Annals of Telecommunications (2024) 79:537–551
https://doi.org/10.1007/s12243-024-01011-x

Towards feature engineering for intrusion detection in IEC–61850 communication networks

Vagner E. Quincozes[1] · Silvio E. Quincozes[2,3] · Diego Passos[1,4] · Célio Albuquerque[1] · Daniel Mossé[5]

Received: 5 June 2023 / Accepted: 4 January 2024 / Published online: 3 February 2024

Abstract
Digital electrical substations are fundamental in providing a reliable basis for smart grids. However, the deployment of the IEC–61850 standards for communication between intelligent electronic devices (IEDs) brings new security challenges. Intrusion detection systems (IDSs) play a vital role in ensuring the proper function of digital substations services. However, the current literature lacks efficient IDS solutions for certain classes of attacks, such as the masquerade attack. In this work, we propose the extraction and correlation of relevant multi-layer information through a feature engineering process to enable the deployment of machine learning-based IDSs in digital substations. Our results demonstrate that the proposed solution can detect attacks that are considered challenging in the literature, attaining an F1-score of up to 95.6% in the evaluated scenarios.

Keywords Feature extraction · Intrusion detection systems (IDSs) · Machine learning (ML) · Digital substations · IEC–61850

1 Introduction

Digital electrical substations are key elements of smart grids, where devices from different manufacturers have to communicate with each other. To achieve this goal, several protocols have emerged. The IEC–61850 [12] standard is one initiative that defines important rules for the interoperability of communication between intelligent electronic devices (IEDs). Nowadays, the IEC–61850 standards are still evolving to facilitate communication in substation networks [32].

While enhanced communication support enables novel applications, it also brings serious information security challenges. The remote access to IEDs exposes substation networks to various threats, such as message injection and denial of service (DoS) attacks. Historically, cyber attacks on the power grid affected hundreds of thousands of people in countries like the USA, Iran, and Ukraine [5]. For example, in 2016, a warning was issued about coordinated cyber attacks on 35 Ukrainian substations, leaving over 225,000 people without power [4]. In 2022, Russian hackers targeted the Ukrainian power grid to cause a blackout that would have hit 2 million people [21]. Moreover, IEDs typically have constrained computing resources that limit the implementation of traditional security mechanisms, such as complex encryption algorithms [7, 31, 37].

Therefore, employing intrusion detection systems (IDSs) is crucial for the security of IEDs in digital substations. IDSs can be characterized as host-based (HIDS) or network-based (NIDS). In particular, NIDS focuses on information collected by multiple hosts: messages transmitted over substation networks are collected and analyzed. HIDS, on the other hand, focuses on data coming from a single host (or IED, in the

✉ Silvio E. Quincozes
silvioquincozes@unipampa.edu.br

Vagner E. Quincozes
vequincozes@id.uff.br

Diego Passos
dpassos@ic.uff.br

Célio Albuquerque
celio@ic.uff.br

Daniel Mossé
mosse@pitt.edu

1 Computer Science Department, Universidade Federal Fluminense, Niterói, Brazil

2 Universidade Federal do Pampa, Campus Alegrete, Alegrete, Brazil

3 Programa de Pós-Graduação em Computação (PPGCO), Universidade de Uberlândia, Uberlândia, Brazil

4 Instituto Superior de Engenharia de Lisboa (ISEL), DEETC, Lisbon, Portugal

5 Computer Science, University of Pittsburgh, Pittsburgh, PA 15260, USA

context of substations) [25]. Among all the information that can be extracted from both IEDs and substation networks, only the relevant features should be fed to the IDSs.

Unfortunately, the current literature lacks solutions to enable effective intrusion detection in substations. Many existing IEC–61850-based IDSs rely on anomaly detection and specification rules that have limited attack detection capabilities—they fail to detect some attack types, such as the masquerade attack [4, 6, 14, 16, 37, 38]. On the other hand, a few studies based on attack signatures focus on data from sources other than IEC–61850 protocols to model the attackers' behavior [15, 24].

A major issue in this field is the extraction of representative characteristics (often called *features* [25]). In our previous work [29], presented at CSNet22, we demonstrated the initial stages of feature extraction by correlating data from both the electrical domain, such as electric voltage and current from IED messages. This approach showed promising results for machine learning-based IDSs.

In the present work, we make significant advancements in intrusion detection tailored to the digital substation domain. Our main contributions are as follows:

- **Feature augmentation:** We augment the feature set from 14 to 53 by introducing 39 engineered features, as an extension of our prior work [29].
- **Extended dataset:** A significantly richer dataset is generated and made publicly available at Kaggle [30]. It comprises traffic from more devices and contains nearly 40 times more attacks than previous datasets.
- **Enhanced detection results:** The new dataset enhanced the capacity of machine learning-driven IDSs to detect intrusions in digital substations.
- **Insightful feature impact analysis:** We also provide novel insights into the importance of individual features in intrusion detection, thereby deepening the understanding of the attack vectors and their impacts.

Furthermore, we categorized the proposed features into four distinct categories which comprise features from the cyber (i.e., network messages and application logs) and physical (i.e., electrical measurements) domains. Our proposed features are extracted through novel feature engineering methods, namely interprotocol and temporal approaches. The remainder of the paper is structured as follows: Section 2 lays down the theoretical foundations on IEC–61850 protocols, security threats, and feature selection for intrusion detection. Section 3 surveys the existing literature. Section 4 describes our multi-layered feature extraction strategy. Section 5 presents the materials and methods used in our evaluation, while Section 6 presents the results of our simulations. Section 7 summarizes our conclusions.

2 Background

To define the power system requirements and communication protocols between IEDs, the International Electrotechnical Commission (IEC) created an international standard named IEC–61850. Its latest edition, IEC–61850-2:2019 [11], was recently published. Three application layer communication protocols were introduced to substation networks. The IEC–61850-8-1 [8] specifies the use of Manufacturing Messaging Specification (MMS) and Generic Object Oriented Substation Event (GOOSE). The former was adopted as a messaging system for exchanging real-time data and supervisory control information between IEDs and control centers (e.g., Supervisory Control and Data Acquisition (SCADA) systems). The latter introduces a communication model which uses fast and reliable mechanisms to transmit messages between IEDs (e.g., protection IEDs) [28]. The IEC–61850-9-2 [9] specifies the sampled value (SV) protocol for transmitting digitized values of current and voltage measurements over the network.

The IEC–61850 standard was conceived with two assumptions about critical messages such as those transmitted by GOOSE and SV protocols: (i) they are time-critical, so any processing overhead is avoided [13, 28], and (ii) they are not particularly vulnerable to cyber attacks because they are usually used in physically protected/isolated networks [7]. As a result, initially, no recommendations were defined to secure these messages. However, these assumptions do not correspond to current scenarios. Existing applications, such as distribution automation and transmission protection schemes, transmit GOOSE messages outside the substations' premises. Also, malware installed from updates for software from manufacturers or infected devices (e.g., computers or flash drives) may expose local networks to external attackers [7].

The IEC–62351 [10] standard defined a series of security measures for substation protocols, including GOOSE. However, due to the low latency requirements imposed by the IEC–61850-5 standard (e.g., 3 ms for messages of type 1A, class P1 [13, 28]), cryptographic algorithms are usually avoided so as not to impose overhead on the latency of these messages. According to the IEC–62351-6, it is not recommended to use encryption in messages that require little computational overhead and response times at this level [7]. Regarding the use of authentication mechanisms, no limitations are established. However, some manufacturers do not implement any security mechanism to avoid increasing the action time in failure situations [7].

2.1 Substation threats

Substation networks can be divided into three levels: process level, bay level, and station level. They are, respectively,

associated with (i) switching equipment used in the transmission of electricity, (ii) an intermediate level where automatic functions with real-time requirements are performed without the need for human intervention (e.g., control and metering, protection, and time synchronization), and (iii) an interface for humans managing the substation, which includes monitoring systems, engineering workstations, SCADA systems, and the remote terminal unit [25]. We focus on the process level, where the following attacks may be launched (they are summarized in Table 1).

