J. Cumplido, C. Alcaraz, and J. Lopez, "Collaborative anomaly detection system for charging stations", *The 27th European Symposium on Research in Computer Security (ESORICS 2022)* vol. 13555, pp. 716736, 2022.
http://doi.org/https://doi.org/10.1007/978-3-031-17146-8_35
NICS Lab. Publications: https://www.nics.uma.es/publications

Collaborative anomaly detection system for charging stations

Jesus Cumplido, Cristina Alcaraz, and Javier Lopez

Computer Science Department, University of Malaga,

Campus de Teatinos s/n, 29071, Malaga, Spain

{cumplido, alcaraz, jlm}@lcc.uma.es

Abstract

In recent years, the deployment of charging infrastructures has been increasing exponentially due to the high energy demand of electric vehicles, forming complex charging networks. These networks pave the way for the emergence of new unknown threats in both the energy and transportation sectors. Economic damages and energy theft are the most frequent risks in these environments. Thus, this paper aims to present a solution capable of accurately detecting unforeseen events and possible fraud threats that arise during charging sessions at charging stations through the current capabilities of the Machine Learning (ML) algorithms. However, these algorithms have the drawback of not fitting well in large networks and generating a high number of false positives and negatives, mainly due to the mismatch with the distribution of data over time. For that reason, a Collaborative Anomaly Detection System for Charging Stations (here referred to as CADS4CS) is proposed as an optimization measure. CADS4CS has a central analysis unit that coordinates a group of independent anomaly detection systems to provide greater accuracy using a voting algorithm. In addition, CADS4CS has the feature of continuously retraining ML models in a collaborative manner to ensure that they are adjusted to the distribution of the data. To validate the approach, different use cases and practical studies are addressed to demonstrate the effectiveness and efficiency of the solution.

Keywords: Collaborative anomaly detection Charging station Machine Learning Voting system

1 Introduction

According to studies carried out in [1], more than 30 million Electric Vehicles (EVs) are predicted to be on the roads by 2030. This encourages organizations to deploy a large number of charging infrastructures in order to meet the energy demand expected from EV batteries. Charging infrastructures are commonly composed of a set of interconnected Charging Stations (CSs), which are remotely controlled by a control system, called CS Management System (CSMS)

[2]. These infrastructures often use Information Technologies (ITs) as well as Operational Technologies (OTs) to provide the system with greater functionality and intelligence, such as online reservations, payments through bank entities and monitoring of charging profiles from external entities, known as Energy Management Systems (EMSs). Further details about the design of charging station infrastructure can be found in Appendix A. However, this convergence of ITs and OTs to create complex networks leads to new cybersecurity risks in power systems [3]. Standards institutions, such as the National Institute of Standards and Technology (NIST), and other governmental organizations of interest, such as the United States (US) Department of Energy, Transportation and Defense, are raising concerns about these new environments under development [4]. In this report, NIST highlighted how CSs are bringing two critical sectors together for the first time: energy and transportation, which have never been electronically connected before. This implies new potential attacks that could directly impact financial terms, business continuity and human safety.

This concern is accompanied by the observed increase in cybercrime attacks on critical infrastructures according to the latest European Union Agency for Cybersecurity (ENISA) threat landscape report [5], where the major critical infrastructure sectors being impacted are healthcare, transportation and energy. Recently, several researchers have developed a novel attack called BrokenWire [6] against rapid chargers with the ability to wirelessly send malicious signals to the targeted vehicle in order to cause electromagnetic interference and disrupt the charging session. In [7], a botnet of compromised EVs and CSs is also launched to simultaneously attack the proper functioning of the power grid by increasing its load in an uncontrolled manner; consequently provoking a Denial of Service (DoS). Also, the authors of [8] show the feasibility of extracting charging session attributes and creating large datasets to lead privacy issues. Many of these threats are also contemplated in [9, 10, 2, 3], where the authors show how charging infrastructures are susceptible to diverse threats. To clarify the influence of these attacks and its impact to the sector, Appendix A details the most common threats to CS components and communications, as well as the highest risk impacts, corresponding to economic damage and energy theft. As stated in [11], the main security weakness is due to the type of CS deployment in public environments and the type of communication, which can be wireless. Consequently, different cybersecurity expert organizations have been working to provide solutions to these threats. Most organizations rely on frameworks and standards to help ensure a structured defense of control systems [12], where detection is a core element.

One of the most widespread detection solutions in the literature is the Anomaly Detection System (ADS) based on Machine Learning (ML) algorithms [13]. These systems are responsible for identifying deviations or outliers, denominated as anomalies, and launch an early alert when these are detected. One of the main advantages of these algorithms is their ability to learn and adapt to the data distribution, thanks to their ability to recognize both known events and unknown anomalies (e.g. those caused by zero-day vulnerabilities). However, in systems such as charging infrastructures, the distribution of data tends

to vary rapidly over time due to their continuous increase in energy demand and the improvements in EV batteries and charging speed. This causes an increase in the False Positive (FP) and False Negative (FN) ratio in ML models by mismatching with the distribution of the data. In addition, charging networks are currently composed of groups of interconnected CSs building up a complex network, usually distributed by zone and managed locally by a CSMS. This distribution also presents the challenge of sharing anomalies and alerts between the independent charging infrastructures, for the purpose of detecting distributed attacks and obtaining global information on the situational awareness of the charging network.

To provide a suitable solution to these challenges, researchers apply Collaborative ADS (CADS) as a protection measure of large networks and large IT ecosystems [14]. CADS is based on the cooperation of different monitors distributed in the system, which act as sensors and collect data. It also contains one or more analysis units that are responsible for intrusion detection by correlating data obtained from sensors. These defensive supports have encouraged us to **contribute with** a novel approach using a centralized CADS – referred to here as CADS for CSs (CADS4CS) – as an optimization measure to adjust detection algorithms according to the real conditions of charging networks. To do so, CADS4CS is composed of a central analysis unit, referred to here as a coordinator, which coordinates a distributed group of standalone ML-based ADSs to: (1) *obtain higher accuracy in anomaly detection using a voting algorithm*; and (2) *continuously retrain ML models in a collaborative and secure manner to ensure that they are always adjusted to the data distribution.* To provide an optimal voting system for CSs, we designed three types of coordinators with three voting algorithms based on: (1) average; (2) weighted average; and (3) mode. Based on this, we conducted several experiments representing various anomaly detection scenarios in charging networks. These scenarios correspond to the use of: (1) a single charging session dataset, which includes the same type of threats and is shared between the different ADSs to validate the performance of ML models; and (2) different datasets, each of which contemplates different anomalies in order for each ADS to validate the effectiveness of the voting algorithms.

This paper is organized as follows. Section 2 summarizes all work related to ML-based anomaly detection on energy consumption in CSs. Section 3 describes the structure and functionality of CADS4Cs. More specifically, Subsection 3.1 defines the open charging session datasets and the types of anomalies, while Subsection 3.2 establishes the design of the central coordinator together with the types of voting algorithms proposed. Subsequently, Section 4 shows the results of different analyses and experiments on various use cases. Finally, Section 5 draws the conclusion from the results obtained and describes future work.

