}$$

$$MAPE\ Loss = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - \hat{Y}_i}{Y_i}\right|, \tag{10}$$

$$R^2 score = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - mean(Y))^2}. \tag{11}$$

The MAPE metric ensures a balanced and relative evaluation of prediction accuracy, that is especially beneficial when predictions involve large numerical values. Conversely, the R^2 score provides a measure of how well observed outcomes are replicated by the model, thereby offering insights into the strength of the model's explanatory power.

To evaluate the privacy implications of our proposed system, particularly in relation to safeguarding against potential inference attacks, we conducted a comprehensive analysis. This analysis focused on the system's resilience against three types of inference attacks: transformation index inference attacks, population inference attacks, and membership inference attacks. Each attack presents unique challenges and aims to exploit different aspects of the system to gain sensitive information either about specific individuals or groups within the dataset. By employing various strategies, including manipulating input data and analyzing the model's output behavior, we aim to assess the system's ability to protect against these potential vulnerabilities.

IV. Results

A. Time series output

To obtain precise energy consumption predictions within specific time periods, we employed two distinct network configurations. The initial approach entailed a direct prediction of total energy consumption, represented as a singular value, necessitating an X_{out} setting of 1 in the terminal fully connected layer. Conversely, the second strategy involved forecasting energy consumption for each timestamp, and then aggregating these to obtain the consumption of a total of 100 timestamps. A comparative analysis of these methodologies revealed superior mean and median values for the time-series output relative to the total consumption prediction, as detailed in Table V. Both models were calibrated using the MAE objective function, with the observed performance disparity attributed to the additional penalization inherent in the MSE function.

B. Performance and robustness

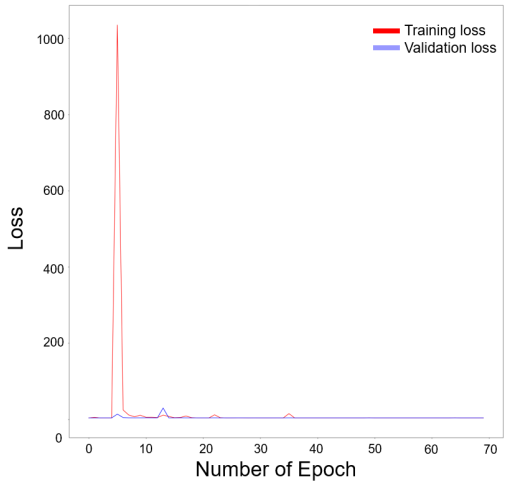
The integration of 2D-DCT enables the model to concurrently process four data clusters corresponding to the four car models. As indicated in Table VI, our best performed model exhibits a reduced error compared to the leading model presented in [3]. For a comprehensive evaluation of both performance and robustness across all models, Table VII enumerates the models, delineating variations in output mode, objective function, and the incorporation of a cumulative sum, assessed via five metrics across four distinct scenarios as detailed in Table IV. The evaluation encompasses two methodologies:

- *Per Timestamp Evaluation:* This metric is computed for individual timestamps, reflecting the mean efficacy of the output neurons.
- *Aggregate Evaluation:* This metric appraises accuracy based on total consumption over the period, indicative of the model's proficiency in predicting driver behavior over an extended time frame.

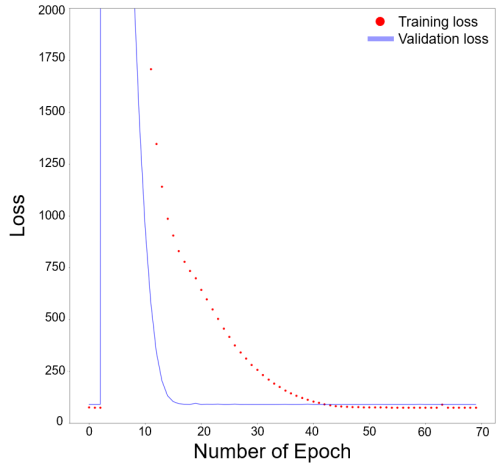
Our analysis of the model performance, as detailed in Table VII, reveals that using MSE as the objective function consistently results in suboptimal performance across various output configurations. During the training phase, the MAE objective function outperformed the MSE function in two key aspects: convergence and convergence speed. The prevalence of zero values in the target consumption time series led to occasional failure in achieving convergence with the MSE function, as reflected in the R^2 score. Furthermore, the rate of