Replay attacks are performed when legitimate messages are captured and transmitted again over the network, without modifying their content. Such transmission may occur at any time (not necessarily immediately after the message is captured).

GOOSE poisoning attacks are executed by strategically tampering with the *status number* ($stNum$) field of GOOSE messages. The purpose of such attacks is to induce IEDs to discard subsequent legitimate messages because their $stNum$ is lower than the expected value, thus resulting in a DoS attack. There are three variations of poisoning attacks [17]: (i) *High-Status Number* attacks transmit a single spoofed GOOSE message with an excessively high *status* number; (ii) *High-Rate Flooding* attacks transmit multiple spoofed messages, each increasing the $stNum$ value by one; and (iii) the *Semantic Attack* observes the network traffic to determine the current $stNum$ and uses a higher value in the fake transmitted message [17, 31]. While the *Semantic Attack* is similar to the *High-Status Number*, it differs by increasing the $stNum$ by a small amount, based on the observed current value. The goal is to hinder the detection of the attack.

Message injection attacks build and transmit false and potentially malicious messages into the network. In the simplest form of message injection, a message is randomly created without observing its consistency with the rules of the IEC–61850 standards (i.e., it may contain invalid field values).

Table 1 Attacks at the process level (based on [25])

Attack class	Description
Replay [5, 31]	Old messages are re-transmitted
Naive injection [7]	Fabricated messages are sent
IEC–61850 injection [7]	IEC–61850 compliant messages are sent
Masquerade [35]	Messages mimicking legitimate ones are sent
Poisoning [18]	The *stNum* is excessively increased
Modification [31]	Specific features are adulterated
Flooding [18, 31]	Many messages are transmitted at high frequency

Packet modification attacks are characterized by tampering with all or part of the content of captured messages to cause an IED to perform improper or unauthorized actions. Once modified, the packet is transmitted to the substation network, emulating a legitimate packet [31]. This attack is possible because critical messages are transmitted on the local substation network without any encryption mechanism [10].

Masquerade is a particular packet modification attack in which the attacker hampers the detection by mimicking the legitimate behavior and fields (i.e., using valid values for status, $sqNum$, $stNum$, MAC address) of captured legitimate messages [31]. A practical version of masquerade is implemented through three simple steps [7]: observation, tampering, and transmission.

Lastly, *flooding attacks* are based on sending multiple fake messages, each message increasing the $stNum$, according to values expected at the subscriber devices, thus appearing legitimate and increasing the difficulty of detection. Additionally, legitimate messages experience delays due to contention in the network or devices during the flooding [25].

2.2 Feature selection for intrusion detection

To detect attacks against substation networks, employing IDSs is fundamental. In this context, feature selection plays an indispensable role in identifying a subset of relevant attributes from a larger set of data, focusing only on those that are most indicative of malicious activity. This procedure is broadly classified into three methods: filter, wrapper, and embedded methods. Filter methods rank features based on their intrinsic relevance, wrapper methods evaluate features by using specific machine learning algorithms, and embedded methods perform feature selection as part of the model training process [26].

Within the scope of feature selection techniques, the Incremental Wrapper Subset Selection (IWSS) utilizes a wrapper-based approach but conducts it incrementally, assessing one feature at a time. This incremental nature makes it computationally more efficient. By selecting the most relevant features for the detection models, IWSS not only enhances the intrusion detection performance but also reduces the computing time needed for training machine learning algorithms and detecting intrusions, a critical aspect when real-time intrusion detection is required at the process level of a substation network. This optimization (i.e., feature selection) is crucial for minimizing both false positives and false negatives, as irrelevant and/or redundant information is removed, thereby bolstering the overall security infrastructure of the substation network [33].

3 Related work

The literature characterizes IDSs into three main groups: specification-based, anomaly-based, or signature-based [25]. Specification-based methods assume predefined static rules that catch suspicious activities when violated. Anomaly-based methods focus on profiling legitimate behavior to distinguish it from anomalous activities. Finally, signature-based methods use a database with labeled samples that represent known attack profiles [2]. We summarize the related works with respect to these three categories in Table 2.

As shown in Table 2, most IDSs designed for substations networks are specification-based [4–6, 14, 16, 37, 38]. This approach has been widely used to secure digital substations by IDSs, blocking traffic that violates predefined rules. However, these naive specification-based IDSs present several issues that limit their effectiveness in detecting intrusions. Firstly, they rely on predefined rules, which can be easily bypassed by attackers who know the specification. Secondly, these systems cannot detect unknown or zero-day attacks, as they are not designed to identify new and emerging threats. Advanced attacks, such as masquerade, are typically not detected by these specification rules. Thirdly, specification-based IDSs may generate false alarms or miss real intrusions, as the rules are often overly generic and may not capture subtle variations in network traffic patterns [25].

Another group of works focuses on deploying anomaly-based IDSs to secure the digital substation. They rely on identifying deviations from normal network behavior to detect intrusions. However, this approach can generate false positives, as variations on legitimate behavior may be misclassified as anomalies. These issues highlight the need for a more sophisticated and adaptive IDS that can effectively detect intrusions in digital substations [25].

The studied signature-based IDSs focus on traditional protocols (i.e., ICMP, FPT, HTTP, and ARP) and a station-level protocol (i.e., MMS). They are built by off-the-shelf tools such as Hydra-THC,[1] and Suricata.[2] Using ML techniques is a strong candidate for deploying efficient signature-based IDSs [27]. Nevertheless, its adoption is still at an early stage in substation networks.

In our previous work [29], we have demonstrated that ML-based IDSs have great potential in detecting attacks in digital substations. In particular, in that work, we have carried out initial efforts in performing feature engineering (i.e., studying how to derive novel features from the existing information available in the substation domain). However, to effectively detect intrusions, these systems would need to be equipped with enhanced features that can effectively capture the complex and dynamic nature of network traffic in substations. Firstly, the IDSs need to be able to extract and correlate relevant multi-layer information, such as network traffic, device configurations, and control commands. Secondly, they need to be able to handle the variability and heterogeneity of network traffic in substations, as different devices and protocols may generate different types of network traffic. Finally, they need to be able to adapt to the changing nature of threats and security requirements in substations, as new and emerging threats may require the development of new features. These requirements highlight the need for enhanced features for ML-based IDSs to effectively detect intrusions in digital substations. The main limitation of our previous work that is addressed in this extended version is the limited feature engineering since it included only 14 features. Moreover, the previous experiments were based on a limited set of attack samples in which only 15 attacks were considered. In this extended version, the feature engineering process generated 53 features and the dataset was expanded to comprise 598 attack samples, greatly improving the statistical significance of the results.

[1] Available at https://github.com/vanhauser-thc/thc-hydra

[2] Available at https://suricata.io/

4 Proposed feature extraction and engineering

This section presents our proposed methodology for feature extraction and engineering tailored for ML-based IDSs in digital substations. This process is based on the use of knowledge from the application domain. We discuss the types of features, their relevance, and the methods employed to extract and enrich them. We categorize these features into four distinct groups: GOOSE features, SV features, application features, and enriched features.

We define the types of features as follows:

- **GOOSE features**: Existing message fields and our proposed aggregated features.
- **SV features**: Features pertaining to voltage and current measurements.
- **Application features**: Derived from IEDs application logs.
- **Enriched features**: Novel features created by combining and processing the above types.

For each group of features, we selected those that are correlated to the IEDs' protection application. These features include current and voltage from the SV protocol due to their correlation with electrical faults, network features selected features from the GOOSE protocol, which are extracted according to their correlation to fault events, and application features from the IED logs.