Table 1: Related work on Machine Learning-based anomaly detection systems

Reference (year)	Applied technique	Learning process	Method	Scenario	Dataset	Energy consumption-based	Collaborative
[15] (2014)	Time series	Unsupervised	Clustering	Building energy consumption	Real data	✓	X
[16] (2019)	RNN	Deep Learning	Regression	Tennessee Valley Authority	Data provided by 30 power meters	✓	X
[17] (2018)	SVM, KNN, Random Forest	Supervised	Classification	Water supply system	Testbed	✓	X
[18] (2018)	TMSE	Supervised	Classification	Industrial Internet of Things	Dataset offered by State Grid of China	✓	X
[19] (2020)	Regression trees	Supervised	Regression	Smart Grid AMI	Experiment	✓	X
[20] (2020)	K-means LSTM	Unsupervised, Deep Learning	Clustering, forecasting	User power consumption	Dataport, a public dataset	✓	X
[21] (2020)	KNN	Supervised	Classification	Charging Station	-	X	X
[22] (2019)	Moving Average, DBSCAN	Unsupervised	Clustering	Charging Station	High frequency harmonic data	X	X
[23] (2020)	Neural Networks, LSTM	Supervised, Deep Learning	Classification	Charging Station	CICIDS 2018 DoS dataset	X	X
[24] (2021)	MHA	Supervised, Deep Learning	Classification	Charging Station	Laboratory, network traffic	X	X
[25] (2021)	KNN, SVM, Random Forest	Supervised	Classification	Charging Station	Experiment	✓	X
CADS4CS (our approach)	Collaborative System	Supervised, Deep Learning	Classification	Charging Station	Open data	✓	✓

2 Related work

In the literature, there are several recent scientific works proposing different solutions for ADSs in industrial and cyber-physical environments. Table 1 shows a summary of the related works, which are associated with ML-based ADSs in energy environments, such as anomalies in energy consumption or CSs.

In [15], Janetzko et al. introduce an anomaly detection algorithm based on time series to detect and visualize unexpected power consumptions in commercial buildings, and then use clustering techniques to classify them. Similarly in [16], the authors study the use of deep learning algorithms, such as Recurrent Neural Networks (RNNs), to remove trend and seasonality from time series data and predict the power anomalies. Other works, related to energy anomaly monitoring and detection, are [17, 18]. Robles-Durazno et al. in [17] propose a supervised learning model for energy monitoring and anomaly detection in a clean water supply system, using classifiers such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Random Forest. In [18], another ML model for detecting energy anomalies is studied by Ouyang et al., where the Three-stage Multi-view Stacking Ensemble (TMSE) model is proposed to detect anomalous power consumption in industrial devices. In addition, other works comprise the ability to predict outliers in the energy. For instance, in [19] a two-level anomaly detection framework based on regression decision trees is proposed with the objective of predicting unexpected power consumption in an Advanced Metering Infrastructure (AMI). The combination of clustering and prediction techniques, such as the K-Means and Long-Short Term Memory (LSTM) techniques, is even analyzed in [20] to predict the power consumption of users in the next hour.

There are also several recent studies on the use of ADSs in CS scenarios. In [21] and [24], anomalous traffic data within the network is identified. An invariant-correlation network and a multivariate time-series segmentation method using the KNN classifier is used in [21], while a Multi-Head Attentions (MHA) model is used in [24] to correlate the network traffic. As an alternative,

Streubel et al. in [22] adapt the identification of irregular patterns in the high harmonic frequency spectrum, described by the CS supraharmonic emissions, and group the detected anomalies according to similar characteristics using the technique known as Density-Based Spatial Clustering of Applications with Noise (DBSCAN). Deep learning models have also been considered in [23] for early detection of DoS attacks against CSs.

Finally, the work in [25] performs a threat detection analysis based on power consumption through ML techniques, which are able to classify three types of states in each CS: normal, risk or accident. As can be seen in the Table 1, [25] is the only work that addresses an ADS based on the CS energy consumption. It is focused on the detection of malfunctioning attacks in the CSs that may lead to a DoS of these systems. In contrast, our approach differs from these works in the use of a collaborative system as an optimization measure for intrusion detection in complex charging networks. Although collaborative intrusion detection systems is not a novel approach [26, 14, 27], its applicability in charging infrastructure environments is. CADS4CS has the ability to detect and learn from different types of power consumption anomalies at CSs, specifically those related to consumption energy deviations in the charging sessions.

3 CADS4CS: datasets and architecture

This section covers the functionalities of the CADS4CS, starting with the definition of the datasets and types of anomalies used in each of the ADSs, and ends with the CADS4CS design and the types of voting algorithms of the coordinator.

3.1 Data models, datasets and anomalies

To gain a correct understanding of the energy data, it is necessary to analyze the behavior of the data distribution and features of the user charging sessions, which usually include the following attributes: (i) total energy consumed (in kWh), (ii) cost or fee, (iii) charge duration, (iv) session duration, (v) type of connector used and (vi) charging speed. Different data models, with derived attributes, have been created from these attributes to train ML models and obtain a high accuracy of anomaly detection at CSs, regardless of their manufacturing model, configuration or the region in which they are located.

We have considered several open charging session databases (dated between 2017 and 2022), whose information comes from different geographic locations and charging networks. These databases correspond with: Boulder [28] and Palo Alto [29] cities in the US; Dundee city [30] and Perth and Kinross council [31] in Scotland, United Kingdom (UK); and charging sessions from the ElaadNL network in the Netherlands [32]. For each of these, the data have been processed and cleaned to a common format, maintaining the aforementioned attributes. In addition to this, we have generated charging session anomalies related to errors or intentional attacks on energy consumption values in the datasets mentioned, which could have a significant impact on the meaning of the monitoring actions

and decision-making. To establish these anomalies, we identified two types of perturbations that affect to the charging session data: *measurement reading errors* or *deliberated attacks* such as *false data injection* or *modification*. These anomalies can influence the following attributes: (1) energy consumed, (2) session duration, (3) charge duration, (4) average power, (5) total cost and (6) no charge (energy consumed is 0); note that these attributes have been considered according to the common charging session features of all selected datasets. In order to understand the perturbation procedure, the anomalies were intentionally injected into the datasets following a random strategy. That is, for each dataset, approximately 20% of samples were extracted, which were intentionally perturbed in some of their attributes in a random manner. Therefore, normal and anomalous samples have been explicitly labeled by us for subsequent studies.

In turn, the previous datasets are applied to form a network of distributed CS clusters, as illustrated in Figure 1. Each cluster contains its own dataset and ADS for local anomaly detection. However, these ADSs have the added problem of being susceptible to increasing the number of FPs and FNs over time due to their mismatch with the data distribution, or due to their inability to detect some unknown or stealthy threats [33]. To avoid this issue, CADS4CS deals with a solution based on a higher-level CADS, which is defined below.

