data

MDPI

Article

CARE to Compare: A Real-World Benchmark Dataset for Early Fault Detection in Wind Turbine Data

Christian Gück *, Cyriana M. A. Roelofs and Stefan Faulstich

Fraunhofer IEE, Joseph-Beuys-Straße 8, 34117 Kassel, Germany; cyriana.roelofs@iee.fraunhofer.de (C.M.A.R.); stefan.faulstich@iee.fraunhofer.de (S.F.)
* Correspondence: christian.gueck@iee.fraunhofer.de

Abstract: Early fault detection plays a crucial role in the field of predictive maintenance for wind turbines, yet the comparison of different algorithms poses a difficult task because domain-specific public datasets are scarce. Many comparisons of different approaches either use benchmarks composed of data from many different domains, inaccessible data, or one of the few publicly available datasets that lack detailed information about the faults. Moreover, many publications highlight a couple of case studies where fault detection was successful. With this paper, we publish a high quality dataset that contains data from 36 wind turbines across 3 different wind farms as well as the most detailed fault information of any public wind turbine dataset as far as we know. The new dataset contains 89 years worth of real-world operating data of wind turbines, distributed across 44 labeled time frames for anomalies that led up to faults, as well as 51 time series representing normal behavior. Additionally, the quality of training data is ensured by turbine-status-based labels for each data point. Furthermore, we propose a new scoring method, called CARE (Coverage, Accuracy, Reliability and Earliness), which takes advantage of the information depth that is present in the dataset to identify good early fault detection models for wind turbines. This score considers the anomaly detection performance, the ability to recognize normal behavior properly, and the capability to raise as few false alarms as possible while simultaneously detecting anomalies early.

Dataset: https://doi.org/10.5281/zenodo.14006163 (accessed on 29 October 2024).

Dataset License: CC BY-SA 4.0 International.

Keywords: dataset; early fault detection; wind turbines; predictive maintenance; anomaly detection; condition monitoring

check for updates

Citation: Gück, C.; Roelofs, C.M.A.; Faulstich, S. CARE to Compare: A Real-World Benchmark Dataset for Early Fault Detection in Wind Turbine Data. *Data* **2024**, *9*, 138. https://doi.org/10.3390/data9120138

Academic Editor: Giuseppe Ciaburro

Received: 12 August 2024
Revised: 11 November 2024
Accepted: 20 November 2024
Published: 23 November 2024

1. Introduction

Wind energy plays a crucial role in the transition to renewable energy, but monitoring and maintaining wind farms and turbines is a costly challenge. These wind farms are often located in regions with challenging weather conditions, leading to complex operating conditions and increased risk of unexpected failures and downtime. Over the past decade, various approaches for condition monitoring, many of which focus on early fault detection using SCADA data, have been investigated [1–3].

A common method to detect component failures early is AD, which identifies outliers or other anomalous patterns in the data. In the context of WT, most AD techniques utilize data from the SCADA system, failure logs, vibration data, and occasionally status and maintenance logs [4]. This paper specifically focuses on AD models used for early fault detection based on SCADA data that are validated using additional failure information.

While there have been several benchmarks [5,6], reviews [7], and comparisons [8] of general AD-algorithms, most of them use data from a wide variety of domains like spacecraft, medical applications, and IT-related data. However, efforts on wind-energy-specific

AD or early fault detection are usually based on non-public inaccessible data. For example, refs. [9,10] use inaccessible data from wind farms that are located in China, and [11–13] use data from anonymized offshore wind farms. These are five recently published examples that lack the ability for meaningful comparisons between the proposed fault detection algorithms. Also, inaccessible data prevent reproducibility of presented results. Some studies have used public wind energy datasets (for example, [4,14–20]), but they lack comprehensive information about anomalies or component faults. The lack of extensive public datasets with both SCADA time series and failure information is a significant limitation in the field of WT SCADA data analysis. To enable meaningful comparisons between early fault detection models in the wind energy domain, new public benchmark datasets are necessary.

The main contribution of our work is the publication of the most extensive WT SCADA dataset[1] for early fault detection yet. This includes high dimensional data from multiple wind farms, information about the WT status at all times, labeled anomalies with annotated starts and ends, and additional fault descriptions. Because the data stem from real-world operating wind farms, they had to be anonymized with the focus on minimizing the loss of useful information and maximizing the meaningfulness of this dataset for early fault detection and predictive maintenance.

In addition to the dataset, we also provide a sophisticated score, the CARE score, for evaluating early fault detection algorithms on this and similar datasets. This score takes into account four key aspects of a high-quality early fault detection model for predictive maintenance. In combination with the dataset, this score provides the possibility to compare a variety of different early fault detection algorithms, from unsupervised to semi-supervised techniques, designed for the use case of predictive maintenance in WT.

The content of this paper is divided into the following sections. First, we give an overview of the related work in Section 2. After that, we introduce the dataset in Section 3.1 by giving information about the layout, the requirements we set for the quality of the data, the labeling process, and the anonymization actions that were taken. Following this, we provide our scoring idea together with a test evaluation of a few selected basic AD strategies and an early fault detection model in Sections 3.2 and 4. Finally, a summary concludes this work in Section 5.

2. Related Work

In the field of AD benchmark data, many studies focus on dataset compositions from various different domains and use cases. Many benchmark datasets also include a mix of artificial data and data from real-world applications. While [5,7,21,22] study a wide spectrum of different AD algorithms for a broad collection of data types, there are also several AD benchmarks that focus on time series data. One example of such benchmarks is the widely used and cited Numenta benchmark [6], which provides a collection of datasets. A more recent and more comprehensive evaluation of AD in time series is found in [8], where over 71 algorithms were evaluated on more than 900 time series.

In the scope of early failure detection for WT, the time series datasets mentioned above are usually too broad to be used for evaluation in this specific context. Also, many are either univariate or synthetic time series and therefore not applicable to early failure detection in SCADA data. Unfortunately, most domain-specific evaluations are conducted on inaccessible data that were provided only for the research in which they are used. As mentioned in the introduction, there are plenty of examples of studies that use such datasets.

There are only a handful of open datasets containing WT SCADA data. The studies [23,24] give an overview about existing datasets for WT. Additionally, the Git-repository [25] summarizes some currently existing datasets, although some of the listed datasets, such as the SCADA data of the ENGIE wind farm "La Haute Borne", are not available anymore. The most relevant public dataset in the context of early fault detection is provided by the EDP open data platform [26], since it is, as far as the authors know, the only one containing information about WT faults in addition to the SCADA data. The faults are provided in the form of a start timestamp for some turbine faults. This dataset was used in the fault detection challenge "hack the wind" [27] hosted by EDP, which is mentioned and evaluated together with the "WeDoWind"-challenge [28] in [19,20]. These challenges focus in particular on the evaluation of AD algorithms based on maintenance cost and potential savings that could be achieved through predictive maintenance. Furthermore, several studies on fault detection have used these data [4,15,17,18]. Although the EDP dataset is widely used, its level of detail regarding the fault information is small, especially in comparison to the inaccessible datasets mentioned before.