Table 2 Current IDSs proposals for digital substations

Ref	IDS approach	Data source	Number of features	Number of records
Hong et al. [5]	Specification-based	GOOSE, SV	N/A	N/A
Hong et al. [6]	Specification-based	GOOSE, SV	N/A	N/A
Hong and Liu [4]	Specification-based	GOOSE, SV	N/A	N/A
Yang et al. [37]	Specification-based	MMS, GOOSE, SV	N/A	N/A
Yang et al. [38]	Specification-based	MMS, GOOSE, SV	N/A	N/A
Kim and Park [16]	Specification-based	GOOSE, SV	N/A	N/A
Kabir-Querrec et al. [14]	Specification-based	MMS, GOOSE, SV	N/A	N/A
Ten et al. [34]	Anomaly-based	Event log	N/A	N/A
Kwon et al. [19]	Anomaly-based	MMS, GOOSE	N/A	N/A
Yoo and Shon [39]	Anomaly-based	MMS, GOOSE	N/A	N/A
Yang et al. [36]	Anomaly-based	*Station Level*	N/A	N/A
Premaratne et al. [24]	Signature-based	ICMP, FTP, HTTP, ARP	N/A	N/A
Kang et al. [15]	Signature-based	MMS	N/A	N/A
Quincozes et al. [29]	Signature-based	GOOSE, SV	14	15
This work	Signature-based	GOOSE, SV	53	598

Note that datasets are not available for most published works

4.1 GOOSE features

We selected four fields from GOOSE messages to be considered features for ML processing based on their relation to common attacker activities. Additionally, we proposed four new features which were derived from them. The new (preprocessed) features are underlined in Table 3, whereas features taken directly from the GOOSE messages are not. Those preprocessed features bring additional information about the changes along consecutive messages. These features are from our previous work [29].

Table 3 Features from GOOSE messages. Our new proposed features are underlined

Feature	Description
1 - time	The GOOSE timestamp
2 - sqNum	The GOOSE sequence number
3 - stNum	The GOOSE status number
4 - cbStatus	A flag that indicates the Circuit-Breaker (CB) status (i.e., open/closed)
5 - sqDiff	A integer that indicates how much the *sqNum* value changed from the last message
6 - stDiff	A integer that indicates how much the *sqNum* value changed from the last message
7 - timeLastMsg	The time elapsed since the last message
8 - recentChange	A flag that indicates recent changes to *cbStatus* (within four messages)

4.2 SV features

In our previous work [29], we considered only three voltage measurement features, which corresponded to the angles of the voltage of the three phases as measured by a single merging unity (MU). In this work, we expand that set of features to 24 by taking voltage measurements from four MUs, as well as including current measurements. Those 24 features are summarized in Table 4. The use of these features was originally performed by [22], where the angle of voltage and current measures were computed at multiple phases.

Anomalous SV measurements may be related to electrical faults. When an electrical fault occurs, the GOOSE messages are expected to have their *cbStatus* field set to *open* value. Therefore, the extracted SV features are correlated to the GOOSE features, as network messages should reflect the circuit breaker status when electrical faults happen. An interesting insight about using both pieces of information together is that they are useful for detecting inconsistencies between messages that may reveal the attacker's actions.

4.3 Application features

Another novel contribution of the present study is the introduction of application features, derived from the logs of IEDs. These features have not been explored in our research and are formally presented in this section. Application features have the potential to indicate the detection of specific events at the originating IED. Despite not always being perfectly correlated with fault events, they may be helpful for building

a more precise model for intrusion detection. The four application features considered in this work are summarized in Table 5.

Table 4 SV features based on its electrical measurements

Feature	Description
9 - $MU1_{v^a_{phase}}$	Angle of electrical voltage measured in Phase A from MU 1
10 - $MU1_{v^b_{phase}}$	Angle of electrical voltage measured in Phase B from MU 1
11 - $MU1_{v^c_{phase}}$	Angle of electrical voltage measured in Phase C from MU 1
12 – $MU2_{v^a_{phase}}$	Angle of electrical voltage measured in Phase A from MU 2
13 – $MU2_{v^b_{phase}}$	Angle of electrical voltage measured in Phase B from MU 2
14 – $MU2_{v^c_{phase}}$	Angle of electrical voltage measured in Phase C from MU 2
15 – $MU3_{v^a_{phase}}$	Angle of electrical voltage measured in Phase A from MU 3
16 – $MU3_{v^b_{phase}}$	Angle of electrical voltage measured in Phase B from MU 3
17 – $MU3_{v^c_{phase}}$	Angle of electrical voltage measured in Phase C from MU 3
18 – $MU4_{v^a_{phase}}$	Angle of electrical voltage measured in Phase A from MU 4
19 – $MU4_{v^b_{phase}}$	Angle of electrical voltage measured in Phase B from MU 4
20 – $MU4_{v^c_{phase}}$	Angle of electrical voltage measured in Phase C from MU 4
21 – $MU1_{i^a_{phase}}$	Angle of electrical current measured in Phase A from MU 1
22 – $MU1_{i^b_{phase}}$	Angle of electrical current measured in Phase B from MU 1
23 – $MU1_{i^c_{phase}}$	Angle of electrical current measured in Phase C from MU 1
24 – $MU2_{i^a_{phase}}$	Angle of electrical current measured in Phase A from MU 2
25 – $MU2_{i^b_{phase}}$	Angle of electrical current measured in Phase B from MU 2
26 – $MU2_{i^c_{phase}}$	Angle of electrical current measured in Phase C from MU 2
27 – $MU3_{i^a_{phase}}$	Angle of electrical current measured in Phase A from MU 3
28 – $MU3_{i^b_{phase}}$	Angle of electrical current measured in Phase B from MU 3
29 – $MU3_{i^c_{phase}}$	Angle of electrical current measured in Phase C from MU 3
30 – $MU4_{i^a_{phase}}$	Angle of electrical current measured in Phase A from MU 4
31 – $MU4_{i^b_{phase}}$	Angle of electrical current measured in Phase B from MU 4
32 – $MU4_{i^c_{phase}}$	Angle of electrical current measured in Phase C from MU 4

Table 5 Application log features

Feature	Description
33 - $MU1_{Log}$	Boolean status from MU 1 application log
34 - $MU2_{Log}$	Boolean status from MU 2 application log
35 - $MU3_{Log}$	Boolean status from MU 3 application log
36 - $MU4_{Log}$	Boolean status from MU 4 application log

4.4 Enriched features

For intrusion detection in substations, having only single isolated measurement values (e.g., voltage angle) is not always sufficient to indicate malicious activities targeting the protection system (i.e., when an attacker mimics electrical faults during normal operation or *vice versa*). Combining multiple such measurements (features) allows us to generate more representative features to distinguish malicious activities from legitimate ones. In particular, the simple use of GOOSE and SV features may not provide enough information to accurately detect advanced attacks. To optimize the performance, it is crucial to carefully assess the relevant information and perform feature engineering to extract the most representative features from multiple data sources.

In this regard, we propose and compare two feature engineering methods: interprotocol and temporal. The first one is based on the correlation between the GOOSE and SV protocols (features 37–42). The second one is based on the correlation between multiple GOOSE messages transmitted at different times (features 5–8). To provide more valuable information, we also aggregate information from multiple fields in the same SV message (features 43–52) or multiple device statuses (feature 53). Below, we describe the principles and methods adopted. The generated features as a result of our feature engineering are presented in Table 6.

According to Meliopoulis [20], three phases are balanced if and only if (i) their voltages vary sinusoidally with time, (ii) the amplitudes of the three phases are equal, and (iii) there is a 120-degree difference between adjacent phase voltages. Thus, to compose new features with greater representation, we assume a three-phase system with three balanced sine waves. Once the voltage and current angle measurements are captured, an approximate 120-degree displacement between the measures from adjacent phases is expected.