TABLE IV: Model performance comparison before and after convergence

Evaluation Scenario	Description	Evaluation metric
Normal Condition	ideal situation	RMSE,MAE,MAPE,R^2
noise present	additive gaussian noise is added during the process	RMSE,MAE,MAPE,R^2
adversarial attack	adversarial example is generated based on gradient obtained	RMSE,MAE,MAPE,R^2
adversarial attack with noise present	adversarial attack under a noisy input	RMSE,MAE,MAPE,R^2
transformation index inference attack	adversarial has a set of data record, trying to figure out transformation index this data belongs to	channel response error similarity decision histogram
population inference attack	adversarial has a set of data record, trying to figure out the model response for population characteristics inference	receiver operating characteristic curve (ROC) Area Under the Curve (AUC)
membership inference attack	adversarial has a set of data record, build shadow model and decide whether the record belongs to the training data	receiver operating characteristic curve (ROC) Area Under the Curve (AUC)



(a) Loss during training for MAE objective function



(b) Loss during training for MSE objective function

Fig. 2: Comparison of converge speed between MAE and MSE as objective function.

TABLE V: Comparison of Time-series output to Total Sum output

Test Name	Xout = 1		Xout = 100	
	RMSE	MAE	RMSE	MAE
Mean test	18.1	6.62	15.9	5.34
Median test	3.2	3.23	2.45	2.45

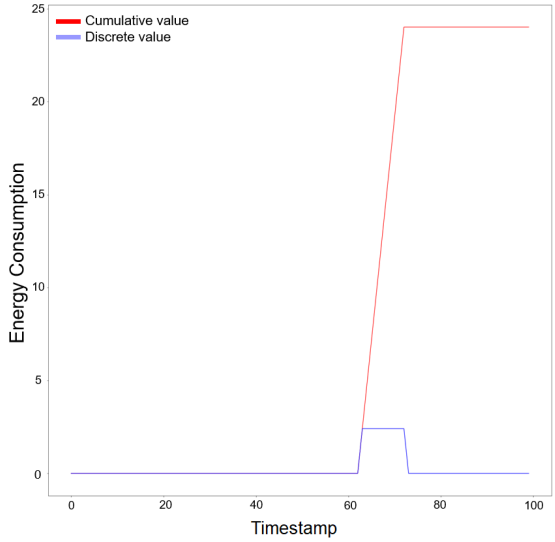


Fig. 3: Illustration of the 0 value reduction from cumulative sum

convergence was significantly slower with the MSE objective function, as shown in Figure 2b, in contrast to the MAE function, depicted in Figure 2a. This slower convergence with MSE is counterintuitive in many cases and is attributed to the large gradients observed during transitions between 'driving' and 'non-driving' states. Implementing a cumulative sum approach, as shown in Figure 3, effectively addressed the issue of zero values in the target time series array, thus enabling successful training convergence. This improvement is

TABLE VI: Performance comparison with existing approaches.

Approach	RMSE per timestamp	RMSE aggregate	MAE per timestamp	MAE aggregate	MAPE	R^2
I. Ullah et al. [4]	N.A.	16.34	N.A.	13.93	N.A.	N.A.
S. Modi et al. [3]	0.54	15.95	0.064	6.17	0.057	0.9116
Proposed method	0.48	9.12	0.063	3.08	0.046	0.9261

TABLE VII: Summary of model performance across various scenarios. Note: Highlighted cells in the $Model\_des$ column signify performance rankings, with colors representing the spectrum from best to worst as follows: green (best), yellow (moderate), and red (worst). Highlighted cells in the $MAPE$ column denote the criteria for selecting the best model. Blue-highlighted cells indicate differential responses under FGSM attack. Cells with N.A. in the R^2 column refer to the score is outside the range of 0 to 1.