The lack of publicly available SCADA datasets of WT is also acknowledged in [3], which highlights that it is a constraint in the progress of WT SCADA applications. Additionally, the absence of publicly available datasets containing real-world anomalies is recognized as a significant obstacle in the development of AD in general, as it may not adequately reflect the performance of methods in real-world applications [7,22].

There is not only a need for additional domain specific public datasets; the data quality and the level of detail also plays an important role. As pointed out by [29], many AD benchmark datasets suffer from flaws that limit their significance. The main flaws are defined as the flaws of "Triviality", "Unrealistic Anomaly Density", "Mislabeled Ground Truth", and the "Run-to-Failure Bias". In the case of publicly available wind SCADA datasets, one common issue is the absence of labels, particularly regarding fault information.

Considering the flaws in datasets, scoring for AD algorithms poses a difficult challenge. Many studies utilize standard classification metrics, such as accuracy, precision, recall, or the AUC of the ROC [5,11,13,30].

While it is possible to evaluate AD algorithms using the AUC-ROC score for all possible thresholds, for most practical applications, it is much more useful to have a high F-score, or a related score. In [31], several variants of F-scores are compared. The standard pointwise F-score is the simplest, but for most use cases, the interest lies in detecting anomaly events, i.e., a continuous set of anomalous time points, rather than individual time points. A composite F-score is introduced, a modification of the classic F-score that takes into account anomaly events through event-wise recall.

Another approach, presented in [32], modifies the classic AUC-ROC metric by generalizing the concept of the ROC to the preceding-window-ROC, thereby adjusting the measure to better fit AD evaluations on time series data from an event-based perspective.

Finally, the Numenta Benchmark [6] defines a score that is supposed to measure the performance of more general AD models for time series data across different domains. The score is based on five key aspects of a good AD model: "detection of all anomalies", "early detection of anomalies", "no false alarms", "uses only real time data", and "automation across all different datasets".

Based on the provided overview of related work, this paper contributes to the progress of early failure detection for predictive maintenance on WT by introducing a new public dataset that offers more detailed information about turbine faults and associated anomalies. Furthermore, the new dataset addresses the flaws identified by [29], although the potential for mislabeled ground truth cannot be completely eliminated in this context, as the start of anomalous behavior is often unclear. The flaw of triviality is tackled by the inclusion of complex anomalies from real-world WT based on feedback of the wind farm operators. Additionally, the proposed CARE score, which differs from standard classification metrics, draws inspiration from the first three key aspects of the Numenta score and the composite F-score from [31], while distinct adaptions and further developments have been made to better fit the specific use case of early failure detection for predictive maintenance on WT.

3. Materials and Methods

3.1. Data

In this section, we describe the new dataset provided with this paper. First, we discuss the requirements for a good dataset for early failure detection in WT in Section 3.1.1. Then, we provide an overview of the data published in Section 3.1.2, including general statistics such as the number of anomalous events and features, as well as data quality. In Section 3.1.3, we explain the process of labeling each time series and datapoint, and in Section 3.1.4, the anonymization process is described.

3.1.1. Data Requirements

During the process of selecting data for this benchmark dataset, seven requirements were defined to ensure the quality and significance of comparisons of algorithms for early failure detection in WT. The requirements are as follows:

1. The dataset must contain as many anomaly events as possible.
2. The dataset must contain different wind farms.
3. The dataset must contain different fault types.
4. The dataset must be balanced, i.e., contain enough prediction data representing normal behavior.
5. Every sub-dataset must contain enough normal behavior data in the intended training time frame. If at least 2/3 of the training data are normal behavior data, we define the sub-dataset to be sufficient.
6. Every sub-dataset must contain at least one whole year worth of data, in order to be able to learn seasonality-related effects.
7. Every anomaly must have an assigned start timestamp. The anomaly end is the start of a turbine fault.

While requirements 1 to 3 are necessary to test the generalization ability of early failure detection algorithms, requirement 4 enables tests for the ability to learn normal behavior effectively. This is particularly important for the evaluation of NBM. Additionally, requirements 5 and 6 ensure the quality of the training data, to guarantee an NBM can be trained. Finally, requirement 7 allows for the evaluation of early failure detection models using classification measures. These requirements ensure that the dataset is of high quality, comprehensive, and balanced to train a proper NBM, with detailed labels to validate the model. All these properties are also relevant for the definition of the score introduced in Section 3.2.1.

3.1.2. Dataset

The data consist of 95 datasets, containing 89 years of SCADA time series distributed across 36 different WT from the three wind farms A, B, and C. The data for wind farm A is based on the earlier mentioned EDP data [26] and consists of five WT of an onshore wind farm in Portugal. From these data, 22 datasets were selected to be included in this data collection. The other two wind farms are offshore wind farms located in Germany. All three datasets were anonymized as described in Section 3.1.4. The overall dataset is balanced, as 44 out the 95 datasets contain a labeled anomaly event and the other 51 datasets represent normal behavior. Each dataset is provided in form of a csv file, with columns defining the features and rows representing the data points of the time series.

The datasets consist of SCADA time series data for each turbine, with a resolution of 10 min. Each dataset includes one year worth of data for training a model, as well as 4 to 98 days of prediction data.

The prediction data are divided into an event time frame, with varying amounts of padding data before and after the event. This padding is used to prevent guessing the event label (“anomaly” or “normal”) based on the amount of prediction data.

The number of features in the datasets varies depending on the wind farm. Wind farm A has 86 features, wind farm B has 257 features, and wind farm C has 957 features. In addition to the sensor data features, each time series includes five descriptive features: a row ID and a timestamp, an asset ID that identifies the WT, a “train_test” column indicating whether the row belongs to the training or prediction data, and a status ID indicating the turbine status at the timestamp.

The remaining features represent sensor measurements. For each sensor, the 10-min average value is available. Some sensors also have additional information in the form of 10-min minimum, maximum, and standard deviation values. The original sensor names have been replaced in order to anonymize the data, as described in Section 3.1.4. Only features that describe power, reactive power, or wind speed are recognizable by their name. To accommodate for the loss in information, additional descriptions are provided for every sensor. These descriptions include a brief text, the unit of the sensor, as well as Boolean indicators that imply whether the sensor represents a regular sensor signal, a counter, or an angle. The most important statistics of the data are summarized in Table 1. The rows "Anomaly events" and "Normal behavior" describe the number of datasets containing an anomaly event and without anomalies, respectively.

Regarding the data quality, there are two challenges. The data for wind farm B and C were provided by the operator, with 0 values replacing all missing values, so large amounts of consecutive 0 values must be treated with caution. Second, note that the status values for wind farms B and C may be inconsistent; often the status is only logged when it changes, which may fail if there is a brief communication error. Also, the status values for wind farm A were derived based on the EDP fault logbook, which only contained start timestamps of the faults (see Section 3.1.3). It is therefore advisable to check the power and wind speed values in addition to the status values to determine whether the turbine has indeed been operating normally.

Table 1. Summary of the dataset.