The voltage displacement between phases a (v^a_{phase}) and b (v^b_{phase}) is referred to as D^{vab}, and between (v^b_{phase}) and phase c (v^c_{phase}) is referred to as D^{vbc}. They are denoted in Eqs. 1 and 2, respectively.

$$D^{vab} \leftarrow v^b_{phase} - v^a_{phase} \simeq 120 \tag{1}$$

$$D^{vbc} \leftarrow v^c_{phase} - v^b_{phase} \simeq 120 \tag{2}$$

Table 6 New features enriched from the correlation of GOOSE messages with electrical measurements (current and voltage) and application features

Feature	Description
37 - $MU1_{cs}$	Computed status for MU1 from its current and voltage
38 - $MU2_{cs}$	Computed status for MU 2 from its current and voltage
39 - $MU3_{cs}$	Computed status for MU 3 from its current and voltage
40 - $MU4_{cs}$	Computed status for MU 4 from its current and voltage
41 - cs	Computed status from the current and voltage from all MUs
42 - *consistency*	Equality flag between *computedStatus* and *cbStatus*
43 - $MU1_{threePhaseV}$	Voltage lag between the three electrical phases of MU 1
44 - $MU2_{threePhaseV}$	Voltage lag between the three electrical phases of MU 2
45 - $MU3_{threePhaseV}$	Voltage lag between the three electrical phases of MU 3
46 - $MU4_{threePhaseV}$	Voltage lag between the three electrical phases of MU 4
47 - $MU1_{threePhaseI}$	Current lag between the three electrical phases of MU 1
48 - $MU2_{threePhaseI}$	Current lag between the three electrical phases of MU 2
49 - $MU3_{threePhaseI}$	Current lag between the three electrical phases of MU 3
50 - $MU4_{threePhaseI}$	Current lag between the three electrical phases of MU 4
51 - $threePhaseV_{Sum}$	Sum of voltage lag between the three phases of all MUs
52 - $threePhaseI_{Sum}$	Sum of current lag between the three phases of all MUs
53 - $anyIED_{Log}$	A flag that indicates if any IED is presenting an alert

By following the same logic, an approximated 120-degree displacement is expected between the current of the phase a (i^a_{phase}) and phase b (i^b_{phase}), which is denoted as D^{iab} in Eq. 3. Similarly, the same displacement is expected between i^b_{phase} and the current of phase c (i^c_{phase}), which is denoted as D^{ibc} in Eq. 4.

$$D^{iab} \leftarrow i^b_{phase} - i^a_{phase} \simeq 120 \tag{3}$$

$$D^{ibc} \leftarrow i^c_{phase} - i^b_{phase} \simeq 120 \tag{4}$$

If D between two adjacent phases is overreaching a threshold (i.e., at most 10 degrees from expected), we set our enriched feature $computedStatus$ (cs) to $true$ (or 1), indicating a balanced state. Otherwise, the value is set to $false$ (or 0). This computing is denoted in Equation 4.4, where x represents the current (i) or voltage (v) values from two consecutive phases (y and z).

$$\text{For all } y, z \in \{1, 2, 3\}, \quad \text{cs} = \begin{cases} 1, & \text{if } |D^{xyz} - 120| > 10^\circ \text{ for x = v or x = i} \\ 0, & \text{otherwise} \end{cases}$$

Different from our previous work [29], in this extended version, we consider a network with four MUs rather than only one. Therefore, in this work, we generated five *computeStatus* features: one feature for each MU and another by combing the four individual MUs' features (i.e., if any of the former is true, the latter is set as true; otherwise, it is set as false). Also, in this work, the *computedStatus* considers both current and voltage rather than only voltages.

Additionally, two other groups of features (see Table 6) were derived from the correlation between GOOSE messages with the D^{vab} and D^{iab} electrical measurements. The *threePhaseV* feature represents the sum of the displacements D^{vab} and D^{vbc}, as illustrated in Eq. 5. The *threePhaseI* feature represents the sum of the displacements D^{vab} and D^{vbc}, as illustrated in Eq. 6. Note that, under normal conditions, these features are expected to have an approximate value of 240, since the expected displacement between adjacent phase voltages is 120 degrees.

$$threePhaseV \leftarrow D^{vab} + D^{vbc} \simeq 240 \tag{5}$$

$$threePhaseI \leftarrow D^{iab} + D^{ibc} \simeq 240 \tag{6}$$

The *consistency* feature is a flag that is set to true if the computed value for the *computedStatus* (cs) matches the *cbStatus* value in the GOOSE message payload. Finally, the $anyIED_{Log}$ feature is a flag that has its value set as true if any of the IEDs has its MU_{Log} feature marked as true.

5 Materials and methods

To assess the proposed features, we implement a dataset generator software to reproduce the behavior of GOOSE and SV messages, encompassing both legitimate and malicious activities into a simulated substation network. We needed to generate the dataset given the lack of availability of real-life datasets that could be used in this research.

5.1 Dataset generator software

We implemented a dataset generator software using the Java programming language. Such a generator comprises scripts to model the legitimate behavior of GOOSE messages following the pattern proposed by the IEC–61850 standards

(e.g., retransmission time, sequence number increment, status number changes).

The new GOOSE messages were generated based on our modeled scenario (Section 5.2). The generated GOOSE messages are assumed to come from one legitimate protection IED and/or one compromised protection IED (attacker). The reproduced SV messages are assumed to come from four MUs.

Furthermore, to generate SV data, our dataset generator software takes as input the energy data (current and voltage) available in [22] and uses their basic features as a reference for generating our proposed features presented in this section.

5.2 Implemented scenario

Our implemented scenario comprises four MUs that generate SV messages and one protection IED that generates GOOSE messages. Note that we do not generate the network message; instead, we simulate it in software and store only the resulting dataset records. This setup is illustrated in Fig. 1.

The protection IED receives SV messages from merging unities and generates GOOSE messages. The SV payload contains current and voltage samples. To reproduce such samples, we use a public power system dataset [23] that contains both samples during normal operation and during electrical faults. From the received SV messages, our traffic generation tool simulated 506 benign GOOSE messages (455 with no events and 41 with electrical faults). These messages carry features that mimic the action of the protection IED in operating an electrical circuit breaker to open (i.e., in case of faults) or close (i.e., in case of line re-establishment). For example, a boolean value in the GOOSE frame is transmitted to represent the state of the circuit breaker, indicating whether it is open or closed.

In addition to legitimate traffic, 598 attack attempts were simulated: 100 are message replay attacks, 101 represent data injection attacks, and 397 represent *masquerade* attacks. Except for replays, attacks change the circuit breaker status (and eventually the values of other fields) to the desired value—typically the opposite of the real state of the system.

5.3 Intrusion detection and performance metrics

We implemented an IDS based on the K-nearest neighbors (KNN) algorithm. We used it as a multi-class classifier, where classes 0 and 1 refer to benign traffic (normal and faults, respectively) and classes 2, 3, and 4 refer to malicious traffic (injection, masquerade, and replay, respectively). To assess the detection performance of KNN, we compute the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). To measure these values, any malicious sample classified into one attack class was computed as a TP. Similarly, any benign sample classified into fault or normal operation class was computed as TN. From these indicators, we compute accuracy, recall, precision, and F1-score [3].

Accuracy is given by the fraction of correct IDS classifications with respect to the total number of samples analyzed (Eq. 7). Recall computes the percentage of detected attacks from the universe of existing attack samples (Eq. 8). Precision denotes the percentage of attack classifications that are, in fact, attacks instead of false positives (Eq. 9). Finally, we also compute F1-score (Eq. 10)—the harmonic mean between precision and recall—as it is a more reliable and widely used metric [1].

$$Accuracy \leftarrow \frac{TP + TN}{TP + FP + TN + FN} \tag{7}$$

$$Recall \leftarrow \frac{TP}{TP + FN} \tag{8}$$

$$Precision \leftarrow \frac{TP}{TP + FP} \tag{9}$$

$$F1 - score \leftarrow 2 \times \frac{Precison \times Recall}{Precision + Recall} \tag{10}$$

6 Experiments

In this section, we present our experiments and discussions. The experiments reported in this section are aimed at answering three key questions related to the deployment of ML-based IDSs in digital substations. Firstly, we evaluate different data sources to determine which one performs better for intrusion detection (Section 6.1). Secondly, we perform feature engineering to generate more valuable features and evaluate their performance to determine the best features for intrusion detection (Section 6.2). Lastly, we perform an assessment to identify the most relevant features for intrusion detection (Section 6.3).