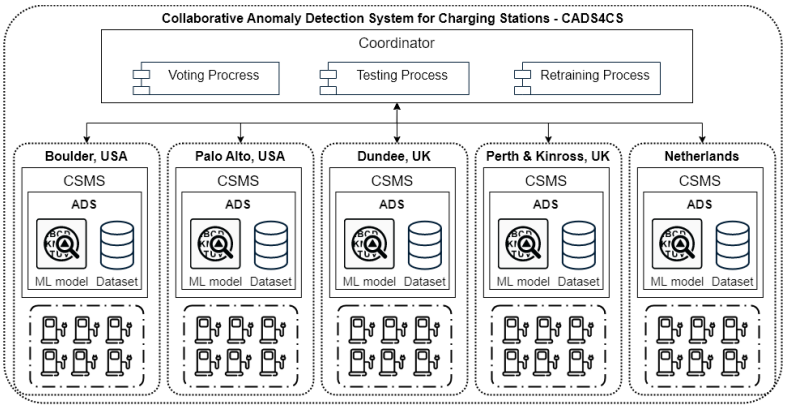

Figure 1: CADS4CS design

3.2 Collaborative anomaly detection system

CADS4CS is based on establishing a centralized node in the CS network that acts as coordinator of the entire charging network. This coordinator is in charge of communicating with each of the ADSs in each CS cluster and collecting predictions, alerts and performance information from each of them. Therefore, the main goal of these coordinators is to detect anomalies at a global level through a simple voting system based on the local predictions, thereby achieving a lower FP and FN ratio than the local ADSs. To do this, another objective is

Algorithm 1 Coordinator Model: voting, testing and retraining

```
Require: adsModels
  procedure VOTING(Xsamples)                          ▷ Unknown charging session samples
    yLocalPreds ⇐ EmptyList()
    for ads ← adsModels do
      yLocalPreds.add(ads.PREDICT(Xsamples))
    end for
    yGlobalPred ⇐ CalculateGlobalPrediction(yLocalPreds)          ▷ Statistical function
    for ads ← adsModels do   ▷ Add Xsamples and yGlobalPred as label to the train dataset
      ads.ADDSAMPLE(Xsamples, yGlobalPred))
    end for
  end procedure
  procedure TESTING(Xsamples, Ysamples)               ▷ Known charging session samples
    yLocalPreds ⇐ EmptyList()
    for ads ← adsModels do
      yLocalPreds.add(ads.PREDICT(Xsamples))
      ads.EVALUATELOCALPREDICTION(yLocalPred, Ysamples)
      ads.ADDSAMPLE(Xsamples, Ysamples)               ▷ Add samples to the train dataset
    end for
    yGlobalPred ⇐ CalculateGlobalPred(yLocalPreds)          ▷ Statistical function
    EVALUATEGLOBALPREDICTION(yGlobalPred, Ysamples)
  end procedure
  procedure RETRAINING
    for ads ← adsModels do
      ads.RETRAIN
    end for
  end procedure
```

to develop continuous retraining measures in a collaborative manner to optimize the performance of the ML models of each local ADS.

These two objectives are carried out thanks to three processes incorporated in the coordinator (as defined in Algorithm 1): (1) *voting process*, where the coordinator evaluates the predictions of each local ADS on an unknown anomaly triggered by one of them, thereby deriving a global prediction using a statistical function, as detailed below; (2) *testing process*, where the coordinator evaluates the performance of each local ADS after generating different samples of normal/anomalous charging sessions, referred here as "tests"; and (3) *retraining process*, where the coordinator sends the order to retrain each of the ML models of each local ADS. To understand its functionality, the voting process and the different types of coordinators are described below.

3.2.1 Voting process:

this process consists in detecting anomalies based on the predictions made by each of the local ADSs, thus obtaining a global prediction from a statistical function (average, weighted-average or mode), as specified in Algorithm 1 (VOTING procedure). Initially, each local ADS individually predicts the charging session samples that are recorded in its CSs. After a local ADS predicts a possible anomaly, it is notified to the coordinator who is in charge of starting the voting phase. The coordinator forwards the received sample to the other local ADSs to make their own local prediction. Note that these ADSs do not know the origin of the sample or whether it corresponds to a sample from another local ADS, or if it is a test generated by the coordinator. The predictions made by the local ADSs are returned to the coordinator, which calculates a global prediction from

a statistical function. Finally, the global prediction serves as a labeled sample that is stored in the ADS datasets for future retraining.

Three types of coordinators have been implemented according to the statistical function used to calculate the global prediction.

- **Average Coordinator (ACoord.)**: the global prediction is calculated as the arithmetic average of the local probability predictions together with a predefined threshold (α). If the average probability obtained is greater than α, the sample is considered to be an anomaly.
- **Weighted-Average Coordinator (WACoord.)**: similarly, the global prediction is calculated as the weighted average of the local probability predictions together with an α. The weights of each local prediction are determined by the F1-score performance metric of each ADS obtained in its last evaluation. Thus, ADSs with optimal ML models will be more heavily weighted than ADSs with worse performing ML models.
- **Mode Coordinator (MCoord.)**: in this case, the thresholds are automatically defined by each ML model and they directly return a discrete prediction, which corresponds to the label 0 if it is considered a normal sample, or 1 if it is an anomaly. This coordinator simply considers the label that appears most often (i.e. the absolute majority) to be the global prediction.

Note that the average and weighted average have been selected based on the correlation of opinions established in [34]. In addition, we have extended the research by also using the mode as a correlation function, which computes discrete values and does not require defining the prediction threshold (alpha) by the coordinator. Following the classification given in [27] and [14], our approach corresponds to a mix between the similarity-based and filter-based approaches. In the remaining sections, we therefore provide a comprehensive analysis, showing the behavior of these types of correlations for different use cases.

4 Analyses, experiments and results

Based on the aforementioned open datasets and the possible charging session anomalies described in Section 3.1, two types of practical analysis are carried out on the CADS4CS:

- **Analysis 1 (A1) − using a shared dataset**: this corresponds to using the same training and testing dataset shared between the different ADSs, with the objective of analyzing the performance of each ML model and the coordinator to detect known anomalies in all ADSs. Note that the shared dataset contains all types of charging session anomalies proposed in Section 3.1. In a real environment, this analysis is useful in scenarios where the same charging network with a centralized and shared dataset incorporates different ADSs to detect possible anomalies.

- **Analysis 2 (A2) − using incomplete datasets**: consists in using different training and test datasets for each of the ADSs of the collaborative system. This analysis simulates the use case of a real scenario where one or more charging networks separate the management and anomaly detection by groups of CSs, where each group contains its own charging session database and ADS (see Figure 1). In addition, each one may be prone to certain types of anomalies and may be unaware of the anomalies of other CS clusters. To achieve this, we have desegregated each dataset to be incomplete, not incorporating all the types of anomalies, as shown in Table 2.

Table 2: Summary of known anomalies in each dataset for **A2**

Anomaly / Dataset	Energy	Charge Duration	Session Duration	Power	Cost	No Charge
Boulder, US	✓	X	X	✓	✓	X
Palo Alto, US	✓	✓	✓	X	X	X
Dundee, UK	X	✓	✓	X	X	✓
Perth and Kinross, UK	✓	X	X	✓	✓	✓
Netherlands	X	✓	✓	✓	✓	X

In order to make a comprehensive study of the performance of the coordinators in different situations, we performed two types of experiments for each of the analyses. These experiments are as follows:

- **Experiment 1 (E1) − based on three sequential phases**: this is a simple procedure to evaluate the performance of each type of coordinator before and after a retraining of the ML models. For this purpose, a set of charging session samples (including all types of anomalies and ordered chronologically) is initially split into 150 sets. After each split (each subset of samples), F1-score metric of each of the ADSs and coordinators is calculated in order to discern the most optimal algorithm.