Output	Model_des	Condition	Eval	RMSE		MAE		MAPE	R^2
				Per Time Evaluation	Aggregate Evaluation	Per Time Evaluation	Aggregate Evaluation		
Single	MAE_Integral	normal	mean	169.6388196	13043.14403	41.91151404	4105.452005	74.04004456	N.A.
			median	26.22239112	1724.089513	17.26880891	1724.089513	8.862396655	
		noisy	mean	169.6388196	13043.14403	41.91151404	4105.452005	74.04004456	N.A.
			median	26.22239112	1724.089513	17.26880891	1724.089513	8.862396655	
		FGSM	mean	169.6388196	13043.14403	41.91151404	4105.452005	74.04004456	N.A.
			median	26.22239112	1724.089513	17.26880891	1724.089513	8.862396655	
		FGSM and noisy	mean	169.6388196	13043.14403	41.91151404	4105.452005	74.04004456	N.A.
			median	26.22239112	1724.089513	17.26880891	1724.089513	8.862396655	
	MAE_NO_Integral	normal	mean	0.539116557	15.94549755	0.06399502	6.167096541	0.056394986	0.9063
			median	0.139748483	2.448822784	0.026804085	2.448822476	0.044250499	
		noisy	mean	0.539116556	15.94549742	0.063995016	6.167096486	0.056394947	0.9063
			median	0.139748483	2.448822784	0.026804088	2.448822696	0.044250499	
		FGSM	mean	0.539116557	15.94549755	0.06399502	6.167096541	0.056394986	0.9063
			median	0.139748483	2.448822784	0.026804085	2.448822476	0.044250499	
		FGSM and noisy	mean	1.970469106	36.00059964	0.06399502	6.167096543	0.056394986	N.A.
			median	0.142002451	2.471036652	0.026804085	2.448822476	0.044250499	
	MSE_Integral	normal	mean	169.329374	12974.65965	63.80273372	6027.101327	297.7113501	N.A.
			median	77.83471144	5524.22187	57.64675905	5524.221457	107.8782035	
		noisy	mean	169.329374	12974.65965	63.80273372	6027.101327	297.7113501	N.A.
			median	77.83471144	5524.22187	57.64675905	5524.221457	107.8782035	
		FGSM	mean	169.329374	12974.65965	63.80273372	6027.101327	297.7113501	N.A.
			median	77.83471144	5524.22187	57.64675905	5524.221457	107.8782035	
		FGSM and noisy	mean	169.329374	12974.65965	63.80273372	6027.101327	297.7113501	N.A.
			median	77.83471144	5524.22187	57.64675905	5524.221457	107.8782035	
	MSE_NO_Integral	normal	mean	6459185.329	105669307.7	4818865.686	98208283.4	46475695.07	N.A.
			median	6027451.364	98903083.82	4866134.538	99316792.01	47594285.91	
		noisy	mean	6459185.329	105669307.7	4818865.686	98208283.4	46475695.07	N.A.
			median	6027451.364	98903083.82	4866134.538	99316792.01	47594285.91	
		FGSM	mean	6459185.329	105669307.7	4818865.686	98208283.4	46475695.07	N.A.
			median	6027451.364	98903083.82	4866134.538	99316792.01	47594285.91	
		FGSM and noisy	mean	6459185.329	105669307.7	4818865.686	98208283.4	46475695.07	N.A.
			median	6027451.364	98903083.82	4866134.538	99316792.01	47594285.91	
	MSE_Integral	normal	mean	0.497530931	92.72131436	0.178895554	90.4412094	1.236067327	0.9202
			median	0.183097631	76.06265407	0.139082501	76.06265259	1.179076462	
		noisy	mean	0.497530931	92.72131436	0.178895554	90.4412094	1.236067327	0.9202
			median	0.183097631	76.06265407	0.139082501	76.06265259	1.179076462	
		FGSM	mean	0.497530931	92.72131436	0.178895554	90.4412094	1.236067327	0.9202
			median	0.183097631	76.06265407	0.139082501	76.06265259	1.179076462	
		FGSM and noisy	mean	0.497530931	92.72131436	0.178895554	90.4412094	1.236067327	0.9202
			median	0.183097631	76.06265407	0.139082501	76.06265259	1.179076462	
	MSE_NO_Integral	normal	mean	34.79818801	284.4204874	14.55894265	150.7833399	140.0155646	N.A.
			median	1.742048669	17.98862561	0.690691881	17.98859211	5.639758539	
		noisy	mean	33.39583942	272.6513777	13.35429384	139.7387359	128.365177	N.A.
			median	1.686370201	17.13726633	0.640824662	16.64647094	5.174663638	
		FGSM	mean	33726.11504	150536.9036	10734	56209	103539	N.A.
			median	527.7092002	2979.915435	355	3198	3443	
		FGSM and noisy	mean	32350.28927	139502.4719	9314	45773	89840	N.A.
			median	470.0744622	2790.603161	311	2762	3012	
Parallel	MAE_Integral	normal	mean	0.480659852	9.197762127	0.076678439	5.496336633	0.190942271	0.9255
			median	0.090650609	4.19404978	0.023917858	4.194049835	0.19489	
		noisy	mean	0.480659852	9.197762127	0.076678439	5.496336633	0.190942271	0.9255
			median	0.090650609	4.19404978	0.023917858	4.194049835	0.19489	
		FGSM	mean	0.480659852	9.197762127	0.076678439	5.496336633	0.190942271	0.9255
			median	0.090650609	4.19404978	0.023917858	4.194049835	0.19489	
		FGSM and noisy	mean	0.480659852	9.197762127	0.076678439	5.496336633	0.190942271	0.9255
			median	0.090650609	4.19404978	0.023917858	4.194049835	0.19489	
	MAE_NO_Integral	normal	mean	0.478487502	9.120882889	0.062757357	3.083468013	0.046344769	0.9261
			median	0.083946527	0.635489055	0.014632967	0.640154745	0.037457418	
		noisy	mean	0.478487502	9.120882889	0.062026751	3.083433281	0.046344769	0.9261
			median	0.083871685	0.635489055	0.013081913	0.635486096	0.037457418	
		FGSM	mean	0.478487502	9.121503188	0.062757357	3.083468013	0.046344769	0.9261
			median	0.083946527	0.640176876	0.014632967	0.640154745	0.037457418	
		FGSM and noisy	mean	0.478487502	9.121503188	0.062757357	3.083468013	0.046344769	0.9261
			median	0.083946527	0.640176876	0.014632967	0.640154745	0.037457418	