6.1 Data source assessment

We start by showing which stand-alone data source can provide more representative information for intrusion detection in IEC–61850 networks. Four groups of features from different data sources are compared: GOOSE, voltage, current, and application. We also consider a fifth group comprising the combined features of the other four (*Basic Set*), as denoted in Eq. 11.

$$BasicSet = \{GOOSE, voltage, current, application\} \tag{11}$$

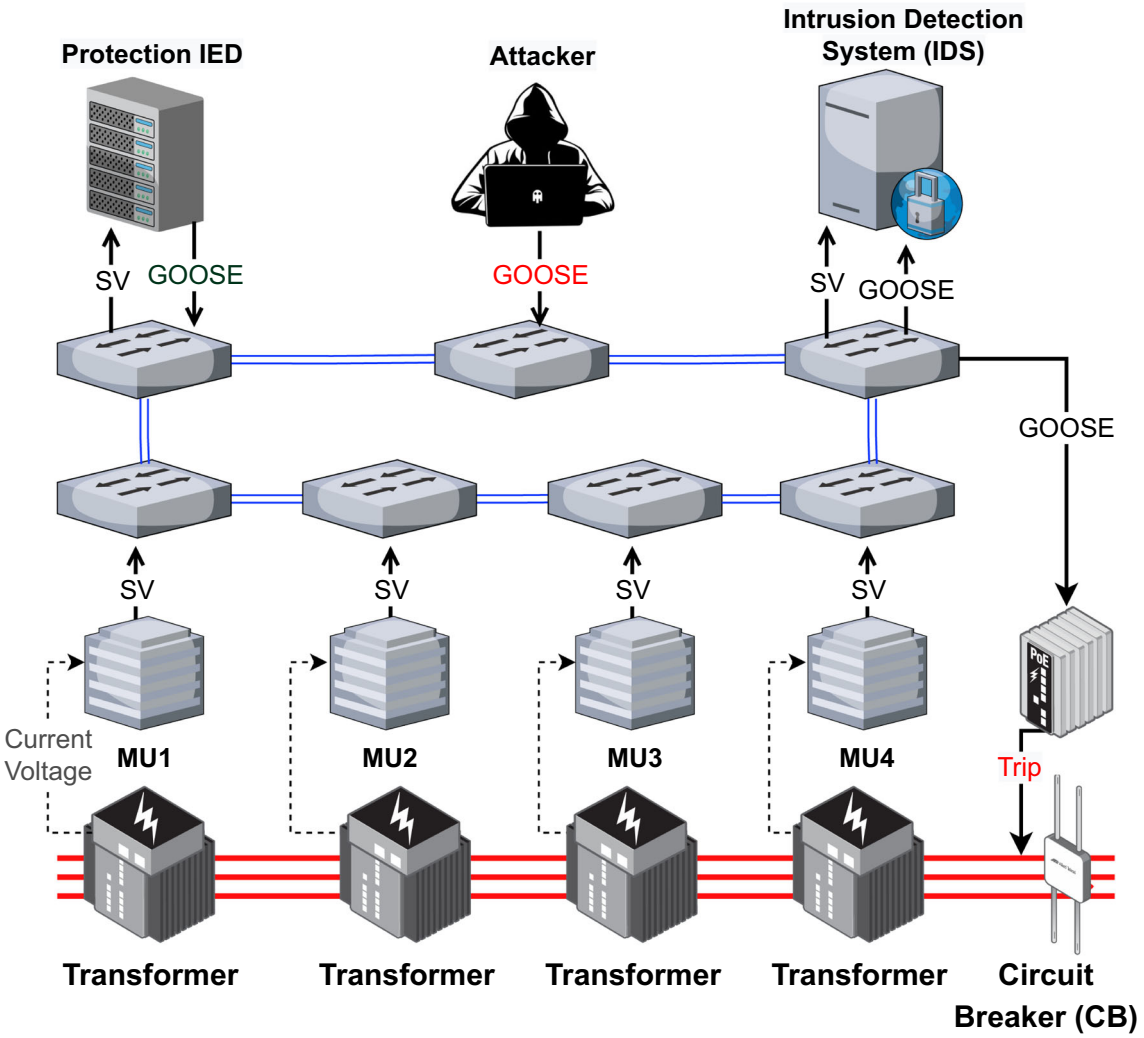

Fig. 1 Simulated substation scenario, based on [25]

Table 7 illustrates the list of features included in each group. Notice that in this experiment, we do not include any enriched feature. To assess all non-enriched (*Basic Set*) features for each data source, we choose the KNN algorithm because it does not perform embedded feature selection. Figure 2 depicts the found results.

We observed that the GOOSE data source performed the best, reaching an F1-score of 84.5%. On the other hand, the voltage data source had an F1-score of 55.7%, while the current data source had an F1-score of 53%. The application data source performed the worst, with an F1-score of only 8.5%. When all information from these data sources is combined (*Basic Set*), the F1-score is 82.6%. In fact, combining multiple data sources without proper feature selection can lead to lower performance results and may harm the F1-score of the ML-based IDS in digital substations. On the other hand, by applying the Incremental Wrapper-based Subset Selection (IWSS) algorithm to the group of features comprising all sources, we were able to improve the results to an F1-score of 89.7%.

Therefore, the main insight of this experiment is that using a single data source may not achieve the best performance, but when combining data sources, it is essential to also perform feature selection to remove irrelevant information to provide better results.

6.2 Feature engineering

In Fig. 3, we compare the performance of interprotocol and temporal correlation methods. We observed that the interprotocol correlation method performs better than the temporal one, reaching an F1-score of 82.8%, while the temporal correlation obtained an F1-score of only 50.7%. Interestingly, using IWSS could not provide further improvements for any of these methods, as well as for their combination. On the other hand, combining these two methods without

Table 7 List of features included in each data source

Data source	Included feature group	
	Without IWSS	With IWSS
GOOSE	1–4	1–4
Voltage	9–20, 43–46, 51	9, 12, 43, 45, 46, 51
Current	21–32, 47–50, 52	47, 52
Application	33–36, 53	35, 36
Basic Set	1–4, 15–53	1–4, 18, 33, 36, 45, 51, 53
Interprotocol	37–42	37, 41, 42
Temporal	5–8	5–8
Inter-Temporal	5–8, 37–42	5–8, 42
All-Features	1–53	1, 3 - 7, 53
IG-Ranked	1–8, 18, 33–36, 40–42, 46–53	1–8, 10, 21–23, 40, 43

performing feature selection can significantly improve the detection performance: such a combination resulted in an F1-score of 96.4%, demonstrating the effectiveness of these feature engineering methods in enhancing the performance of ML-based IDSs in digital substations.

Nevertheless, one question remains: Why does IWSS fail to improve the results? One potential answer is greedy nature, which may remove features without realizing that, combined with other features, they are important. However, at this point, it is still not clear which features may present this behavior in the experimented digital substation networks scenario.

6.3 Most relevant features

To gain a deeper understanding of the role of the most important features, firstly, it is necessary to identify them. While finding the best feature subset is an NP-hard problem, an alternative is to apply existing feature selection methods to reach an optimized set of features. In this section, we investigate the performance of the KNN algorithm for different feasible feature subsets.

Our first goal is to determine whether using the filter-based information gain (IG) to rank (and select) the features can improve the results with respect to the full set of features. Then, we aim to discover if the wrapping-based method IWSS can outperform IG. Finally, we explore their combination by applying IWSS to the features selected by IG. The results for these experiments are compiled in Fig. 4.

The experiments using the IG algorithm selected all 53 features that presented an information gain higher than zero ($IG > 0$). Although both the IG and IWSS algorithms were able to improve the F1-score for the KNN algorithm, from 95.6 to 97.7% and 99.4%, respectively, the combination of both algorithms is actually slightly worse than using only IWSS, yielding an F1-score of 99%. This suggests that using only the IWSS algorithm is more effective in optimizing detection performance in the studied scenario. Therefore, from this point on, we assume that the most relevant known features are those selected by the IWSS. In Fig. 5, we detail the effect of IWSS—in comparison to the full set of features—in the other evaluated metrics: precision, recall, and accuracy. IWSS presented superior results for all metrics, with an outstanding value of 100% in precision.