  For **E1**, we consider three application phases, as shown in Algorithm 2 − Experiment 1 procedure: (1) *pre-retraining voting phase*, where each ADS predicts the first 50 splits and the coordinators, based on the predictions of the ADSs, compute their global predictions using their corresponding statistical function; (2) *retraining phase*, during the next 50 splits, each ADS predicts the samples, adds the sample to its dataset and retrains its ML model with the training dataset updated so far; and (3) *post-retraining voting phase*, where again the last 50 splits are predicted locally by the ADSs and the coordinators compute the global predictions.

- **Experiment 2 (E2) − based on two cyclic phases**: this is a methodology where voting, testing and retraining processes are continuously executed. It is based on two cyclic phases, which alternate during the 150 sets of samples splits, as shown in Algorithm 2 − Experiment 2 procedure.

  More specifically, **E2** includes: (1) *voting phase*, where each ADS and the coordinator collaboratively predict a subset of samples that is finally

added to the training set of each ML model, using the coordinator's global predictions as labels (as indicated in the VOTING procedure of Algorithm 1); and (2) *testing phase*, where each ADS predicts a subset of samples again, but this time adds the original samples, with the real labels, to the training set (as indicated in the TESTING procedure of Algorithm 1). After the completion of each phase, the ML models are retrained using their own updated training dataset (with global prediction labels and real test labels). Note that **E2** aims to assess whether continuous retraining of the ML models is required, interleaving the voting process and the testing process.

4.1 A1: analyzing CADS4CS using a shared dataset

A1 intends to validate the behavior of the coordinators and ADSs when databases are shared for **E1** and **E2**, in addition to plotting the learning result of ML models that best fit the types of perturbations. Table 3 shows the ML classifiers chosen for each ADS: Decision Trees (DT), Random Forest (RF); CatBoost [35], eXtreme Gradient Boosting (XGBoost) [36], and Multi-Layer Perceptron (MLP). We have chosen these classifiers because they have shown the best results in terms of efficiency and accuracy, as also reflected in the following studies [37] [38], [39] and [40]. It is important to note that for **A1** and **A2**, we have

Algorithm 2 Experiments: **E1** and **E2**

```
Require: coord, X, Y
  Xsplits, Ysplits ⇐ SPLIT(X, Y, 150)
  procedure EXPERIMENT 1
    for i ← 1 to 50 do                                   ▷ Pre-Retraining Voting Phase
      Xsamples, Ysamples ← Xsplit[i], Ysplit[i]
      yGlobalPred ← coord.VOTING(Xsamples)
      EVALUATEF1SCORE
    end for
    for i ← 51 to 100 do                                 ▷ Retraining Phase
      Xsamples, Ysamples ← Xsplit[i], Ysplit[i]
      yGlobalPred ← coord.TESTING(Xsamples, Ysamples)
      EVALUATEF1SCORE
      coord.RETRAINING
    end for
    for i ← 51 to 150 do                                 ▷ Post-Retraining Voting Phase
      Xsamples, Ysamples ← Xsplit[i], Ysplit[i]
      yGlobalPred ← coord.VOTING(Xsamples)
      EVALUATEF1SCORE
    end for
  end procedure
  procedure EXPERIMENT 2
    for i ← 1 to 150 do
      Xsamples, Ysamples ← Xsplit[i], Ysplit[i]
      if i is odd then                                   ▷ Voting Phase
        yGlobalPred ← coord.VOTING(Xsamples)
      else                                               ▷ Testing Phase
        yGlobalPred ← coord.TESTING(Xsamples, Ysamples)
      end if
      EVALUATEF1SCORE
      coord.RETRAINING
    end for
  end procedure
```

established $\alpha = 0.4$ as the anomaly probability threshold. This value is pre-established as the optimum found after several studies with different thresholds.

Table 3: Features of the ADSs in **A1** and **A2**

	Analysis 1		**Analysis 2**		**Machine Learning**
	Dataset (full)	**Size**	**Dataset (incomplete)**	**Size**	
ADS1	Dundee	200K	Boulder	40K	CatBoost
ADS2			Dundee	180K	DT
ADS3			Netherlands	12K	MLP
ADS4			Palo Alto	200K	RF
ADS5			Perth & Kinross	80K	XGBoost

4.1.1 E1 – based on three sequential phases:

Figures 2 and 3 show the evolution of the F1-score metric for both individual ADSs and for each of the coordinator classes. The results clearly illustrate the best ML models with higher precision and recall, as well as the usefulness of the coordinators in this type of scenario. More specifically, after the retraining phase, slight improvements (approximately an increase of F1 score between 0.01 and 0.04) can be observed in the performance of the ADSs and coordinators, due to the fact that the ADSs have been initially trained with a shared dataset with all anomalies (cf. Section 3.1).

From Figure 2, we highlight how the ML models of ADS1 and ADS5, corresponding to the Catboost and XGBoost, show significantly better results providing a 0.9 F1 score in the best case. This is followed by ADS2 and ADS4, corresponding to the use of DT and RF classifiers, with a 0.82 score, and finally ADS3 (corresponding to the MLP model) presents the worst results with a 0.78 score. In this first experiment, we can observe how the ML models based on decision trees, such as DT, RF, CatBoost and XGBoost, are good classifiers for the aforementioned dataset and the perturbations given in it. They are able to train and quickly detect large deviations in the normal distribution of the dataset, as stated in [38]. Moreover, the CatBoost and XGBoost models are even better since they share the use of an efficient and effective implementation of the gradient boosting algorithm to obtain an optimal classifier based on decision trees. In contrast, the neural network used by the MLP classifier has not been able to fit correctly with the data distribution, resulting in a high number of FPs and FNs.

As can be seen in Figure 2, the types of coordinators present similar results to the ADSs. ACoord. and MCoord. return an evolution curve slightly inferior to ADS1 and ADS5. While WACoord, which uses the F1-score metric of the ADSs as weights, performs similarly to the best ADSs, such as ADS1 and ADS5. For such scenarios, WACoord. can be useful to ensure that the coordinator's detection has as low an FP and FN ratio as possible.

