Sustainable Energy, Grids and Networks 32 (2022) 100803

Contents lists available at ScienceDirect

Sustainable Energy, Grids and Networks

journal homepage: www.elsevier.com/locate/segan

# Robust disturbance classification in power transmission systems with denoising recurrent autoencoders

André Kummerow *, Mohammad Dirbas, Cristian Monsalve, Steffen Nicolai, Peter Bretschneider

*Fraunhofer IOSB, IOSB-AST, Fraunhofer Institute of Optronics, System Technologies and Image Exploitation, Fraunhofer Center for Machine Learning, Am Vogelherd 90, 98693 Ilmenau, Germany*

ARTICLE INFO

*Article history:*
Received 9 December 2021
Received in revised form 3 May 2022
Accepted 5 June 2022
Available online 11 June 2022

*Keywords:*
Phasor measurements
Disturbance classification
Denoising autoencoders
Recurrent neural networks
Data quality

ABSTRACT

The automated classification of grid disturbances based on phasor measurements is a key technology for the reliable operation of power transmission systems. The predominant use of simulated training data limits the applicability of existing classification approaches due to the missing consideration of measurement errors or data quality issues. To mitigate these shortcomings, this study presents a robust disturbance classification procedure incorporating denoising recurrent autoencoders within a novel two-stage training approach. The developed disturbance classification procedure is evaluated for different noise characteristics and dataset combinations created with an optimization based error model. Experimental results based on a generic power transmission system show superior performance of the proposed two-stage design compared to a conventional, one-stage model training.



## 1. Introduction

### 1.1. Disturbance classification with simulated training data

A real-time analysis of phasor measurements enables an automated identification and localization of grid disturbances (e.g. generator trips, short circuits, outages of renewable energy resources) and is an important monitoring function for today's power transmission systems [1–7]. Typically, multiple phasor measurement unit (PMU) devices are deployed at several substations in the grid to provide high resolution and time-synchronized voltage, current and frequency data streams (wide area monitoring system). A machine learning based detection of grid disturbances enables a fast and reliable system operation by activating appropriate countermeasures (e.g. remedial action schemes) to mitigate cascading failures or large-scale supply disruptions [8–10].

Common classification approaches rely on a sufficient and representative training database including all relevant disturbance events. The exclusive use of historical PMU measurements is not practical for training and evaluating the proposed classification approach, as this would require many measured disturbance events occurring during system operation and would not capture previously unobserved disturbance event of future operation conditions. Therefore, dynamic grid simulations are used to synthetically generate the necessary PMU signals for various disturbance events and steady state conditions. As a consequence, simulation model deficiencies (e.g. inaccurate model parameters) can lead to large signal deviations and misclassifications during the application phase (inference). This especially accounts for the modelling of the PMU devices and their associated measurement errors, which are not included in current dynamic simulation models. As in other application fields, machine learning models and especially artificial neural networks suffer from deviations between training and test data characteristics and may lead to misclassifications during the inference.

Based on preliminary work [1,2], the basic architecture of a generic PMU based disturbance classification system is shown in Fig. 1.

Dynamic simulations $\boldsymbol{X}_{\mathrm{S}}$ are used during the training phase and phasor measurements $\boldsymbol{X}_{\mathrm{P}}$ from a wide area monitoring system during the application phase. Each input matrix consists of one or more samples from multiple PMU signals over a given time period. The classification approach analyses the normalized PMU signals $\boldsymbol{X}_{\mathrm{N}}$ and creates a representative feature representation $\underline{f}$ to estimate the location $\hat{y}_{\mathrm{Loc}}$ and type $\hat{y}_{\mathrm{Type}}$ of a particular disturbance event. In case of artificial neural networks, the feature extraction and classification modules can be learned together via backpropagation by minimizing an empirical error.

### 1.2. Error characteristics of phasor measurements

The IEEE C37.118 standard (since 2018: IEC 60255-118-1-2018 [11]) describes different performance requirements for the

* Corresponding author.
*E-mail address:* andre.kummerow@iosb-ast.fraunhofer.de (A. Kummerow).