6.4 Feature relevance discussion

Finally, to explain the importance of the selected features, we analyze them in light of the domain knowledge. To measure how much each feature contributed to the F1-score, we performed the following experiment. We start with a given set of features and measure its F1-score. Then, for each feature in that set, we recompute the F1-score after removing only that particular feature. We repeated that experiment twice: first for the set of features selected by IWSS (Fig. 6)—from all available features—and then for the complete set of features itself (Fig. 7).

The main finding is that the removal of features with indexes 4 and 7 significantly affects (negatively) the performance of the IDS, either considering the complete set of available features (Fig. 7) or the subset given by IWSS (Fig. 6). Another interesting finding is that from the features selected by IWSS, either feature index 1 or feature index 6 can be removed without impacting the systems' performance—however, we identified that removing both results in a worse accuracy. Finally, removing features 3 or 5 can slightly increase IWSS performance.

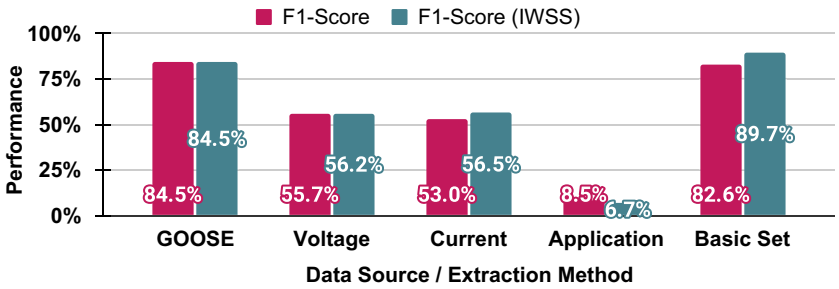

Fig. 2 Data source comparison: GOOSE messages have more representative features than other sources, but combining them can give more information

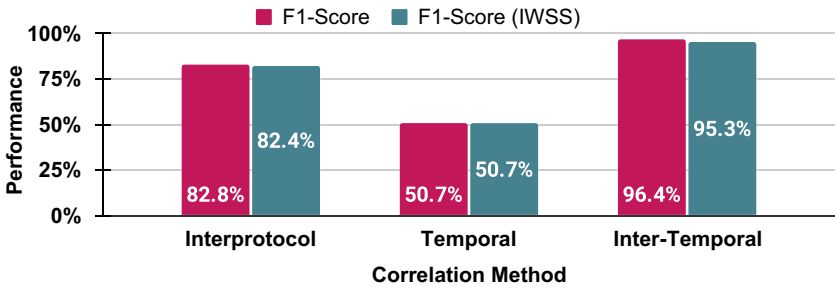

Fig. 3 Correlation method assessment: Interprotocol correlation is better than temporal correlation, but combining them performs even better

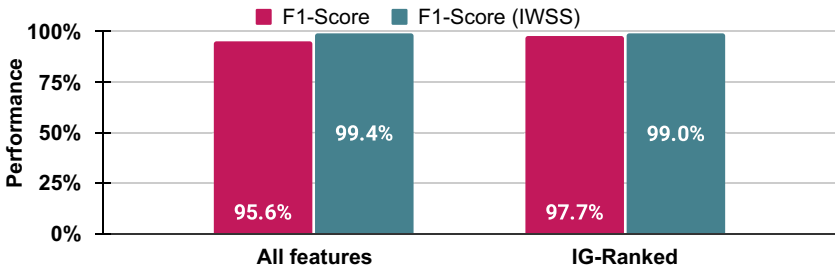

Fig. 4 Feature selection: Ranking-based feature selection (with IG algorithm) can improve the F1-score, but using only IWSS is more effective

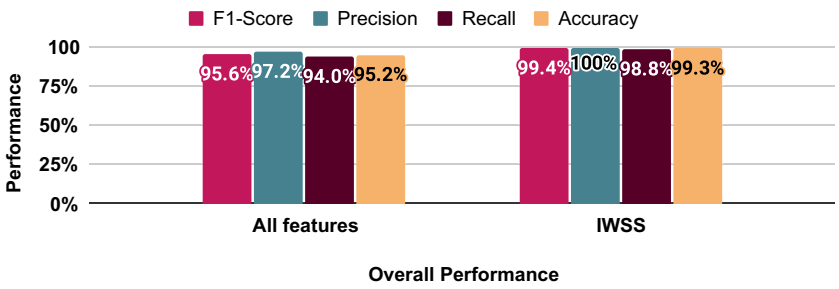

Fig. 5 Results for all metrics

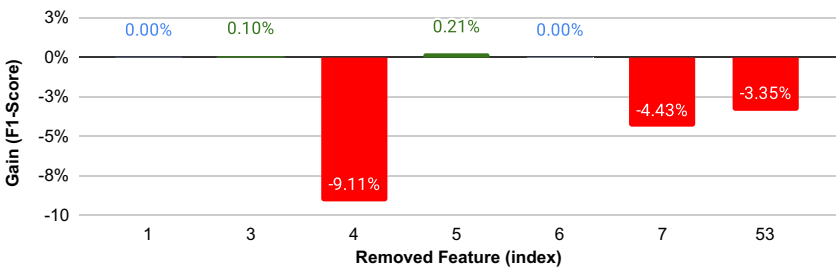

Fig. 6 Impact on F1-score of removing each feature selected by IWSS

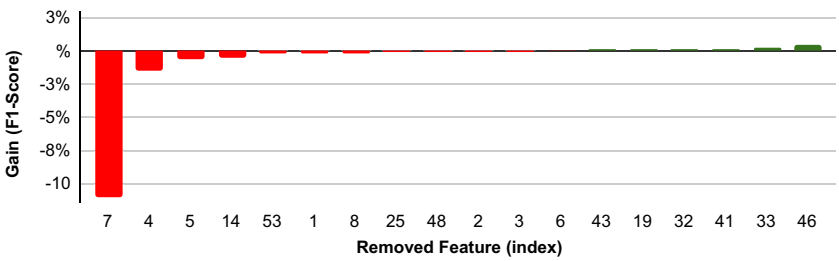

Fig. 7 Impact on F1-score of removing each feature from the *All-Features* complete dataset (features whose removal does not present any positive/negative impact were suppressed)

In the upcoming subsections, we will move the focus of our discussion to the features obtained through the IWSS algorithm, which improved the performance of the KNN algorithm. These features extract pertinent information from the data, enabling KNN to make more accurate predictions with a smaller input. Analyzing these features is important to provide insights into the underlying patterns and essential characteristics of the data, contributing to a better understanding of intrusion detection in digital substations.

6.4.1 IWSS feature 1—GOOSE timestamp (time)

The *GOOSE timestamp* feature (index 1) is an important element of the GOOSE message transmission because it indicates the exact time at which a given message was sent. While this feature was originally selected by IWSS to be part of a reduced subset of features, removing it did not impact the F1-score. However, when considering the complete set of features, removing it results in a 0.19% loss in F1-score. This suggests that while the feature may not be essential in a smaller subset, it can still have a positive effect overall.

Overall, the *GOOSE timestamp* relevance may be explained by the ability that it can bring to an IDS in checking (i) timestamp validity, (ii) time synchronization, and (iii) the GOOSE message age. Firstly, if a message has an incorrect or invalid timestamp, it may indicate that the message has been altered in transit or generated by an unauthorized device. Secondly, a GOOSE message's timestamp should be synchronized with the network's time source to ensure accurate and consistent timestamps. Thus, if the timestamps in a GOOSE message are not synchronized, it may indicate a problem with the network's time source or a malfunctioning device. Finally, the age of a GOOSE message can be determined by comparing its timestamp with the current time. Suppose a message is received with a timestamp that is significantly older than the current time. In that case, it may indicate that the message has taken an unusual or unexpected path through the network, which could indicate a security breach. In our scenario, injection attacks may cause the aforementioned situations.