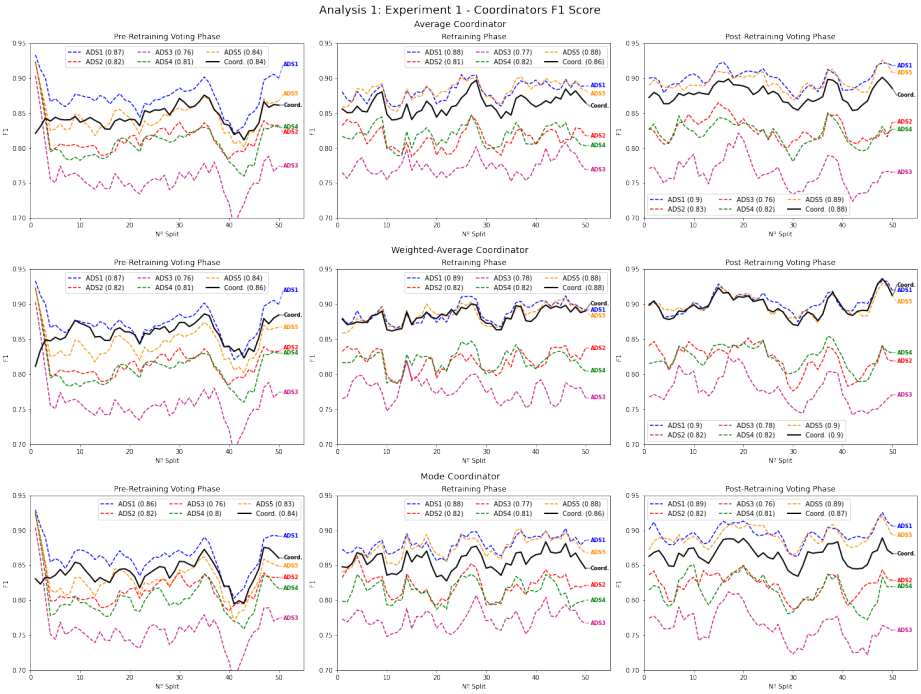

Figure 2: F1-score evolution of all ADSs in each coordinator during **A1-E1**

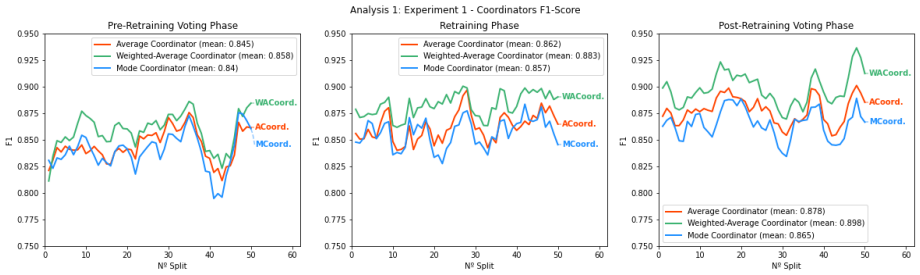

Figure 3: F1-score evolution in each coordinator during **A1-E1**

4.1.2 E2 – based on two cyclic phases:

in this experiment, we can observe results very similar to those obtained in **E1**, but with slight improvements, as shown in Figures 4 and 5 (from 0.81 to 0.93 F1-score). Both ADSs and coordinators achieve an increase in the mean F1-score by approximately 3 hundredths with respect to the **E1** results, particularly in the case of the coordinators. For each type of coordinator, as shown in Figure 5, there is a slight positive trend in the evolution of F1 over time (splits), which implies a continuous improvement and adjustment of the ML models with the

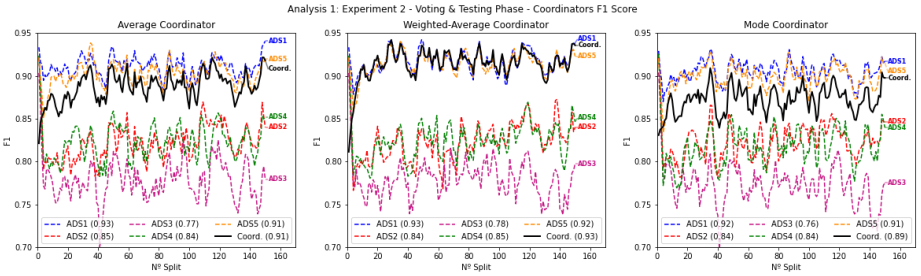

Figure 4: F1-score evolution of all ADSs in each coordinator during **A1-E2**

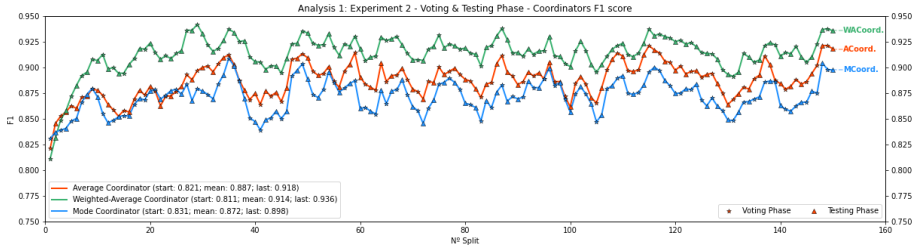

Figure 5: F1-score evolution in each coordinator during **A1-E2**

distribution of the dataset. This means that the use of the combination of global predictions with the tests generated by the coordinator as new training samples are useful for detecting future known and unknown anomalies.

4.2 A2: analyzing CADS4CS using incomplete datasets

In this second analysis, each ADS has a different and incomplete dataset, as shown in Table 3. This scenario corresponds to more faithful use cases in practice in real environments, where a charging network is further divided into different groups of CSs according to certain parameters, such as location, model or manufacturer. This distribution facilitates the management of each group, which would have their own CSMS, EMS, security policies, protocols and, above all, their own charging session dataset and ADS. Therefore, the aim of this analysis is to check if the coordinator succeeds in detecting any type of anomaly, which may be unknown to some of the ML models.

4.2.1 E1 – based on three sequential phases:

as shown in Figure 6, in the pre-retraining voting phase, ADSs present results with low precision (F1-score below 0.5), which suggests the detection of a large number of FPs and FNs. This is because the ADSs have initially been trained with incomplete datasets and their ML models are unable to detect certain types of unknown anomalies. However, the different types of coordinators are