TABLE VIII: Summary of model performance skewness across various scenarios. Note: traffic light indicator in front of the number indicate the skewness ranking, with colors representing the spectrum from best to worst as follows: green (best), yellow (moderate), and red (worst).

Output	model_des	Condition	RMSE Per Time Evaluation	RMSE Aggregate Evaluation	MAE Per Time Evaluation	MAE Aggregate Evaluation	MAPE
Single	MAE_Integral	normal	5.469235348	6.565235989	1.427006649	1.381229033	7.354040282
		noisy	5.469235348	6.565235989	1.427006649	1.381229033	7.354040282
		FGSM	5.469235348	6.565235989	1.427006649	1.381229033	7.354040282
		FGSM and noisy	5.469235348	6.565235989	1.427006649	1.381229033	7.354040282
	MAE_NO_Integral	normal	2.857763206	5.511495096	1.387509977	1.518392657	0.274448597
		noisy	2.857763199	5.511495043	1.38750951	1.518392408	0.274447713
		FGSM	2.85776319	5.511495043	1.387509977	1.518392657	0.27444977
		FGSM and noisy	2.857763206	5.511495096	1.387509978	1.518392657	0.27444977
Parallel	MSE_Integral	normal	1.717298562	0.219012346	0.2862594	0.1830957	0.048335174
		noisy	1.717298562	0.219012346	0.2862594	0.1830957	0.048335174
		FGSM	1.717298562	0.219012346	0.2862594	0.1830957	0.048335174
		FGSM and noisy	1.717298562	0.219012346	0.2862594	0.1830957	0.048335174
	MAE_Integral	normal	4.302334486	1.1930503	2.205907432	0.310508184	0.020256189
		noisy	4.302334486	1.1930503	2.205907432	0.310508184	0.020256189
		FGSM	4.302334486	1.1930503	2.205907432	0.310508184	0.020256189
		FGSM and noisy	4.302334486	1.1930503	2.205907432	0.310508184	0.020256189
Parallel	MAE_NO_Integral	normal	4.699908217	13.35254128	3.288764991	3.816754129	0.23726546
		noisy	4.704994469	13.35254128	3.741412797	3.8852086142	0.23726546
		FGSM	4.704994469	13.35254128	3.288764991	3.816754129	0.23726546
		FGSM and noisy	4.699908217	13.24841092	3.288764991	3.816754129	0.23726546

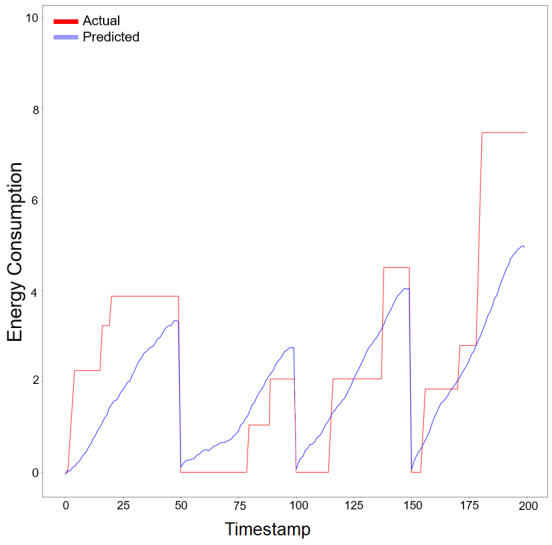


Fig. 4: Illustration showing predicted value failing to catch up with the speed of increment.