6.4.2 IWSS feature 3—status number (stNum)

Another feature that was examined in the IWSS selection was the *stNum* feature (index 3). It was found that when this feature was removed from the IWSS selection, a small gain of 0.1% in F1-score was observed. However, when considering the full set of features, removing *stNum* resulted in a 0.09% decrease in F1-score. This suggests that the importance of the *stNum* feature varies depending on the specific subset of features that are being analyzed. While it may not be critical in some contexts, it can still play a role in improving the F1-score of the overall analysis in certain cases.

The GOOSE *stNum* field can help detect cyber attacks because any unexpected changes in the value of this field within a GOOSE message can suggest an attempt to manipulate or disrupt the normal synchronization and monitoring of the devices in the substation automation system. By monitoring and analyzing this field, an IDS can gain more information to potentially detect cyber attacks. However, when fewer features are analyzed (and less information are available to characterize the benign or malicious behavior), the *stNum* may confuse the IDS because its value can change frequently and rapidly, making it difficult for an IDS to distinguish between normal and abnormal changes. This can lead to a higher false positive rate, as well as the potential for missed or delayed detection of actual cyber attacks. Therefore, when analyzing a reduced subset of features, it may be more challenging to accurately interpret the significance of changes in the *stNum* field, compared to when analyzing the complete set of features.

6.4.3 IWSS feature 4—circuit breaker status (cbStatus)

The *cbStatus* (index 4) is an essential feature of GOOSE messages, especially for fault protection applications as it indicates the current position of the CB (open or closed). When this feature was excluded, it resulted in a 1.5% drop in the F1-score of the full set and a 9.1% drop in the IWSS subset. These results highlight the critical role that this feature plays in detecting security breaches and protecting the network from malicious activity.

The main reason this feature is useful is that the CB is critical for ensuring the proper operation and safety of the power system. Any unexpected change in the state of the CB within a GOOSE message can be an indication of a security breach or malicious activity, such as a masquerade attack or a replay attack, in which an attacker attempts to manipulate the state of the CB to cause a disruption or failure in the system.

6.4.4 IWSS feature 5—sequence number difference (sqDiff)

Each GOOSE message contains a *sqNum* which is used to indicate the sequence number of the message being sent. By examining the difference between the sequence numbers of two consecutive GOOSE messages *stDiff* feature (index 5), an IDS may detect the presence of an intruder who is disrupting the typical behavior observed in the substation network.

However, while removing *stDiff* from the complete set of features causes a loss of F1-score of 0.60%, its removal from the subset selected by IWSS provides an increase of 0.21%. This result suggests the presence of other features that are well correlated with *stDiff* in the complete feature set. Therefore, when this feature is removed, the information captured by these correlated features may also be lost, leading

to a decrease in performance. On the other hand, in the subset selected by IWSS, these correlated features may have already been removed, making the removal of the *stDiff* feature more beneficial.

6.4.5 IWSS feature 6—status number difference (stDiff)

The *stDiff* feature (index 6) computes the difference between the status numbers of two consecutive GOOSE messages. Consequently, it is expected that by examining the `stDiff` feature, an IDS may detect the presence of an intruder who is disrupting the normal communication pattern between the devices.

However, when this feature was removed from the complete set, an increase of 0.1% in the F1-score was observed. Also, it had no effect on the IWSS subset. These results indicate that this feature may not be as critical in detecting security breaches as some of the other features in these two scenarios.

To seek an explanation of why IWSS selected this feature, it is important to consider its impact on the overall F1-score of the intrusion detection system could be dataset-dependent. In other words, while the *sqDiff* feature did not improve the F1-score when considered stand-alone in the resulting dataset after IWSS filtering, it was still selected by IWSS because, in the particular moment in which it was added, it presented an improvement to the classification performance.

Overall, the selection of features by the IWSS is a data-driven process, and the relevance of each feature depends on its interaction with the other features and their collective ability to improve the F1-score of the intrusion detection system in a specific scenario. Therefore, while the *stDiff* feature may be useful in some cases, its inclusion in the feature set should be carefully evaluated to ensure that it is contributing to the F1-score of the IDS. In our particular context, we found it is not necessary to be included as a part of the final feature subset selected.

6.4.6 IWSS feature 7—time from last message (timeLastMsg)

The *timeLastMsg* feature (index 7) is a critical indicator of network health and stability because it provides valuable information on the periodicity of GOOSE messages, enabling the identification of messages that may have been delayed or injected by malicious actors.

By analyzing this feature, the intrusion detection system can quickly detect any abnormalities and take the necessary action to mitigate the threat. Therefore, the *timeLastMsg* feature should be carefully considered in any analysis of GOOSE messages, as it is an important component for maintaining the integrity and security of the network.

Indeed, the importance of the *timeLastMsg* feature cannot be overstated. Its removal resulted in a substantial drop in the F1-score, with a 10.99% decrease in the full set of features and a 4.4% decrease in the IWSS feature subset. This indicates that the *timeLastMsg* plays a critical role in the F1-score of the IDS, and its absence can significantly impact the system's ability to detect cyber threats. Therefore, it is imperative to include this feature in any analysis of GOOSE messages to ensure that the IDS can accurately identify and respond to anomalies and potential security breaches.

6.4.7 IWSS feature 53—alerts in any IED (anyIED)

The *anyIED* feature (index 53) is a flag extracted from the application logs that indicates whether any IED in a substation has raised an alert. The flag is used to alert operators and other monitoring systems that an abnormal event may be occurring in the substation, such as a fault or electrical disturbance, which could potentially impact the reliability or safety of the power grid.

The *anyIED* flag can be useful for intrusion detection by providing an indication of potentially malicious activity or abnormal behavior within the substation network. For example, if an attacker is attempting to modify or disrupt the normal operation of the substation, this may be reflected in changes to the IEDs statuses that are detected by the combination of the *anyIED* flag with other information. For example, the *time* and *timeLastMsg* features can provide information on the frequency and timing of GOOSE messages, which can be compared to the occurrence of alerts from IEDs. Additionally, the *stNum* and *cbStatus* features can help identify any abnormal changes in the status of GOOSE messages or circuit breakers, which may be related to the occurrence of alerts. Finally, the *sqDiff* feature can be used to analyze the sequence of GOOSE messages, which can provide information on the occurrence of events and potential causes of alerts (e.g., status changes would reset the sequence number).

By analyzing the *anyIED* flag in conjunction with these other features, it may be possible to identify patterns or anomalies that are indicative of cyber attacks or other security breaches. The absence of this feature caused a 0.21% drop in the full set F1-score and a 3.3% drop in the IWSS selection. This suggests that this feature plays an important role in the IWSS, although its impact on the F1-score of the full subset is not as significant.

7 Conclusion

Digital substations are critical infrastructure in the power industry. With the increasing reliance on digital substations, it is imperative to prevent cyber attacks that could lead to power outages, safety hazards, and economic losses. In our

previous work [29], we demonstrated the initial steps in feature extraction by correlating data from the electrical domain and GOOSE messages. In this work, we implemented novel features, increasing the total number from 14 to 53 features. Also, we experimented with more attacks, almost 40 times more attacks than our previous work. Finally, based on the produced larger dataset, we extended our analysis to better explain how attackers affect the behavior of the features studied.

In particular, we proposed novel features for ML-based IDSs in digital substations based on the international standard IEC–61850, which specifies the communication protocols and communication services for electrical power substation automation. The proposed method extracts relevant multi-layer information from the digital substation, taking into account the specific characteristics of IEC–61850. The extracted features are used to train an ML model for intrusion detection. This approach promotes more satisfactory detection performance for the ML-based IDSs deployed in digital substations.

As a result of this work, we provide novel insights into the impact of the best features in the intrusion detection process, enhancing our understanding of the system's behavior. Our experiments showed that the proposed method achieved an F1-score of up to 95.6%, demonstrating its effectiveness in detecting intrusions in digital substations. When a proper feature selection method is used, such results can be further improved to up to 99.4%. Therefore, this work provides a necessary solution for enhancing the security of digital substations and opens up new opportunities for further research in this field.

Finally, the proposed method can serve as a benchmark for the development of more advanced IDSs and contribute to the advancement of the power industry towards a safer, more secure, and more efficient security solution.