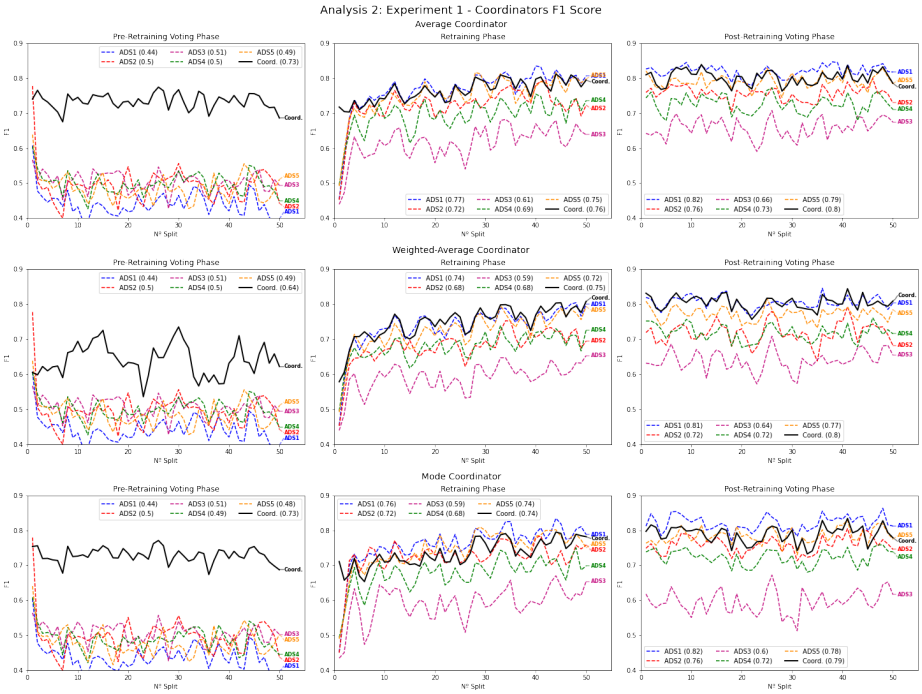

Figure 6: F1-score evolution of all ADSs in each coordinator during **A2-E1**

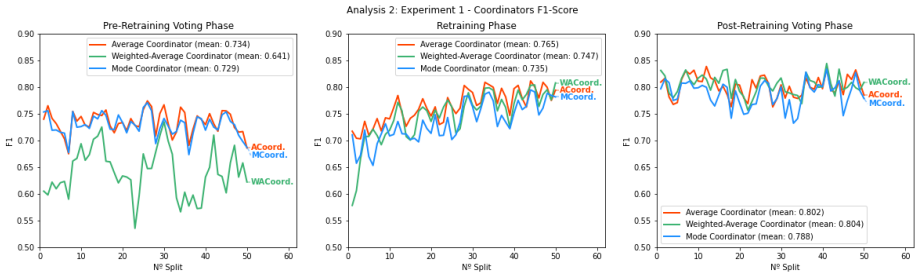

Figure 7: F1-score evolution in each coordinator during **A2-E1**

able to unify the individual detections of the ADSs and provide more accurate predictions, thus obtaining a higher F1-score evolution than the ADSs. In this case, as shown in Figure 7, ACoord. and MCoord. present better results (above 0.7); while WACoord., due to its high dependence on the F1-score of the ADSs that are used as weights in the weighted average, results in a higher number of FPs and FNs.

However, these results change completely at the end of the retraining phase, where in this case, all ML models are retrained with the samples obtained by the coordinators' global prediction during the pre-retraining voting phase and

the tests generated by the coordinators during the retraining phase. Therefore, in the last phase (post-retraining voting phase), a significant improvement in the F1-score evolution of the ADSs and coordinators is evident, since the ML models are now able to detect unknown anomalies from their datasets. After the retraining, ADS1 and ADS5, corresponding to the Catboost and XGBoost ML models, remain optimal models with mean F1 scores of 0.82 and 0.78, respectively. This is followed by ADS2 (using DT classifier) and ADS4 (using RF classifier) with scores of 0.76 and 0.72. Finally, ADS3, which uses a Deep Learning model such as MLP, presents significantly lower precision and recall than the rest with an average F1 score of 0.64. As in the previous analysis, the three types of coordinators present similar results to the best ADSs, where ACoord. and WACoord. are the optimum. This feature is also depicted in Figure 7.

4.2.2 E2 – based on two cyclic phases:

in this experiment, a continuous retraining is again performed combining the voting and testing phases. The purpose of this experiment is to observe whether the ADSs succeed in optimizing their ML parameters in a collaborative way, thanks to the global predictions and tests generated by the coordinator. As shown in Figure 8 and 9, an F1-score evolution with positive trend is achieved for both local ADSs and the various coordinators. With this, we observe that the results obtained are significantly higher than the **E1** results after the retraining phase, especially for the ACoord. and WACoord., which achieve an F1-score of almost 0.85 in the last splits.

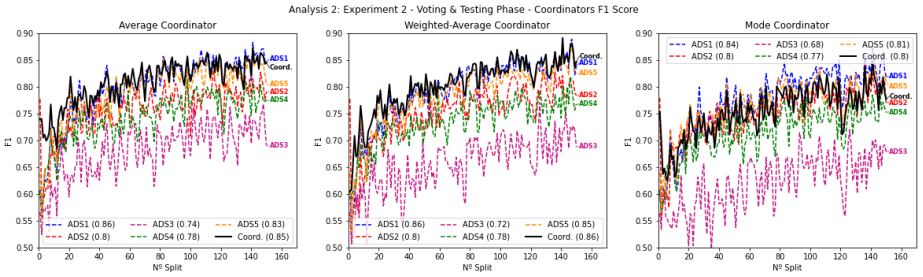

Figure 8: F1-score evolution of all ADSs in each coordinator during **A2-E2**

4.3 Discussions: A1 vs A2

Table 4 summarizes the mean F1-score for each ADS and coordinator and for each analysis and experiment. From this table, we first observe that ML models generated using the gradient boosting technique (such as CatBoost and XGBoost) correspond to the optimal ML models. These models show high precision in detecting the types of charging session anomalies discussed in this

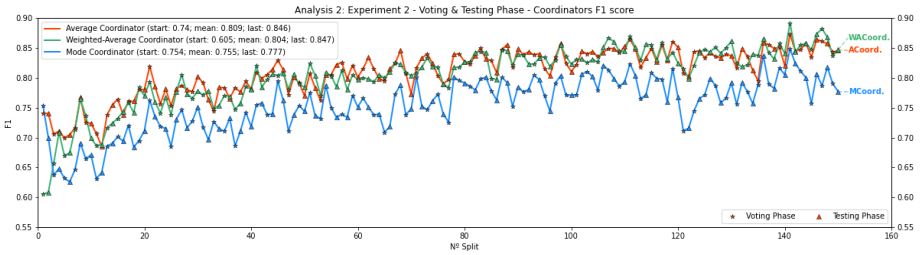

Figure 9: F1-score evolution in each coordinator during **A2-E2**

paper. Specifically, the Catboost algorithm, used in **ADS1**, corresponds to the optimum ML model, achieving an F1-score of up to 0.93 in **A1-E2**.

On the other hand, we can see how the suggested types of coordinators can be useful in certain use cases. ACoord. and MCoord. are quite useful in scenarios where the performance of each ADS is unknown or their ML models are incapable of detecting certain types of unknown anomalies, as occurs in **A2-E1** in the pre-retraining voting phase (Figure 6); while WACoord. is particularly useful when combined with a continuous retraining of the ML models and testing phases, as also observed in **A1-E2** and **A2-E2**. In Table 4, we can appreciate how WACoord. presents the optimal F1-scores reaching the value of 0.936 in **A1-E2**. This coordinator presents better results compared to the average and mode coordinators, because its correlation prioritizes the predictions of the ADSs with greater precision. Thus, the rate of FPs and FNs is reduced.

Finally, from this table we can also conclude that **E2** offers better performance compared to **E1**. This means that it is advisable in these use cases to follow a continuous retraining methodology where voting and testing processes are alternately combined. This allows the ML models to dynamically adjust to the data distribution, obtaining higher precision over time. However, looking at the results in Table 4 we also highlight that there are still too many FPs and FNs in this scenario, which can saturate human operators. This implies that it is still necessary to advance in this line of research, addressing new solutions to classify anomalies by optimizing existing detection methods.