	Wind Farm A	**Wind Farm B**	**Wind Farm C**	**Overall**
Turbines	5	9	22	36
Datasets	22	15	58	95
Anomaly events	11	6	27	44
Normal behavior	11	9	31	51
Features	86	257	957	-
Sensors	54	63	238	-

3.1.3. Data Labeling

The data are labeled on two levels. The first level are the so-called event labels. If a dataset contains an anomaly event inside the prediction time frame, the dataset is labeled as an anomaly. If this is not the case, it is labeled as normal. The anomaly labels have been determined either based on direct feedback by the wind farm operators or based on documented faults in the form of service reports and fault logbooks. The normal labels have been determined by a combination of feedback of the wind farm operators, manual inspection of the data, and expert knowledge.

For wind farm A, all anomaly event starts were defined based on the available EDP fault logbook, which only defines start timestamps for each fault. Since no further information is available, analysis of the data before every fault was used to determine possible event starts. The "true" anomaly event starts for wind farm A can differ from the set ones.

For wind farms B and C, all starts of the anomaly events were defined based on data analysis, feedback of the wind farm operator, service report documents, and expert knowledge. While the true starts of the anomaly events could potentially differ from the set ones in some cases, it is highly unlikely that the defined events start too early. If anything, anomaly event start could be earlier than defined.

The second level of labeling assigns a label to each timestamp of every dataset. These labels are called status IDs. For wind farms B and C, they are derived from the original operating modes that were provided by the wind farm operators in combination with service report information. For wind farm A, this information was not provided. In this case, the status IDs were based on the fault information from the logbook provided by EDP. For each turbine fault, the preceding 14 days were marked with the status ID 4 (fault), and the 3 days after the fault timestamp were marked with the status ID 3 (service mode).

The time ranges around the turbine fault were set with the aim in mind to reduce the risk of including anomalous behavior in the training data. As no information is available on the duration of anomalies before and after the given faults, the time ranges were chosen conservatively.

The status labels can be used to infer whether a given data point represents normal WT behavior. The status IDs, their description, and whether we consider the status normal are found in Table 2.

Table 2. Status descriptions.

Status ID	Description	Considered Normal
0	Normal operation without limitations	True
1	Derated power generation with a power restriction	False
2	Asset is idling and waits to operate again	True
3	Asset is in service mode/service team is at the site	False
4	Asset is down due a fault or other reasons	False
5	Other operational states for example system test, setup, ice build-up, or emergeny power	False

3.1.4. Anonymization

Due to confidentiality reasons, the data of wind farms B and C were anonymized. The anonymization includes the removal of all information that can directly identify the wind farms, such as the name of the wind farm, the original names of each WT, the turbine type, and the location. The wind farm names were replaced with the generic names “Wind Farm A”, “Wind Farm B”, and “Wind Farm C” while the WT names were replaced by randomized asset IDs. However, the asset IDs were assigned in a way that makes it still possible to link different datasets that belong to the same WT.

Additionally, the timestamps of each dataset were shifted by a number of years so that datasets start within the year 2022. This anonymization of the recording year maintains the consistency of seasonal data, but it also distorts the temporal order of the datasets.

Names of the original SCADA features were replaced by a numeration of the features. Only features that describe power, reactive power, or wind speed are recognizable by their name. Additionally, power and reactive power features have been scaled with the rated power of the turbine. This way, it is still possible to clean and analyze the data using the power curve of the WT.

All status information was aggregated from the original status data of the WT, and the name of each status condition was replaced by a number in combination with a brief description. Wind farms B and C contain detailed status information while wind farm A only contains status information that indicate turbine faults.

3.2. Anomaly Detection Evaluation

Evaluation of early failure detection algorithms poses a difficult task. On one hand, the perfect early failure detection should detect all anomalies as soon as possible, without any false alarms; on the other hand, labeling of anomalies and finding proper start and end times of anomaly events cannot be done perfectly.

Although the ground truth of every AD evaluation is almost certainly flawed [29], standard classification metrics, like the F-score, accuracy, and precision, are often used to measure the performance of AD algorithms to compare them with other algorithms or to show their overall performance [33]. The F-score, in particular, is widely used, but it cannot be applied to evaluate the performance of AD on normal data since true negatives are not considered in the F-score. This is one of the reasons why metrics like the F-score are not suitable for a complete evaluation of early fault detection models.

To tackle these problems, we introduce the CARE score in Section 3.2.1 to evaluate models on the dataset described in Section 3.1. The score is composed from four sub-scores, each evaluating a key aspect of a good early failure detection model. In addition to that, we conduct a test evaluation Section 4 to showcase the CARE score and dataset.

3.2.1. The CARE Score

In the context of early fault detection for predictive maintenance the performance of models is often difficult to assess. To address this, we introduce the CARE score for evaluating early fault detection algorithms in an operational predictive maintenance setting. Expanding upon the first three criteria of the Numenta score [6], the CARE score focuses on four key aspects that a good early fault detection model should excel in, which are:

1. Coverage: Detection of as many correct anomalies as possible;
2. Accuracy: Recognition of normal behavior;
3. Reliability: Few false alarm events;
4. Earliness: Detection of anomalies before faults become critical.

The CARE score consists of four sub-scores, each representing one of the four aspects mentioned above. The first and fourth sub-scores measure the pointwise classification performance of an algorithm on datasets where anomaly events are present. The second sub-score considers the model's performance on datasets without any anomalous data, i.e., its ability to recognize normal behavior accordingly. The third sub-score assesses classification performance on aggregated events, by applying an eventwise classification measure.

3.2.2. Score Definition

Coverage

In order to measure the coverage—i.e., the classification performance on a time series with anomalies—the F_β-score as described by [34] is used. At first, the prediction time frame of a given anomaly event dataset is filtered. All data points with an abnormal status ID according to Table 2 are ignored. This is important because these data points are usually very easy to detect. Moreover, the wind farm operator is already informed about the abnormal behavior through the status ID. Thus, these data points are irrelevant in the context of predictive maintenance and they would dilute the score. Now, let $\mathbf{g}$ be the ground truth of all data points with a normal status ID within the prediction time frame and $\mathbf{p}$ is the corresponding prediction of an early fault detection model. The F_β-score is then defined as

$$F_\beta(\mathbf{g},\mathbf{p}) = \frac{(1+\beta^2)\cdot tp(\mathbf{g},\mathbf{p})}{(1+\beta^2\cdot tp(\mathbf{g},\mathbf{p}) + \beta^2\cdot fn(\mathbf{g},\mathbf{p}) + fp(\mathbf{g},\mathbf{p}))}, \tag{1}$$

where $tp(\mathbf{g},\mathbf{p})$ is the number of true positives based on $\mathbf{g}$ and $\mathbf{p}$, $fn(\mathbf{g},\mathbf{p})$ is the number of false negatives, and $fp(\mathbf{g},\mathbf{p})$ is the number of false positives. In this case, a value of $\beta = \frac{1}{2}$ is chosen to give more weight to precision than recall, thereby penalizing excessive false positives.