due to the substantial penalties applied during iterations with predominantly zero outputs. While the cumulative sum method promotes training convergence and increases the model's resilience to adversarial inputs, it slightly reduces accuracy. This reduction in accuracy occurs because the cumulative output rises sharply with significant target values, leading the network to produce smoother outputs. This, in turn, impedes the model's ability to precisely track incremental changes, as illustrated in Figure 4. We also observed an overfitting issue during the model improvement process in our preliminary work [19]. To address this, we adopted an early-stop strategy and conducted three additional rounds of training with a randomly split dataset to confirm that the overfitting issue was resolved.

In the context of FGSM attacks, a notable escalation in the mean MSE evaluation was exclusively observed for the MAE_NO_Integral model (highlighted in blue), suggesting that only a subset of the attacks led to erroneous predictions. Given the model's susceptibility to adversarial attacks, we can infer that both the cumulative sum methodology and the parallel transformation layer contribute positively in combatting adversarial instances. Table VIII presents the percentage disparity between mean and median values; a more pronounced divergence signifies a higher skew in the accuracy distribution, thereby implying reduced stability in the model's output. By analyzing the performance trajectory, it is evident that in instances where the model exhibits acceptable performance (as listed in Table VIII, excluding MAE_integral), the

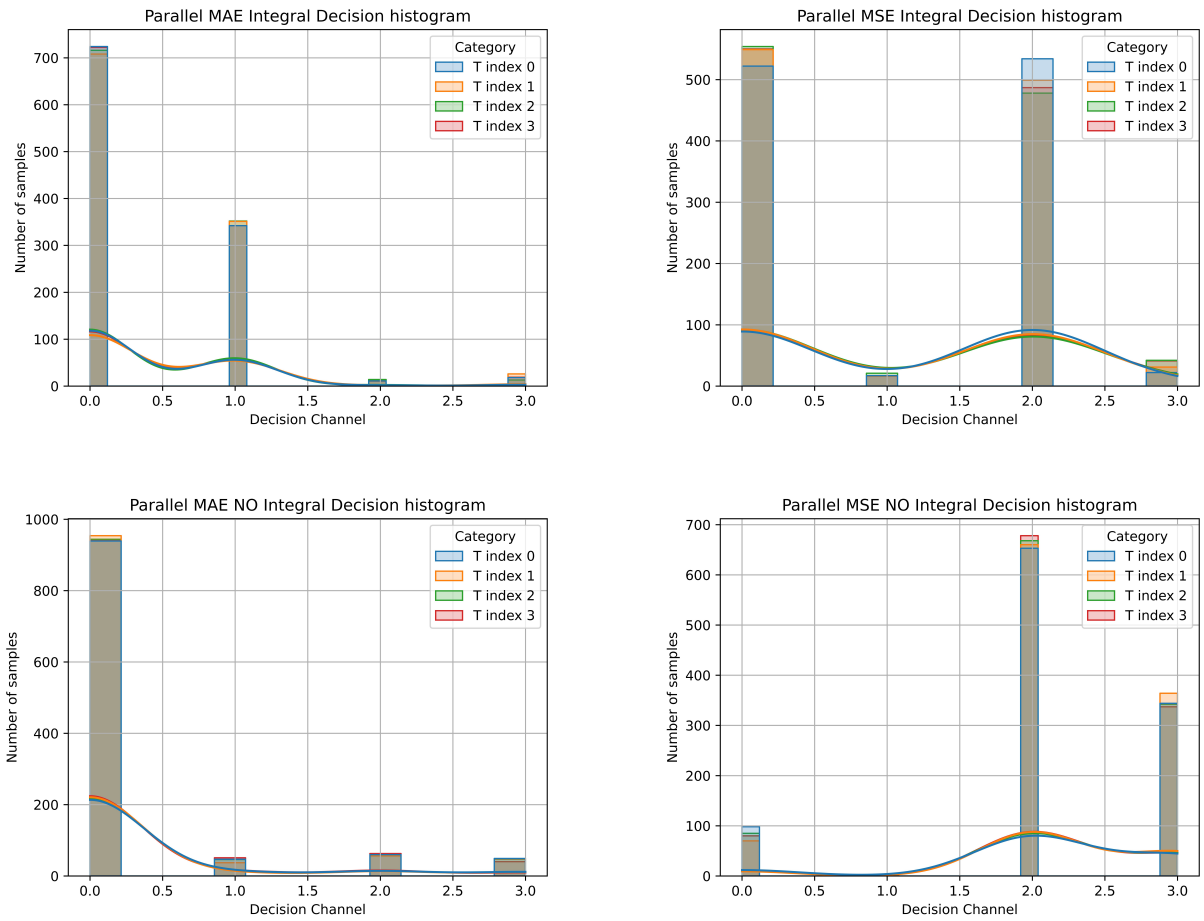
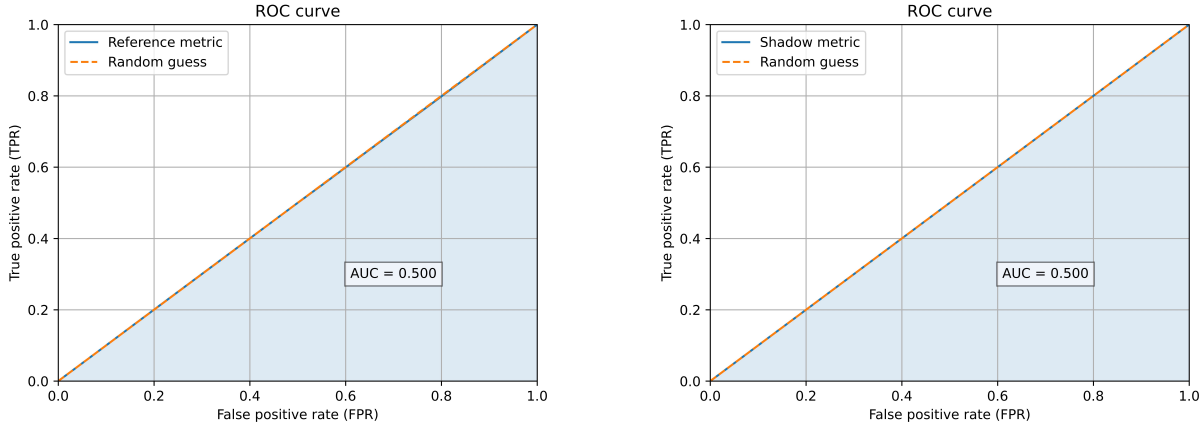