https://doi.org/10.1016/j.segan.2022.100803

**Nomenclature**

| | |
|---|---|
| $\overline{C}$ | class configuration |
| $\underline{e}$ | error signal/component |
| $\underline{f}$ | feature vector |
| $\underline{h}_E^t$, $\underline{h}_D^t$ | encoder (E) and decoder (D) hidden state vector |
| $K$ | number of disturbance classes |
| $\mathcal{L}$ | Laplacian distribution |
| $M$ | number of PMU devices/sensors |
| $\mathcal{N}$ | Gaussian distribution |
| $\widehat{p}$ | (estimated) posterior probability vector |
| $\overline{R}$ | number of GRU hidden units |
| $s^t$ | score value at time step $t$ |
| $t$ | time step |
| $T$ | Number of time steps per sample |
| $x_F^t$, $x_U^t$ | frequency (F) and voltage magnitude (U) value at time step $t$ |
| $\boldsymbol{X}_P$, $\boldsymbol{X}_P^N$ | raw and normalized (N) measured observation matrix |
| $\boldsymbol{X}_S$, $\boldsymbol{X}_S^N$ | raw and normalized (N) simulated observation matrix |
| $y$ | (true) disturbance class label |
| $\hat{y}_{Loc}$ | (estimated) disturbance location |
| $\hat{y}_{Type}$ | (estimated) disturbance type |
| $\alpha^t$ | attention weight at time step $t$ |
| $\underline{\theta}_E$, $\underline{\theta}_D$, $\underline{\theta}_C$ | trainable parameters of the encoder (E), decoder (D) and classifier (C) |

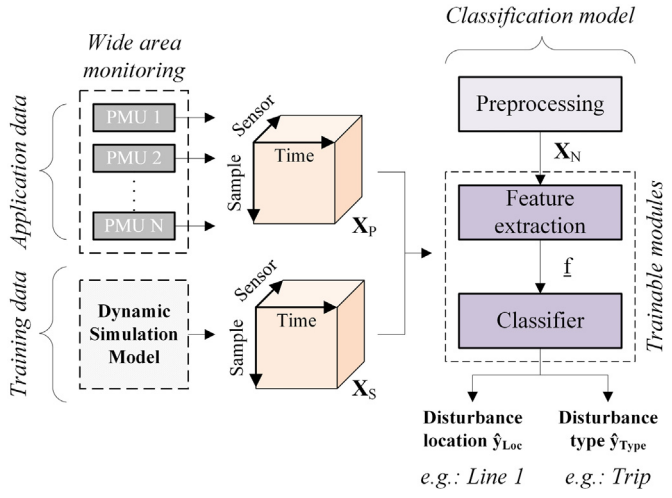


**Fig. 1.** Main principle of a PMU based disturbance classification system using simulated training data.

transmission of phasor measurement values (e.g. the total vector error) but does not adequately describe measurement errors by PMU devices. Several studies [12–16] investigated the characteristics of PMU induced measurement errors, which can be caused mainly by complex grid operations due to a high number of different control devices and malfunctions of PMU ancillary components (e.g. drifting internal clocks, low-accuracy instrument transformers). In most cases, the error component follows a white Gaussian noise or Laplacian distribution with a certain signal-to-noise-ratio (SNR) and is additionally interspersed with missing values. Furthermore, data anomalies or outliers as well as low-frequency oscillations (< 1 Hz) can additionally bias PMU measurement signals. Based on literature findings, Table 1 summarizes the most relevant information regarding PMU measurement error characteristics, which are used to build up the optimization based error model in paragraph 2.

**Table 1**
Overview of typical error characteristics in PMU signals.

| Characteristic | Value/ Range | References |
|---|---|---|
| *Gaussian noise* | | |
| SNR | 30–50 dB | [12–14] |
| Sample mean | 0 | |
| *Laplacian noise* | | |
| SNR | 30–50 dB | [14,15] |
| Sample mean | 0 | |
| Scale parameter | 0.001 | |
| *Missing values* | | |
| Ratio | 2% | [12,13,16] |

### 1.3. Main contributions and paper organization

This study provides a noise-tolerant recurrent neural network based PMU based disturbance classification method, taking into account Gaussian and non-Gaussian distributed measurement errors and missing values. Previous work [16–19] mainly focus on more simple error distributions and do not provide suitable countermeasures to increase the classification results in presence on arbitrary noise signals. For this, an optimization based error model (introduced in [20]) has been extended to approximate Laplacian distributed error signals with an additional randomized imputation of missing values. To minimize misclassifications resulting from PMU induced measurement errors, a two-stage training approach is introduced by incorporating denoising recurrent autoencoders to create robust feature representations for the underlying classification task. Multivariate PMU frequency and voltage signals are analysed with gated recurrent units (GRU) and an additional attention model to capture the necessary information to distinguish between different disturbance events. This enables an error-tolerant identification and localization of grid disturbances using simulated training data.