Table 4: Summary of F1-score results

		ADS1	**ADS2**	**ADS3**	**ADS4**	**ADS5**	**ACoord.**	**WACoord.**	**MCoord.**
A1	**E1**	**0.90**	0.82	0.78	0.82	**0.90**	0.878	**0.898**	0.865
	E2	**0.93**	0.84	0.78	0.85	0.92	0.918	**0.936**	0.898
A2	**E1**	**0.82**	0.76	0.64	0.72	0.78	0.802	**0.804**	0.788
	E2	**0.86**	0.8	0.72	0.78	0.85	0.846	**0.847**	0.777

The best performing method in every experiment is marked in bold

5 Conclusion and future work

In this paper, we have carried out a comprehensive analysis of the detection of charging session anomalies in different types of charging stations (slow, fast and rapid charging mode). Firstly, we have collected different open data of charging sessions to later simulate various errors and threats, generating anomalies and deviations in different attributes of the sessions. We have then defined a collaborative anomaly detection system capable of coordinating and retraining a group of independent Machine Learning models. Finally, after performing different analyses and experiments, we have observed how in certain scenarios the collaborative system is successful in achieving a high F1-score with a low false positive and negative ratio, and in establishing an effective procedure for continuous retraining of the ML models. Of all the ML models trained and evaluated, decision tree variants – such as random forest classifier and gradient boosting techniques (CatBoost and XGBoost algorithms) – are the optimal models in these cases. As future work, we intend to extend the approach considering the actual drawbacks of the collaborative detection systems, such as data privacy and trust as stated in [26] and [27]; and integrate the approach in a real charging infrastructure within the "Smart and Secure EV Urban Lab II" project.

5.0.1 Acknowledgements

This work has been supported by the "Smart and Secure EV Urban Lab II" through the Second Own Plan of Smart-Campus of the University of Malaga, by the EC under the SealedGRID project (H2020-MSCA-RISE-2017) with GA no. 777996, by the Ministry of Science and Innovation under SECUREDGE project (PID2019-110565RB-I00 − AEI/10.13039/501100011033/),
and by the Andalusian Government under the SAVE project (P18-TP-3724).

A Design and threats of a public charging infrastructure

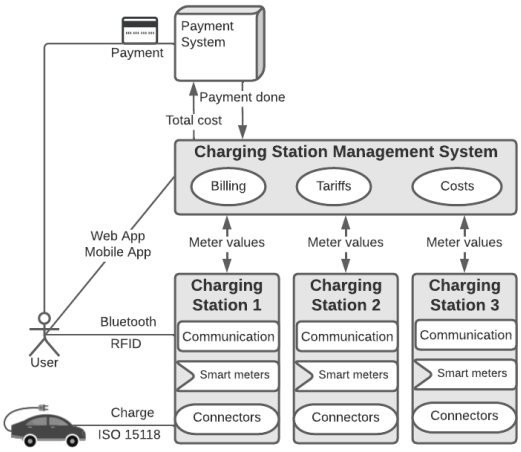

Figure 10: Deployment diagram of a public charging infrastructure

This appendix provides an overview of the components that compose a charging infrastructure and clarifies the level of susceptibility of these infrastructures to attacks according to the state of the art. Public CSs are usually managed by a CSMS, which has the ability to use ITs and OTs to efficiently control each CS and its charging sessions initialized by the users. Specifically, this control center is in charge of authenticating, authorizing and billing users, and diagnosing. Figure 10 shows a generic public charging infrastructure based on the Open Charge Point Protocol (OCPP) standard [41].

The combination of ITs and OTs in these cyber-physical systems leads to new security risks that must be considered right from the design stage. Above all, the addition of new functionalities, communications and external actors in the charging infrastructures open the door to new threats to the system. For this reason, we include in this appendix a high-level review of the state of the art [42, 43, 44] to show the susceptibility of this infrastructure to attacks and their impact on the end user and the power grid. Among the most common threats are: (T1) natural disasters, (T2) physical damage, (T3) DoS, (T4) identity theft or spoofing, (T5) malware injection, (T6) false data injection, (T7) tampering and (T8) sniffing or information disclosure.

Table 5 shows a summary of these threats with their corresponding environmental, social and economic impacts. As can be seen in the table, blackouts, economic damages and energy theft correspond to the impacts with the greatest likelihood and risk in a public charging infrastructure. This work therefore aims to mitigate these impacts through the use of Machine Learning techniques

Table 5: Summary of threats and impacts on a public charging infrastructure

Threat \ Impact	Blackout	Energy theft	Equipment damage	Economic damage	Data leak
T1	✓	X	✓	✓	X
T2	✓	X	✓	✓	X
T3	✓	X	X	✓	X
T4	X	✓	X	✓	✓
T5	✓	✓	✓	✓	✓
T6	X	✓	X	✓	X
T7	✓	✓	✓	✓	X
T8	X	X	X	X	✓

for anomaly detection. Its scope has been limited to the detection of threats related to T6 and T7, and focuses on studying the normal behavior of energy consumption data.

References

[1] Deloitte. Electric vehicle trends | Deloitte Insights. visited on 2022-05-18.

[2] Joseph Antoun, Mohammad Ekramul Kabir, Bassam Moussa, Ribal Atallah, and Chadi Assi. A Detailed Security Assessment of the EV Charging Ecosystem. *IEEE Network*, 34(3):200–207, 2020.

[3] Zoya Pourmirza and Sara Walker. Electric Vehicle Charging Station: Cyber Security Challenges and Perspective. *9th IEEE International Conference on Smart Energy Grid Engineering, SEGE*, pages 111–116, 2021.

[4] Suzanne Lightman and Tanya Brewer. Symposium on Federally Funded Research on Cybersecurity of Electric Vehicle Supply Equipment (EVSE). 2020.

[5] ENISA. *ENISA Threat Landscape 2021.* 2021.

[6] Sebastian Köhler, Richard Baker, Martin Strohmeier, and Ivan Martinovic. Brokenwire : Wireless Disruption of CCS Electric Vehicle Charging, 2022. visited on 2022-05-25.

[7] Omniyah Gul M Khan, Ehab El-Saadany, Amr Youssef, and Mostafa Shaaban. Impact of electric vehicles botnets on the power grid. In *IEEE Electrical Power and Energy Conference*, pages 1–5. IEEE, 2019.

[8] Alessandro Brighente, Mauro Conti, Denis Donadel, and Federico Turrin. EVScout2. 0: Electric vehicle profiling through charging profile. *arXiv preprint arXiv:2106.16016*, 2021.

[9] Juan E. Rubio, Cristina Alcaraz, and Javier Lopez. Addressing Security in OCPP: Protection Against Man-in-The-Middle Attacks. *9th IFIP International Conference on New Technologies, Mobility and Security, NTMS 2018 - Proceedings*, pages 1–5, 2018.

[10] Panda Security. Electric vehicle charging stations are vulnerable to hacker attacks, 2022. visited on 2022-05-03.

[11] Cristina Alcaraz, Javier Lopez, and Stephen Wolthusen. OCPP Protocol: Security Threats and Challenges. *IEEE Transactions on Smart Grid*, 8(5):2452–2459, 2017.

[12] Mark Bristow. A SANS Survey: OT/ICS Cybersecurity. pages 1–23, 2021.