Accuracy

To measure the performance on datasets that exclusively contain normal behavior, the Acc as described by [35] is used. Since there are no true positives for the prediction time frames of those datasets, F_β from Equation (1) would always be 0. With the same reasoning as in the Section Coverage, only data points with a normal status ID are relevant. Let $\mathbf{g}$ be the ground truth of all data points with a normal status ID within the prediction time frame, and $\mathbf{p}$ is the corresponding prediction. Then Acc is calculated as

$$Acc(\mathbf{g},\mathbf{p}) = \frac{tn(\mathbf{g},\mathbf{p})}{fp(\mathbf{g},\mathbf{p}) + tn(\mathbf{g},\mathbf{p})}. \tag{2}$$

where $tn(\mathbf{g}, \mathbf{p})$ is the number of true negatives based on $\mathbf{g}$ as well as $\mathbf{p}$, and $fp(\mathbf{g}, \mathbf{p})$ is the number of false positives. Note that, in contrast to the standard accuracy, there are no true positives and false negatives, since only datasets containing normal behavior are considered and data points with an abnormal status ID are excluded.

Reliability

False alarms on an event basis are taken into account by the event based F_β-score (EF_β). First, each time series prediction needs to be classified as either "anomaly event detected" or "normal behavior". For this, we first calculate the maximum "criticality", which is a counter-like measure. Given the prediction timestamps $t_1, \ldots, t_N$, the status information $s_{t_i} \in \{0, 1\}$ where 1 corresponds to a normal status and 0 to an abnormal status, and the prediction $p_{t_i} \in \{0, 1\}$ where 1 represents a detected anomaly and 0 no detected anomaly for $i = 1, \ldots, N$, the criticality is computed by Algorithm 1.

Algorithm 1 Criticality Algorithm

```
crit ← [0, 0, ..., 0] ∈ ℕ^(N+1)
for i ∈ {1, ..., N} do
    if s_{t_i} = 0 then
        if p_{t_i} = 1 then
            crit[i] ← crit[i − 1] + 1
        else
            crit[i] ← max{crit[i − 1] − 1, 0}
        end if
    else
        crit[i] ← crit[i − 1]
    end if
end for
crit ← crit[1 : N]
```

After calculating the criticality for the entire prediction time frame, the maximum criticality is compared to a threshold t_c, which we set to 72. The idea behind this threshold is that, in order to reach a criticality of 72, the algorithm must either detect at least 72 anomalies in a row, which equates to 12 h of consecutive anomalies, or even more anomalies in the case of non-consecutive detected anomalies. Setting the threshold at 72 was found to be the most appropriate for all 95 time series in this dataset and generally depends on the length of the time series and the use-case specific definitions of detected anomaly events. For shorter events, a lower threshold is more appropriate.

If the threshold is exceeded, the prediction is counted as a detected anomaly event (i.e., an alarm was raised). If the maximum criticality is below 72, the prediction is counted as a normal event prediction (i.e., no alarm). These event prediction labels are then compared to the true dataset labels and the F_β-score is calculated as defined in Equation (1), where $\beta = \frac{1}{2}$ is usually chosen to penalize false positives further.

Earliness

Similar to the F_β-score, the second sub-score—the WS—is also only applied to anomaly events. As a modified version of the weighted score defined in the Numenta benchmark [6], this score weights detected anomalies during the beginning of defined anomaly events higher than detected anomalies at the end of the event time frame. However, instead of discarding additional detected anomalies within the event time frame, all detected anomalies will be considered with positive weights. The piecewise linear function shown in Figure 1 is used to weight the predicted timestamps. In the first half of the event, all detected anomalies are assigned a weight of 1. In the second half of the event, the weights decrease linearly to 0, as the detected anomalies become less important to the wind farm operator the closer they are to the actual turbine fault.

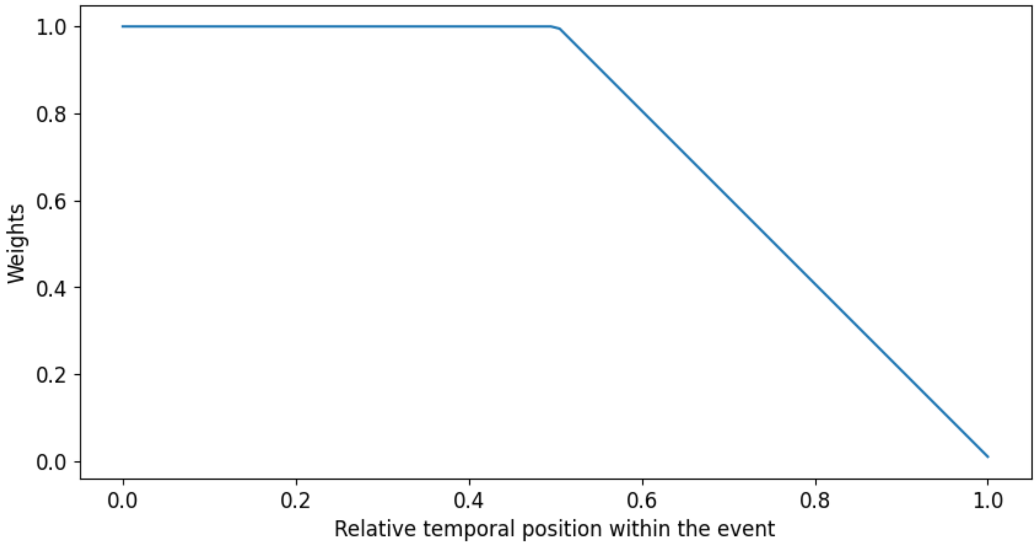

Figure 1. Weight function of the weighted score (WS) for early anomaly detection.

In order to apply this weighting function to anomaly events of different lengths, the length of each event is used to convert the event timestamps to the relative position in the interval $[0, 1]$, where 0 corresponds to the beginning of the anomaly event and 1 corresponds to the end of the event, i.e., the start of a downtime or a fault as detected by other systems. Let the consecutive timestamps of an anomaly event a be $t_1 < t_2 < \cdots < t_M$ while $\mathbf{p}_a := (p_{t_1}, \ldots, p_{t_N}) \in \{0, 1\}^N$ denotes the corresponding prediction, where 1 marks a detected anomaly and 0 means no detected anomaly. The WS of this anomaly event is then calculated as

$$WS(\mathbf{p}_a) = \frac{\sum_{i=1}^{M} w_{t_i} \cdot p_{t_i}}{\sum_{i=1}^{M} w_{t_i}} \tag{3}$$

where $w_{t_i} \in [0, 1]$ is the weight for the timestamp t_i.

CARE

Finally, the CARE score is calculated by combining the four sub-scores. This is done by calculating the event based score EF_β and the averages $\overline{F_\beta}$, $\overline{WS}$, and $\overline{Acc}$. Here, $\overline{F_\beta}$ is the arithmetic mean over all F_β-scores of datasets containing an anomaly event, $\overline{WS}$ is the arithmetic mean over all WS of datasets containing an anomaly event, and $\overline{Acc}$ is the average over all Acc of datasets representing normal behavior.