Fig. 5: Histograms of channel responses across all multi-output models, with legends denoting the corresponding actual channels of each record.



(a) Result of Population inference attack.

(b) Result of membership inference attack.

Fig. 6: ROC and AUX evaluation for population inference attack and membership inference attack.

employment of a cumulative sum strategy potentially bolsters the model's performance consistency.

C. Privacy

Algorithm 3: Algorithm of transformation index inference attack

Attacker has a set of driving record D

create data set metric list **for** d_i *in* D **do**
  create metric list **for** c *in* $range(4)$ **do**
    `// c is guessed channel number`
    Initialize dummy input with size (200,9)
    $dummy[c*50:(c+1)*50,:] = d_i$
    obtain MSE, MAE and MAPE at channel output
  end
  append MSE, MAE, MAPE **if** *Decision with one record* **then**
    Index = argmin(MSE or MAE or MAPE)
  else
    append to data set metric list
  end
end
if d_i *in* D *belongs to the same model* **then**
  Index = argmin(mean(data set metric))
else
end

The proposed transformation layer is partitioned into four sub-regions, each corresponding to a different car model. This structure inherently poses a potential vulnerability: an adversary could attempt to infer whether a specific driving record is associated with a particular car model. This type of attack, known as a transformation index inference attack, operates under the assumption that the attacker possesses certain driving records. The attacker's strategy involves padding these records with zeros at the beginning and end, effectively manipulating the record's position within the transformation layer. By subsequently analyzing the model's output, the attacker aims to determine whether the manipulated record corresponds to the guessed car model. This determination is based on a comparison of channel evaluation metrics, as described in Algorithm 3. Here, the term "channel" refers to the transformation index used to allocate records to the appropriate sub-region. To assess the resilience of our model against such an attack, we simulated this scenario using driver records sampled from the test dataset. We then plotted a histogram map of the channel responses, counting the number of records classified under each channel for all multi-output models. Figure 5 summarizes the model's behavior when different driver records, encoded to the presumed channel, were introduced. Notably, the trained model exhibited a uniform channel response, unaffected by the absence of record information from other sub-regions. This outcome demonstrates that the proposed method maintains its functional integrity while safeguarding against transformation index inference attacks.