[13] Weiyu Zhang, Qingbo Yang, and Yushui Geng. A survey of anomaly detection methods in networks. *Proceedings - 1st International Symposium on Computer Network and Multimedia Technology, CNMT*, pages 10–12, 2009.

[14] Emmanouil Vasilomanolakis, Shankar Karuppayah, Max Muhlhauser, and Mathias Fischer. Taxonomy and survey of collaborative intrusion detection. *ACM Computing Surveys*, 47(4):1–33, 2015.

[15] Halldór Janetzko, Florian Stoffel, Sebastian Mittelstädt, and Daniel A. Keim. Anomaly detection for visual analytics of power consumption data. *Computers and Graphics (Pergamon)*, 38(1):27–37, 2014.

[16] Keith Hollingsworth, Kathryn Rouse, Jin Cho, Austin Harris, Mina Sartipi, Sevin Sozer, and Bryce Enevoldson. Energy Anomaly Detection with Forecasting and Deep Learning. *Proceedings - IEEE International Conference on Big Data, Big Data*, pages 4921–4925, 2018.

[17] Andres Robles-Durazno, Naghmeh Moradpoor, James McWhinnie, and Gordon Russell. A supervised energy monitoring-based machine learning approach for anomaly detection in a clean water supply system. *International Conference on Cyber Security and Protection of Digital Services, Cyber Security*, pages 1–8, 2018.

[18] Zhiyou Ouyang, Xiaokui Sun, Jingang Chen, Dong Yue, and Tengfei Zhang. Multi-View Stacking Ensemble for Power Consumption Anomaly Detection in the Context of Industrial Internet of Things. *IEEE Access*, 6:9623–9631, 2018.

[19] Amara Korba Abdelaziz, Nouredine Tamani, Yacine Ghamri-Doudane, and Nour El Islem Karabadji. Anomaly-based framework for detecting power overloading cyberattacks in smart grid AMI. *Computers and Security*, 96:101896, 2020.

[20] C. Chahla, H. Snoussi, L. Merghem, and M. Esseghir. A deep learning approach for anomaly detection and prediction in power consumption data. *Energy Efficiency*, 13(8):1633–1651, 2020.

[21] Yu Wei Chung, Mervin Mathew, Cole Rodgers, Bin Wang, Behnam Khaki, Chicheng Chu, and Rajit Gadh. The framework of invariant electric vehicle charging network for anomaly detection. *IEEE Transportation Electrification Conference and Expo, ITEC*, pages 631–636, 2020.

[22] Tim Streubel, Christoph Kattmann, Adrian Eisenmann, and Krzysztof Rudion. Detection and monitoring of supraharmonic anomalies of an electric vehicle charging station. *IEEE Milan PowerTech, PowerTech*, pages 1–5, 2019.

[23] Manoj Basnet and Mohd Hasan Ali. Deep learning-based intrusion detection system for electric vehicle charging station. *2nd International Conference on Smart Power and Internet Energy Systems, SPIES*, pages 408–413, 2020.

[24] Yidong Li, Li Zhang, Zhuo Lv, and Wei Wang. Detecting Anomalies in Intelligent Vehicle Charging and Station Power Supply Systems with Multi-Head Attention Models. *IEEE Transactions on Intelligent Transportation Systems*, 22(1):555–564, 2021.

[25] Yanjie Li, Xiaoyu Ji, Dongxiao Jiang, and Tao Men. Abnormal Detection System Design of Charging Pile Based on Machine Learning. *IOP Conference Series: Earth and Environmental Science*, 772(1):0–5, 2021.

[26] Wenjuan Li, Weizhi Meng, and Lam For Kwok. Surveying trust-based collaborative intrusion detection: State-of-the-art, challenges and future directions. *IEEE Communications Surveys & Tutorials*, 24(1):280–305, 2021.

[27] Chenfeng Vincent Zhou, Christopher Leckie, and Shanika Karunasekera. A survey of coordinated attacks and collaborative intrusion detection. *Computers & Security*, 29(1):124–140, 2010.

[28] Open-Data Boulder Colorado. Electric Vehicle Charging Station Energy Consumption, 2021. visited on 2022-05-08.

[29] City of Palo Alto. Electric Vehicle Charging Station Usage (July 2011 - Dec 2020) · Open Data · City of Palo Alto, 2021. visited on 2022-05-08.

[30] Drive Dundee Electric. Electric Vehicle Charging Sessions Dundee - Datasets, 2019. visited on 2022-05-08.

[31] Perth & Kinross Council. Electric Vehicle Charging Station Usage - Datasets - Perth and Kinross - Open Data, 2021. visited on 2022-05-08.

[32] Elaad NL. Data delen @ Elaad NL, 2021. visited on 2022-05-08.

[33] Lorena Cazorla, Cristina Alcaraz, and Javier Lopez. Cyber Stealth Attacks in Critical Information Infrastructures. *IEEE Systems Journal*, 12:1778–1792, 2018.

[34] Juan E. Rubio, Mark Manulis, Cristina Alcaraz, and Javier Lopez. Enhancing security and dependability of industrial networks with opinion dynamics. In *European Symposium on Research in Computer Security (ESORICS 2019)*, volume 11736, pages 263–280, 09/2019 2019.

[35] Yandex. CatBoost - open-source gradient boosting library. visited on 2022-05-22.

[36] XGBoost. XGBoost Documentation — xgboost 1.6.0 documentation. visited on 2022-05-22.

[37] Moloud Abdar, Neil Yuwen Yen, and Jason Chi-Shun Hung. Improving the diagnosis of liver disease using multilayer perceptron neural network and boosted decision trees. *Journal of Medical and Biological Engineering*, 38(6):953–965, 2018.

[38] Cristina Alcaraz, Lorena Cazorla, and Gerardo Fernandez. Context-awareness using anomaly-based detectors for smart grid domains. In *9th International Conference on Risks and Security of Internet and Systems*, volume 8924, pages 17–34, Trento, 04/2015 2015. Springer International Publishing, Springer International Publishing.

[39] Mukesh Kumar Mishra and Rajashree Dash. A comparative study of chebyshev functional link artificial neural network, multi-layer perceptron and decision tree for credit card fraud detection. In *2014 International Conference on Information Technology*, pages 228–233, 2014.

[40] Sohrab Mokhtari, Alireza Abbaspour, Kang K. Yen, and Arman Sargolzaei. A machine learning approach for anomaly detection in industrial control systems based on measurement data. *Electronics*, 10(4), 2021.

[41] Open Charge Alliance. OCPP 2.0.1, 2020. visited on 2022-05-24.

[42] Raju Gottumukkala, Rizwan Merchant, Adam Tauzin, Kaleb Leon, Andrew Roche, and Paul Darby. Cyber-physical System Security of Vehicle Charging Stations. *IEEE Green Technologies Conference*, 2019.

[43] Narayan Bhusal, Mukesh Gautam, and Mohammed Benidris. Cybersecurity of Electric Vehicle Smart Charging Management Systems. *52nd North American Power Symposium, NAPS*, 2020.

[44] Farzam Nejabatkhah, Yun Wei Li, Hao Liang, and Rouzbeh Reza Ahrabi. Cyber-Security of Smart Microgrids: A Survey. *Energies*, 14(1):27, 2020.