The final CARE score takes two special cases into account. If no anomaly events were detected at all, the CARE score will be 0. Also, if $\overline{Acc}$ falls below 0.5, the predictions are worse than uniformly distributed random predictions. In this case, the final score will be equal to $\overline{Acc}$. Outside of these two special cases, the CARE score is defined by a weighted average WA of all sub-scores:

$$WA := \frac{1}{\sum_{i=1}^{4} \omega_i} \left(\omega_1 \overline{F_\beta} + \omega_2 \overline{WS} + \omega_3 EF_\beta + \omega_4 \overline{Acc}\right), \tag{4}$$

where we choose $\omega_1 = \omega_2 = \omega_3 = 1$ and $\omega_4 = 2$ in order to weight the normal datasets to the same degree as datasets containing an anomaly event, and $\beta = \frac{1}{2}$.

To summarize, the CARE score is defined by

$$CARE := \begin{cases} 0, & \text{if no anomalies were detected} \\ \overline{Acc}, & \text{if } \overline{Acc} < 0.5 \\ WA, & \text{else.} \end{cases} \tag{5}$$

3.3. Comparison to Other Scoring Methods

The CARE score combines classic and advanced scoring methods to enhance early fault detection in predictive maintenance, focusing on anomaly detection while minimizing false alarms. Contrary to measures like the AUC-ROC, all four sub-scores considered are threshold-specific performance measures. This is a property that makes comparisons of algorithms for the sake of early fault detection less significant, but in exchange it raises the significance of the score in a real-world operative predictive maintenance setting. As mentioned in [31], operators of wind farms and other assets are primarily interested in accurately detecting anomalies while minimizing false alarms. Additionally, this enables comparison of the same NBM with different threshold methods. In this subsection, the CARE score is compared to known scores suited to anomaly detection models and specific early fault detection models.

3.3.1. F_β-Score

While the F_β-score (Equation (1)) is well suited for the pointwise evaluation of anomaly detection models regarding their ability to detect anomalies within a dataset, it is not able to express the ability to classify normal behavior in datasets where the number of anomalies is minimal or even absent. Since the ability to correctly recognize normal behavior over potentially long periods of time is a required property for reliable early fault detection models, the F_β-score on its own is not sufficient for the evaluation of early fault detection models.

3.3.2. Acc-Score

In contrast to the F_β-score, the Acc-score (Equation (2)) also considers the number of true negatives for the evaluation, which empowers it to correctly evaluate the aforementioned long periods of normal behavior in a meaningful way. However, the Acc score is inferior to the F_β-score when it comes to the task of assessing the anomaly detection performance, since the latter takes into account true positives and allows for use-case dependent weighting of precision and recall. Consequently, the accuracy of an early fault detection model is not sufficient for evaluation in itself.

3.3.3. Composite F-Score

In the context of early fault detection, a good pointwise classification is mandatory, but the ability to detect events is comparably meaningful. Scores such as the composite F-score proposed in [31] aim to combine pointwise and eventwise evaluations, which is an idea that the CARE score also builds upon. However, the composite F-score, simlarly to the classic F-score, cannot express the ability to accurately recognize normal behavior. Therefore, it is not comprehensive enough to evaluate early fault detection models.

As a combination of the classic F_β-score, the Acc-score, and the eventwise F_β-score, the CARE score reflects on the ability to detect anomalies and the ability to recognize normal behavior in a pointwise manner, as well as on the event level. This makes the CARE score more effective in assessing early fault detection model performances than the other scores.

3.3.4. Numenta Score

To conclude the comparison of the CARE score to other scoring methods, the Numenta score [6] must be mentioned as well. Since this score also inspired the creation of the CARE score by incorporating an earliness reward, it addresses the evaluation of early fault detection algorithms in a different way than classic classification scores. Nevertheless, the

Numenta score may yield a high positive value when only a single timestamp is flagged as an anomaly, when it is an early detection. In such a situation, the context of the entire event is overlooked. In contrast, the CARE score would assign a lower value to such cases, reflecting its emphasis on recognizing the full scope of the event rather than relying solely on isolated detections.

3.3.5. Score Modification

In some cases, it is beneficial to adapt the CARE score for different use cases. As the challenges [27,28] show, maintenance costs for different turbine faults are often considered when assessing the performance of early fault detection models. The CARE score can be adjusted to take such costs into account by replacing $\overline{F_\beta}$ and $\overline{WS}$ by weighted averages.

Let $a_1, \ldots, a_N$ be all datasets containing anomaly events and $\boldsymbol{\omega} := (\omega_1, \ldots, \omega_N)$ be the cost-based importance weights of each anomaly. The weighted averages are then defined by

$$\overline{F_\beta^\omega} := \frac{1}{\sum_{i=1}^N \omega_i} \sum_{i=1}^N \omega_i \cdot F_\beta(\mathbf{g}_{a_i}, \mathbf{p}_{a_i}) \tag{6}$$

$$\overline{WS^\omega} := \frac{1}{\sum_{i=1}^N \omega_i} \sum_{i=1}^N \omega_i \cdot WS(\mathbf{p}_{a_i}), \tag{7}$$

where $\mathbf{g}_{a_i}$ is the ground truth for the prediction time frame of dataset a_i and $\mathbf{p}_{a_i}$ is the corresponding model prediction. Additionally, the weights in Equation (4) can be altered to better suit the use case.

For the following mini-benchmark section, no further modification of weights are made, since the necessary information about maintenance costs for each fault in the dataset are not available for wind farms B and C.

4. Results

For the purpose of using the dataset for benchmarks of algorithms for fault detection in WT, and showcasing the newly defined CARE score, python implementations of an NBM based on an AE approach and a simple isolation forest approach are compared to the trivial strategies “all anomaly”, “all normal”, and “random”.

4.1. Simple Approaches

While “all anomaly” just classifies every timestamp as an anomaly and “all normal” does the opposite, “random” assigns the prediction for every timestamp independently based on a 50/50 probability. The slightly more complex isolation forest approach uses the implementation from the python package “sklearn” [36] with “n_estimators” = 100 and “contamination” = 0.09, as well as a principal component analysis in order to reduce the dimensionality of the input data, such that 99% of the variance is kept. All hyperparameters were selected by manual tests.

4.2. Autoencoder Approach

The AE NBM is a more sophisticated early fault detection approach. The model used is a further developed version of the AD procedure described in [37]. This model consists of an AE model trained on data representing normal behavior and a calibrated threshold to detect anomalies. The AE models for each wind farm contain three to five hidden layers and are optimized using the Adam algorithm. The hyperparameters, such as the number of units in the hidden layers, the learning rate, and the amount of noise to regularize the AE, were adjusted using the python package “Optuna” [38]. An overview of the model hyperparameters for the baseline models is provided in Table 3. The AE is trained on 75% of the normal training data, while 25% of the data are randomly selected for validation. The training lasts at most 200 epochs with an early stopping option, which is triggered if the L2-norm of the reconstruction error on the validation data does not decrease for 3 consecutive epochs. Based on a calibrated threshold, predicted timestamps are assigned the label "anomaly" if the L2-norm of the corresponding reconstruction error exceeds the threshold; otherwise, they will receive the label "normal".