Moreover, we extended our privacy analysis by adopting the methodology presented in [29]. This was employed to evaluate the best performed model that is trained with MAE as the objective function and no cumulative sum implementation at its output. This comprehensive evaluation encompassed two inference attacks:

- *Population inference attack:* This type of attack aims to deduce sensitive information about a population subset based on aggregated data. The attacker leverages the model's output to make generalized inferences about characteristics or patterns common to a group, without pinpointing individual members.
- *Membership inference attack:* In contrast, this attack seeks to ascertain whether a specific individual's data was included in the training dataset used by the machine learning model. The attacker uses the model's predictions, combined with knowledge of the target's characteristics, to infer their presence in the dataset.

Both evaluations employed the receiver operating characteristic curve (ROC) and Area Under the ROC Curve (AUC) as metrics to gauge the accuracy of the attacker's guesses regarding data belonging to the targeted member group. Remarkably, the AUC evaluation under both scenarios yielded a score of 0.5 (as depicted in Figure 6), indicating an equivalent chance of a random guess. This result underscores the model's robustness in protecting user privacy against such inference attacks.

V. Conclusion

This paper presented a method for predicting electric vehicle energy consumption. The primary objective was to delve into driver behavior analytics and harness this information to forecast energy requirements accurately, employing a novel computational model for this purpose.

One of the key innovation of the proposed system is its unique ability to categorize input data into distinct segments based on car models and restructure the data utilizing the 2D-DCT technique. This transformation layer is pivotal, offering the dual advantages of parallel processing capabilities and ensuring the independent analysis of each data segment, thereby preserving the unique behavioral patterns inherent to each. Our analysis shows that employing the MAE as the objective function during the training phase offered superior resilience in scenarios characterized by data noise and potential adversarial attacks, outperforming the MSE based implementation. This resilience is crucial in real-world applications where data irregularities are the norm rather than the exception. Furthermore, the robustness of the proposed system was demonstrated against three sophisticated inference attacks, simulating scenarios that an adversary might employ in attempts to breach data privacy. This resilience against inference attacks not only underscores the system's robust design but also its suitability for real-world application scenarios where data security is of paramount importance.

VI. Acknowledgment

This research was supported in part by Singapore Ministry of Education Academic Research Fund Tier 1 Grants R-263-000-E78-114 and R-263-001- E78-114.

References

[1] "Global co2 emissions from transport by sub-sector in the net zero scenario, 2000-2030." https://www.iea.org/data-and-statistics/charts/global-co2-emissions-from-transport-by-sub-sector-in-the-net-zero-scenario-2000-2030-2, accessed: 2023-2-30.

[2] B. O. Varga, A. Sagoian, and F. Mariasiu, "Prediction of electric vehicle range: A comprehensive review of current issues and challenges," *Energies*, vol. 12, no. 5, 2019. [Online]. Available: https://www.mdpi.com/1996-1073/12/5/946

[3] S. Modi, J. Bhattacharya, and P. Basak, "Estimation of energy consumption of electric vehicles using deep convolutional neural network to reduce driver's range anxiety," *ISA transactions*, vol. 98, pp. 454–470, 2020.

[4] I. Ullah, K. Liu, T. Yamamoto, R. E. A. Mamlook, and A. Jamal, "A comparative performance of machine learning algorithm to predict electric vehicles energy consumption: A path towards sustainability," *Energy & Environment*, vol. 33, no. 8, pp. 1583–1612, 2022. [Online]. Available: https://doi.org/10.1177/0958305X211044998

[5] M. Al-Rubaie and J. M. Chang, "Privacy-preserving machine learning: Threats and solutions," *IEEE Security & Privacy*, vol. 17, no. 2, pp. 49–58, 2019.

[6] O. Travesset-Baro, M. Rosas-Casals, and E. Jover, "Transport energy consumption in mountainous roads. a comparative case study for internal combustion engines and electric vehicles in andorra," *Transportation Research Part D: Transport and Environment*, vol. 34, pp. 16–26, 2015.

[7] A. I. Croce, G. Musolino, C. Rindone, and A. Vitetta, "Traffic and energy consumption modelling of electric vehicles: Parameter updating from floating and probe vehicle data," *Energies*, vol. 15, no. 1, p. 82, 2022.

[8] X. Wu, D. Freese, A. Cabrera, and W. A. Kitch, "Electric vehicles' energy consumption measurement and estimation," *Transportation Research Part D: Transport and Environment*, vol. 34, pp. 52–67, 2015.