Funding This work is supported in part by CAPES, CNPq, FAPERJ, and CGI/FAPESP.

Data Availability The data is available at Kaggle (https://www.kaggle.com/datasets/sequincozes/ereno-iec61850-ids)

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Bej S, Davtyan N, Wolfien M, Nassar M, Wolkenhauer O (2021) LoRAS: an oversampling approach for imbalanced datasets. Mach Learn 110(2):279–301
2. Bostani H, Sheikhan M (2017) Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach. Comput Commun 98:52–71
3. Fawcett T (2006) An introduction to ROC analysis. Pattern Recognit Lett 27(8):861–874. https://doi.org/10.1016/j.patrec.2005.10.010. http://www.sciencedirect.com/science/ARTICLE/pii/S016786550500303X
4. Hong J, Liu C (2019) Intelligent electronic devices with collaborative intrusion detection systems. IEEE Trans Smart Grid 10(1):271–281
5. Hong J, Liu C, Govindarasu M (2014) Detection of cyber intrusions using network-based multicast messages for substation automation. In: Innovative smart grid technologies (ISGT), pp 1–5. IEEE
6. Hong J, Liu CC, Govindarasu M (2014) Integrated anomaly detection for cyber security of the substations. IEEE Trans Smart Grid 5(4):1643–1653
7. Hoyos J, Dehus M, Brown TX (2012) Exploiting the GOOSE protocol: a practical attack on cyber-infrastructure. In: 2012 IEEE Globecom workshops, pp 1508–1513. IEEE
8. IEC (2022) Communication networks and systems for power utility automation - Part 8-1: Specific communication service mapping (SCSM) - Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3. IET
9. International Electrotechnical Commission (2004) IEC 61850-9-2 communication networks and systems in substations - Part 9-2: Specific Communication Service Mapping (SCSM) - Sampled values over ISO/IEC 8802-3, 1 edn. IET
10. International Electrotechnical Commission (2010) IEC 62351 security, 1 edn. IET
11. International-Electrotechnical-Commission (2019) Communication networks and systems for power utility automation - Part 2: Glossary (Second Edition). IET
12. International-Electrotechnical-Commission (2022) Communication networks and systems for power utility automation - ALL PARTS. IET
13. International-Electrotechnical-Commission (2022) Communication networks and systems in substations–Part 5: Communication requirements for functions and device models. IET
14. Kabir-Querrec, M., Mocanu, S., Thiriet, J.M., Savary, E (2015) Power utility automation cybersecurity: IEC 61850 specification of an intrusion detection function. In: 25th European safety and reliability conference (ESREL 2015). CRC Press
15. Kang, B., McLaughlin, K., Sezer, S (2016) Towards a stateful analysis framework for smart grid network intrusion detection. In: Proceedings of the 4th international symposium for ICS & SCADA cyber security research, pp 124–131
16. Kim J, Park J (2018) FPGA-based network intrusion detection for IEC 61850-based industrial network. ICT Express 4(1):1–5
17. Kush N, Ahmed E, Branagan M, Foo E (2014) Poisoned GOOSE: exploiting the GOOSE protocol. In: Proceedings of the twelfth australasian information security conference, vol 149, pp 17–22. Australian Computer Society, Inc
18. Kush N, Branagan M, Foo E, Ahmed E (2014) Poisoned GOOSE: exploiting the GOOSE protocol. In: Proceedings of the twelfth Australasian information security conference (AISC 2014), pp 17–22. Australian Computer Society, Inc
19. Kwon Y, Kim HK, Lim YH, Lim JI (2015) A behavior-based intrusion detection technique for smart grid infrastructure. In: 2015 IEEE Eindhoven PowerTech, pp 1–6. IEEE
20. Meliopoulis AS (2017) Power system grounding and transients: an introduction. Routledge
21. O'Neillarchive PH (2022) Russian hackers tried to bring down Ukraine's power grid to help the invasion. https://www.technologyreview.com/2022/04/12/1049586/russian-hackers-tried-to-bring-down-ukraines-power-grid-to-help-the-invasion/. Acessed 05 Sep 2022
22. Pan S, Morris T, Adhikari U (2015) Developing a hybrid intrusion detection system using data mining for power systems. IEEE Trans Smart Grid 6(6):3104–3113

23. Pan S, Morris T, Adhikari U (2015) Developing a hybrid intrusion detection system using data mining for power systems. IEEE Trans Smart Grid 6(6):3104–3113
24. Premaratne UK, Samarabandu J, Sidhu TS, Beresh R, Tan JC (2010) An intrusion detection system for IEC61850 automated substations. IEEE Trans Power Delivery 25(4):2376–2383
25. Quincozes SE, Albuquerque C, Passos D, Mossé D (2021) A survey on intrusion detection and prevention systems in digital substations. Comput Netw 184(107):679
26. Quincozes SE, Passos D, Albuquerque C, Mossé D, Ochi LS (2022) An extended assessment of metaheuristics-based feature selection for intrusion detection in cps perception layer. Ann Telecommun 77(7–8):457–471
27. Quincozes SE, Raniery C, Ceretta Raul Albuquerque C, Passos D, Mosse D (2019) A counselors-based intrusion detection architecture. In: 9th Latin American network operations and management symposium (LANOMS 2019), pp 1–8. IFIP
28. Quincozes SE, Soares AAZ, Oliveira W, Cordeiro EB, Lima RA, Muchaluat-Saade DC, Ferreira VC, Lopes Y, Vieira JL, Uchôa LM et al (2019) Survey and comparison of SDN controllers for teleprotection and control power systems. In: LANOMS
29. Quincozes VE, Quincozes SE, Albuquerque C, Passos D, Mossé D (2022) Feature extraction for intrusion detection in IEC-61850 communication networks. In: 2022 6th Cyber security in networking conference (CSNet), pp 1–7. IEEE
30. Quincozes VE, Quincozes SE, Passos D, Albuquerque C, Mossé D (2023) Power system intrusion dataset. Available at https://www.kaggle.com/datasets/sequincozes/power-system-intrusion-dataset/data, Accessed 20 Oct 2023
31. Rashid MTA, Yussof S, Yusoff Y, Ismail R (2014) A review of security attacks on IEC61850 substation automation system network. In: Proceedings of the 6th international conference on information technology and multimedia, pp 5–10. IEEE
32. Saadi K, Abbou R (2022) On IEC 61850 communication networks in smart grids system: methodology of implementation and performances analysis on an experimental platform. Int J Energy Res 46(1):89–103
33. Silva EF, Naves N, Quincozes SE, Quincozes VE, Kazienko JF, Cheikhrouhou O (2023) GDLS-FS: scaling feature selection for intrusion detection with GRASP-FS and distributed local search. In: International conference on advanced information networking and applications, pp 199–210. Springer
34. Ten CW, Hong J, Liu CC (2011) Anomaly detection for cybersecurity of the substations. IEEE Trans on Smart Grid 2(4):865–873
35. Ustun TS, Farooq SM, Hussain SS (2019) A novel approach for mitigation of replay and masquerade attacks in smartgrids using IEC 61850 standard. IEEE Access 7:156,044–156,053
36. Yang Q, Hao W, Ge L, Ruan W, Chi F (2019) FARIMA model-based communication traffic anomaly detection in intelligent electric power substations. IET Cyber-Phys Syst Theory Appl 4(1):22–29
37. Yang Y, McLaughlin K, Gao L, Sezer S, Yuan Y, Gong Y (2016) Intrusion detection system for IEC 61850 based smart substations. In: 2016 IEEE power and energy society general meeting (PESGM), pp 1–5. IEEE
38. Yang Y, Xu HQ, Gao L, Yuan YB, McLaughlin K, Sezer S (2016) Multidimensional intrusion detection system for IEC 61850-based SCADA networks. IEEE Trans Power Delivery 32(2):1068–1078
39. Yoo H, Shon T (2015) Novel approach for detecting network anomalies for substation automation based on IEC 61850. Multimed Tools Appl 74(1):303–318