The calibration of the threshold differs depending on the wind farm. For wind farms A and B, an adaptive threshold is used, inspired by the work in [30,39]. Here, a NN regression model is used to learn the mapping of the AE input data to the L2-norm of RE. The NN consists of 3 layers with around 20 to 40 units in the hidden layer, ReLU activations and the Adam algorithm is used for optimization. For training, the same data that the AE is validated on is used, i.e., the part of the validation data representing normal behavior. The training lasts, at most, 300 epochs, with the same early stopping mechanism as in the AE training. During prediction, the new input data are evaluated by the NN and provide an expected RE ϵ. This expected RE is increased by adding the sensitivity parameter $\gamma \in [0, \infty]$ and then compared to the actual RE of the AE model. While parameter γ has to be optimized for each wind farm separately, values from the interval $[0.2, 0.4]$ seem to be a good fit for the provided datasets. If the actual RE is larger than $\epsilon + \gamma$, the corresponding timestamp is detected as an "anomaly". For the determination of the optimal number of units in the hidden layer of the NN and the value for γ, a hyperparameter optimization with "Optuna" is used. The final hyperparameter values of the thresholds used for this evaluation can be found in Table 3.

For wind farm C, a fixed threshold is calibrated. For this, the L2-norm of the reconstruction errors (anomaly score) of all AE validation data is computed. Afterwards, the constant threshold is selected by iterating over the calculated anomaly score values and choosing the value that maximizes the $F_{\frac{1}{2}}$-score based the ground truth defined by the normal behavior labels, which can be derived from the status IDs as shown in Table 2.

Table 3. Hyperparameters of the autoencoder-base anomaly detection model.

	Wind Farm A	**Wind Farm B**	**Wind Farm C**
Number of units in hidden layers	44, 25, 4, 25, 44	40, 15, 40	133, 83, 20, 83, 133
Learning rate	0.0018	0.003	0.0056
Noise	0.06	0	0
Batch size	64	64	64
Threshold	adaptive	adaptive	max. $F_{\frac{1}{2}}$-score
Threshold NN hidden layers	23	36	-
γ	0.344	0.234	-

Finally, the AE NBM is supplemented with an additional data filter. In order to remove potentially implausible data from the training data of the AE, a status based on wind speed and power enhances the normal behavior labels given by the turbine operational status from Table 2. For the determination of the new status information, timestamps are marked as not normal if the wind speed is within the normal operation range of the WT and the power is close or equal to 0.

4.3. Scoring of the Approaches

At first, the four sub-scores of the CARE score are evaluated for all five approaches. The results are visualized in Figure 2. While "all anomaly" obviously performs well in detecting anomalies, it, of course, has the worst possible accuracy on normal data. The opposite is, of course, the case for "all normal". The isolation forest behaves similar to "all anomaly", since it detects a lot of anomalies but performed very poorly in recognizing normal behavior. Finally, the AE approach has a high accuracy on normal data and a good performance on the event based $F_{\frac{1}{2}}$-score, but it suffers a little bit from an overall low number of detected anomalies.

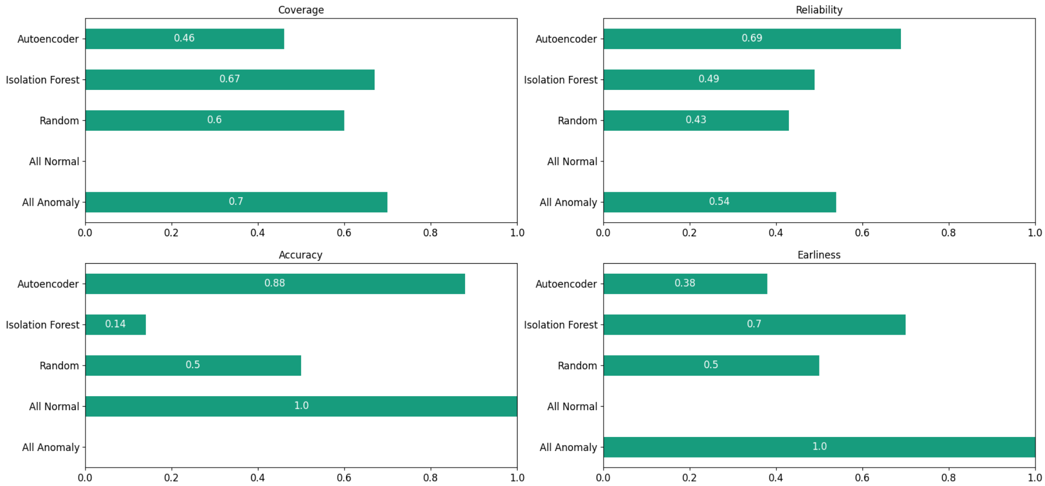

Figure 2. Sub-scores of CARE score over all 95 sub-datasets for a few selected approaches.

When it comes to the final CARE score shown in Figure 3, the trivial strategies "all anomaly" and "all normal" both get the score 0 because they run into the special cases described at the end of Section 3.2.1. The strategy "random" gets the CARE score of 0.5 and sets the lower bound to beat for good any anomaly detection algorithms. The isolation forest approach does not outperform that threshold since it is not able to recognize normal behavior appropriately with the chosen parameter configuration. With a score of 0.66, the AE approach represents a good anomaly detector since it is able to detect anomalies while also recognizing normal behavior very well and having a good classification performance on aggregated events.

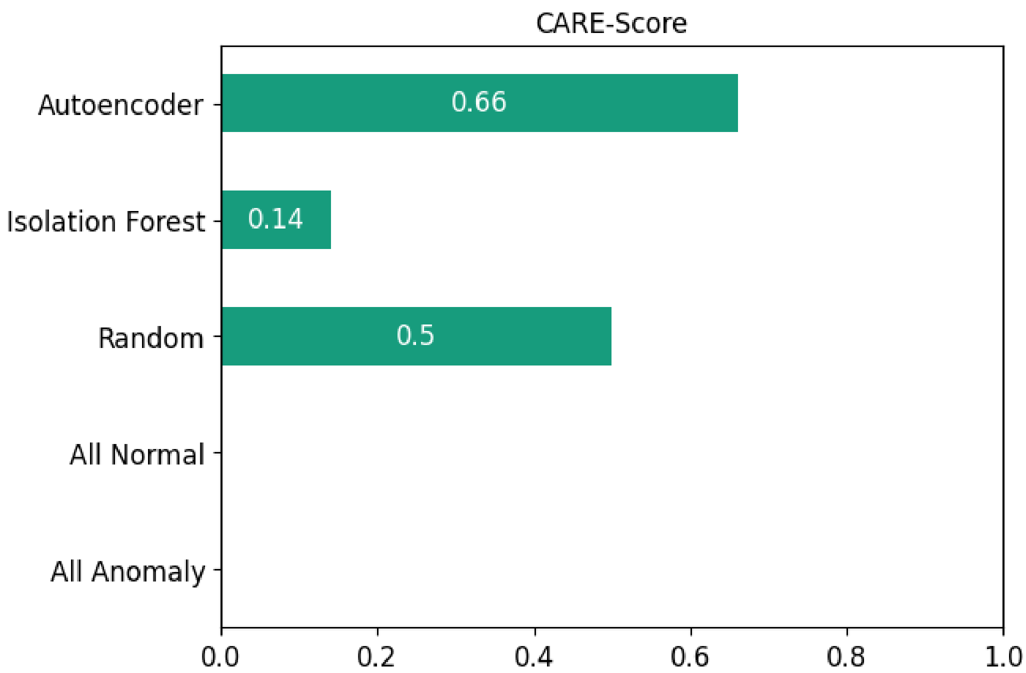

Figure 3. CARE score evaluation over all 95 sub-datasets for a few selected approaches.