[9] R. Zhang and E. Yao, "Electric vehicles' energy consumption estimation with real driving condition data," *Transportation Research Part D: Transport and Environment*, vol. 41, pp. 177–187, 2015.

[10] M. Koengkan, J. A. Fuinhas, M. Belucio, N. K. Alavijeh, N. Salehnia, D. Machado, V. Silva, and F. Dehdar, "The impact of battery-electric vehicles on energy consumption: A macroeconomic evidence from 29 european countries," *World Electric Vehicle Journal*, vol. 13, no. 2, 2022.

[11] X. Qi, G. Wu, K. Boriboonsomsin, and M. J. Barth, "Data-driven decomposition analysis and estimation of link-level electric vehicle energy consumption under real-world traffic conditions," *Transportation Research Part D: Transport and Environment*, vol. 64, pp. 36–52, 2018, the contribution of electric vehicles to environmental challenges in transport. WCTRS conference in summer.

[12] C. De Cauwer, W. Verbeke, T. Coosemans, S. Faid, and J. Van Mierlo, "A data-driven method for energy consumption prediction and energy-efficient routing of electric vehicles in real-world conditions," *Energies*, vol. 10, no. 5, 2017.

[13] K. Liu, J. Wang, T. Yamamoto, and T. Morikawa, "Exploring the interactive effects of ambient temperature and vehicle auxiliary loads on electric vehicle energy consumption," *Applied Energy*, vol. 227, pp. 324–331, 2018, transformative Innovations for a Sustainable Future – Part III.

[14] K. Liu, T. Yamamoto, and T. Morikawa, "Impact of road gradient on energy consumption of electric vehicles," *Transportation Research Part D: Transport and Environment*, vol. 54, pp. 74–81, 2017.

[15] A. D. Alvarez, F. S. Garcia, J. E. Naranjo, J. J. Anaya, and F. Jimenez, "Modeling the driving behavior of electric vehicles using smartphones and neural networks," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 3, pp. 44–53, 2014.

[16] J. Felipe, J. C. Amarillo, J. E. Naranjo, F. Serradilla, and A. Díaz, "Energy consumption estimation in electric vehicles considering driving style," in *2015 IEEE 18th international conference on intelligent transportation systems*. IEEE, 2015, pp. 101–106.

[17] A. Skuza and R. Jurecki, "Analysis of factors affecting the energy consumption of an ev vehicle-a literature study," in *IOP Conference Series: Materials Science and Engineering*, vol. 1247, no. 1. IOP Publishing, 2022, p. 012001.

[18] W. Liao, S. Wang, B. Bak-Jensen, J. R. Pillai, Z. Yang, and K. Liu, "Ultra-short-term interval prediction of wind power based on graph neural network and improved bootstrap technique," *Journal of Modern Power Systems and Clean Energy*, vol. 11, no. 4, pp. 1100–1114, 2023.

[19] X. Hu and B. Sikdar, "Energy consumption prediction of electrical vehicles through transformation of time series data," in *2023 IEEE 3rd International Conference on Sustainable Energy and Future Electric Transportation (SEFET)*, 2023, pp. 1–7.

[20] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," 2017.

[21] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2015.

[22] C. Gaete-Morales, H. Kramer, W.-P. Schill, and A. Zerrahn, "An open tool for creating battery-electric vehicle time series from empirical data, emobpy," *Scientific Data*, vol. 8, no. 1, p. 152, 2021.

[23] "Kuhnimhof, T. & Nobis, C. mobilität in deutschland – mid: Ergebnisbericht." https://elib.dlr.de/125879/, accessed: 2023-3-5.

[24] Z. Wang, T. Oates *et al.*, "Encoding time series as images for visual inspection and classification using tiled convolutional neural networks," in *Workshops at the twenty-ninth AAAI conference on artificial intelligence*, vol. 1. AAAI Menlo Park, CA, USA, 2015.

[25] X. Hu and B. Sikdar, "Sub-group based machine learning for gas consumption prediction," in *Proc. IEEE CSDE*, 2021, pp. 1–6.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[27] E. A. Smirnov, D. M. Timoshenko, and S. N. Andrianov, "Comparison of regularization methods for imagenet classification with deep convolutional neural networks," *AASRI Procedia*, vol. 6, pp. 89–94, 2014, 2nd AASRI Conference on Computational Intelligence and Bioinformatics.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[29] S. K. Murakonda and R. Shokri, "Ml privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning," 2020.