5. Discussion

In order to give a complete overview about the published dataset, the inherent limitations of the dataset are discussed within this section. Although the dataset is the most detailed and extensive benchmark dataset for early fault detection on wind turbines yet, there are some limitations to the usage.

Since the dataset contains many different types of WT failures with a relatively low number of examples per type, it is best suited for the evaluation of NBM in the context of early fault detection. Direct classification algorithms and failure specific approaches are not favored by the the low number of examples for each failure type within the dataset.

Another limiting factor for the significance of the dataset are the derived anomaly and failure labels. As the provided dataset stems from real-world data, the labeled failures may be incorrect in some cases. Due to the largely unknown build-up time of wind turbine failures and environmental influences, the start of anomalies cannot always be determined with certainty. To counteract this possible issue and to ensure the quality of the dataset, the probability of label errors was reduced as much as possible by considering different sources of information and feedback from the wind farm operators during the labeling process, as described in Section 3.1.3.

6. Conclusions

With the purpose of reducing the limitations that come with the lack of public data for WT AD benchmarks, a new dataset was published. Composed out of multiple WT across three wind farms, the dataset shows greater detail in anomaly labels and additional information than datasets that are currently available. By formulating requirements for benchmark datasets for early fault detection in WT, the data quality and the ability to test for generalization of early fault detection models were ensured. The balanced nature of the dataset, with similar amounts of anomalous data and examples of normal behavior, allows for more detailed and meaningful comparison studies of early fault detection algorithms.

Furthermore, we proposed an evaluation method, the CARE score, that fully uses the informational depth the dataset provided. By considering the four key aspects of a good early fault detection model—detecting many anomalies, early detection, few false alarms, and correct recognition of normal behavior—the CARE score provides a measure for the all-around performance of a good early fault detection model for predictive maintenance on WT.

To demonstrate the combination of dataset and CARE score, a test evaluation was conducted. A sophisticated early fault detection algorithm was compared to the popular and simpler isolation forest and three trivial strategies. This evaluation shows the importance of not neglecting normal behavior recognition while trying to detect as many anomalies as possible, as it was the case for the isolation forest approach.

As a subject for future research, the provided dataset and scoring method can be used to compare a wide range of WT early fault detection models on an equal and transparent basis in order to push the progress in this field further and find good early fault detection algorithms.

To further enhance the development of benchmark datasets in the field of WT early fault detection, the authors strongly encourage others to share their data. Increasing the availability of high-quality benchmark datasets will facilitate more comprehensive and rigorous evaluations of early fault detection models in this domain.

Author Contributions: Conceptualization, C.G. and C.M.A.R.; methodology, C.G. and C.M.A.R.; software, C.G. and C.M.A.R.; validation, C.G., C.M.A.R. and S.F.; formal analysis, C.G. and C.M.A.R.; investigation, C.G. and C.M.A.R.; resources, C.G., C.M.A.R. and S.F.; data curation, C.M.A.R. and S.F.; writing—original draft preparation, C.G. and C.M.A.R.; writing—review and editing, S.F.; visualization, C.G.; supervision, S.F.; project administration, C.M.A.R.; funding acquisition, S.F. All authors have read and agreed to the published version of the manuscript.

Funding: The development of methods presented was funded by the German Federal Ministry for Economic Affairs and Climate Action (BMWK).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in the study are openly available on "Zenodo": https://zenodo.org/records/14006163 (accessed on 29 October 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

WT	wind turbine
AD	anomaly detection
ML	machine learning
MSE	mean square error
NBM	normal behavior model
SCADA	Supervisory Control and Data Acquisition
WS	weighted score
Acc	accuracy score
O&M	Operation & Maintenance
CARE	Coverage Accuracy Reliability Earliness
NN	neural network
AE	autoencoder
RE	reconstruction error
AUC	area under the curve
ROC	receiver operating characteristic curve

Note

[1] The data can be found on "Zenodo": https://doi.org/10.5281/zenodo.14006163 (accessed on 29 October 2024).

References

1. Tautz-Weinert, J.; Watson, S.J. Using SCADA data for wind turbine condition monitoring—A review. *IET Renew. Power Gener.* **2017**, *11*, 382–394. [CrossRef]
2. Helbing, G.; Ritter, M. Deep Learning for fault detection in wind turbines. *Renew. Sustain. Energy Rev.* **2018**, *98*, 189–198. [CrossRef]
3. Pandit, R.; Wang, J. A comprehensive review on enhancing wind turbine applications with advanced SCADA data analytics and practical insights. *IET Renew. Power Gener.* **2024**, *18*, 722–742 [CrossRef]
4. Latiffianti, E.; Sheng, S.; Ding, Y. Wind Turbine Gearbox Failure Detection Through Cumulative Sum of Multivariate Time Series Data. *Front. Energy Res.* **2022**, *10*, 904622. [CrossRef]
5. Han, S.; Hu, X.; Huang, H.; Jiang, M.; Zhao, Y. ADBench: Anomaly Detection Benchmark. In *Proceedings of the Advances in Neural Information Processing Systems*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2022; Volume 35, pp. 32142–32159.
6. Lavin, A.; Ahmad, S. Evaluating Real-Time Anomaly Detection Algorithms—The Numenta Anomaly Benchmark. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015; pp. 38–44. [CrossRef]
7. Pang, G.; Shen, C.; Cao, L.; Hengel, A.V.D. Deep Learning for Anomaly Detection: A Review. *ACM Comput. Surv.* **2022**, *54*, 1–38. [CrossRef]
8. Schmidl, S.; Wenig, P.; Papenbrock, T. Anomaly detection in time series: A comprehensive evaluation. *Proc. VLDB Endow.* **2022**, *15*, 1779–1797. [CrossRef]
9. Zhang, C.; Hu, D.; Yang, T. Research of artificial intelligence operations for wind turbines considering anomaly detection, root cause analysis, and incremental training. *Reliab. Eng. Syst. Saf.* **2024**, *241*, 109634. [CrossRef]
10. Yang, L.; Zhang, Z. A Conditional Convolutional Autoencoder-Based Method for Monitoring Wind Turbine Blade Breakages. *IEEE Trans. Ind. Inform.* **2021**, *17*, 6390–6398. [CrossRef]
11. Morrison, R.; Liu, X.; Lin, Z. Anomaly detection in wind turbine SCADA data for power curve cleaning. *Renew. Energy* **2022**, *184*, 473–486. [CrossRef]
12. Schröder, L.; Dimitrov, N.K.; Verelst, D.R.; Sørensen, J.A. Using Transfer Learning to Build Physics-Informed Machine Learning Models for Improved Wind Farm Monitoring. *Energies* **2022**, *15*, 558. [CrossRef]
13. McKinnon, C.; Carroll, J.; McDonald, A.; Koukoura, S.; Infield, D.; Soraghan, C. Comparison of New Anomaly Detection Technique for Wind Turbine Condition Monitoring Using Gearbox SCADA Data. *Energies* **2020**, *13*, 5152. [CrossRef]
14. Jia, X.; Han, Y.; Li, Y.; Sang, Y.; Zhang, G. Condition monitoring and performance forecasting of wind turbines based on denoising autoencoder and novel convolutional neural networks. *Energy Rep.* **2021**, *7*, 6354–6365. [CrossRef]
15. de Sá, F.P.G.; Brandão, D.N.; Ogasawara, E.; Coutinho, R.d.C.; Toso, R.F. Wind Turbine Fault Detection: A Semi-Supervised Learning Approach with Automatic Evolutionary Feature Selection. In Proceedings of the 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), Niteroi, Brazil, 1–3 July 2020; pp. 323–328. [CrossRef]

16. Udo, W.; Muhammad, Y. Data-Driven Predictive Maintenance of Wind Turbine Based on SCADA Data. *IEEE Access* **2021**, *9*, 162370–162388. [CrossRef]
17. Tang, Z.; Shi, X.; Zou, H.; Zhu, Y.; Yang, Y.; Zhang, Y.; He, J. Fault Diagnosis of Wind Turbine Generators Based on Stacking Integration Algorithm and Adaptive Threshold. *Sensors* **2023**, *23*, 6198. [CrossRef]
18. Jankauskas, M.; Serackis, A.; Šapurov, M.; Pomarnacki, R.; Baskys, A.; Hyunh, V.K.; Vaimann, T.; Zakis, J. Exploring the Limits of Early Predictive Maintenance in Wind Turbines Applying an Anomaly Detection Technique. *Sensors* **2023**, *23*, 5695. [CrossRef]
19. Barber, S.; Izagirre, U.; Serradilla, O.; Olaizola, J.; Zugasti, E.; Aizpurua, J.I.; Milani, A.E.; Sehnke, F.; Sakagami, Y.; Henderson, C. Best Practice Data Sharing Guidelines for Wind Turbine Fault Detection Model Evaluation. *Energies* **2023**, *16*, 3567. [CrossRef]
20. Barber, S.; Lima, L.A.M.; Sakagami, Y.; Quick, J.; Latiffianti, E.; Liu, Y.; Ferrari, R.; Letzgus, S.; Zhang, X.; Hammer, F. Enabling Co-Innovation for a Successful Digital Transformation in Wind Energy Using a New Digital Ecosystem and a Fault Detection Case Study. *Energies* **2022**, *15*, 5638. [CrossRef]
21. Nassif, A.B.; Talib, M.A.; Nasir, Q.; Dakalbab, F.M. Machine Learning for Anomaly Detection: A Systematic Review. *IEEE Access* **2021**, *9*, 78658–78700. [CrossRef]
22. Ruff, L.; Kauffmann, J.R.; Vandermeulen, R.A.; Montavon, G.; Samek, W.; Kloft, M.; Dietterich, T.G.; Muller, K.R. A Unifying Review of Deep and Shallow Anomaly Detection. *Proc. IEEE* **2021**, *109*, 756–795. [CrossRef]
23. Effenberger, N.; Ludwig, N. A collection and categorization of open-source wind and wind power datasets. *Wind Energy* **2022**, *25*, 1659–1683. [CrossRef]
24. Menezes, D.; Mendes, M.; Almeida, J.A.; Farinha, T. Wind Farm and Resource Datasets: A Comprehensive Survey and Overview. *Energies* **2020**, *13*, 4702. [CrossRef]
25. Letzgus, S. Wind Turbine SCADA Open Data. 2023. Available online: https://github.com/sltzgs/Wind_Turbine_SCADA_open_data (accessed on 13 March 2024).
26. EDP Inovação. *EDPR Wind Farm Open Data: Wind Turbine SCADA Signals and Historical Failure Logbook from 2016 and 2017*; EDP: Lisbon, Portugal, 2018.
27. EDP Inovação. *Hack the Wind: Wind Turbine Failures Detection*; EDP: Lisbon, Portugal, 2018.
28. Eastern Switzerland University of Applied Sciences. We Do Wind: EDP Challenges Space. 2021. Available online: https://www.wedowind.ch/spaces/edp-challenges-space (accessed on 18 March 2024).
29. Wu, R.; Keogh, E.J. Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 2421–2429. [CrossRef]
30. Chen, H.; Liu, H.; Chu, X.; Liu, Q.; Xue, D. Anomaly detection and critical SCADA parameters identification for wind turbines based on LSTM-AE neural network. *Renew. Energy* **2021**, *172*, 829–840. [CrossRef]
31. Garg, A.; Zhang, W.; Samaran, J.; Savitha, R.; Foo, C.S. An Evaluation of Anomaly Detection and Diagnosis in Multivariate Time Series. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 2508–2517. [CrossRef]
32. Carrasco, J.; López, D.; Aguilera-Martos, I.; García-Gil, D.; Markova, I.; García-Barzana, M.; Arias-Rodil, M.; Luengo, J.; Herrera, F. Anomaly detection in predictive maintenance: A new evaluation framework for temporal unsupervised anomaly detection algorithms. *Neurocomputing* **2021**, *462*, 440–452. [CrossRef]
33. Stetco, A.; Dinmohammadi, F.; Zhao, X.; Robu, V.; Flynn, D.; Barnes, M.; Keane, J.; Nenadic, G. Machine learning methods for wind turbine condition monitoring: A review. *Renew. Energy* **2019**, *133*, 620–635. [CrossRef]
34. Flach, P.; Kull, M. Precision-Recall-Gain Curves: PR Analysis Done Right. In *Advances in Neural Information Processing Systems 28, Proceedings of the NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December* 2015; Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2015; Volume 1, pp. 838–846.
35. Metz, C.E. Basic principles of ROC analysis. *Semin. Nucl. Med.* **1978**, *8*, 283–298. [CrossRef]
36. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
37. Roelofs, C.M.; Lutz, M.A.; Faulstich, S.; Vogt, S. Autoencoder-based anomaly root cause analysis for wind turbines. *Energy AI* **2021**, *4*, 100065. [CrossRef]
38. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 2623–2631.
39. Zhao, H.; Liu, H.; Hu, W.; Yan, X. Anomaly detection and fault analysis of wind turbine components based on deep learning network. *Renew. Energy* **2018**, *127*, 825–834. [CrossRef]