## 2. Extended optimization based error model

Based on preliminary work [20] an extended optimization based error model is used to create different error components of PMU measurements. The error model architecture is given in Fig. 2.

The basic variant of the optimization based error model estimates an erroneous PMU signal $\underline{x}_P$ from the (normalized) dynamic simulation signal $\underline{x}_S$ as follows

$$\underline{x}_P = \underline{x}_S + \underline{e} \text{ with } \underline{x}_P = \left[x_P^t\right]_{t=0}^{t=T}. \quad (1)$$

For this, Brent's optimization [21] is used to calculate the additive noise component $\underline{e}$ by adjusting a scaling factor $k$ to meet a given signal-to-noise-ratio (SNR) $r_{SNR}$. This is done for each vector $\underline{x}_S$ of the observation matrix $\boldsymbol{X}_S$ to build up the measurement matrix $\boldsymbol{X}_P$. More details about the optimization based error model can be found in [20]. In the literature, the noise component is typically sampled from a Gaussian distribution $\mathcal{N}$ with a mean $\mu = 0$ and a unit variance $\sigma = 1$. In this study, Laplacian distributed $\mathcal{L}$ error signals are considered as well with a mean $\mu = 0$ and a scale parameter $b$. With that, each entry $e^t$ of the noise signal $\underline{e}$ can be calculated as follows

$$e^t = k \cdot \varepsilon^t \text{ with } \varepsilon^t \sim \mathcal{N}(\mu, \sigma) \text{ or } \varepsilon^t \sim \mathcal{L}(\mu, b). \quad (2)$$

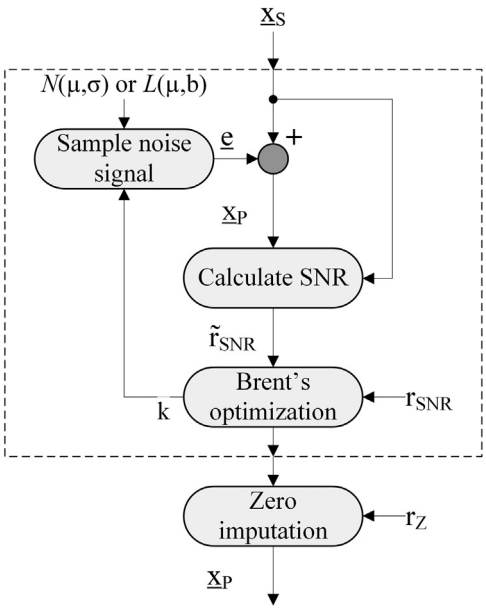


**Fig. 2.** Extended optimization based error model architecture.

Additionally, missing PMU measurement values are considered within that study and are modelled with a zero imputation procedure. With that, the missing values are imputed randomly in the matrix $\boldsymbol{X}_{\mathrm{P}}$ by specifying a ratio for zero values $r_{\mathrm{Z}} \in [0, 1]$. Since the optimization based error model is applied after the data normalization, the missing values appear as zero values at the model input side.

## 3. Disturbance classification with denoising recurrent autoencoders

This section describes the methodological details of the proposed error-tolerant disturbance classification approach. Starting from the definition of the general classification task in Section 3.1, the two-stage training approach incorporating recurrent denoising autoencoders is presented in Section 3.2. Section 3.3 gives additional information about the encoding of multivariate PMU frequency and voltage signals using GRUs and specialized attention mechanisms.

### 3.1. General problem setting

In general, the simultaneous identification and localization of grid disturbances, as introduced in [2], can be solved with regular classification approaches such as artificial neural networks. The input data $\boldsymbol{X}_{\mathrm{P}}$ (measured) or $\boldsymbol{X}_{\mathrm{S}}$ (simulated) consists of frequency and voltage magnitude signals over a fixed time period $T$ from $M$ PMU devices distributed over the grid

$$\boldsymbol{X}_{\mathrm{P,S}} = \left[\underline{x}^{t}_{\mathrm{P,S}}\right]_{t=0}^{t=\mathrm{T}} \text{ with } \boldsymbol{X}_{\mathrm{P}} \in \mathbb{R}^{2\times \mathrm{T}\times \mathrm{M}}. \tag{3}$$

In contrast to existing disturbance classification approaches, no full observability of the grid with PMUs is required, because the classification model directly predict the disturbance type $y_{\mathrm{Type}}$ and location $y_{\mathrm{Loc}}$ from the available PMU signals. The spatiotemporal correlation between the input signals (see also Fig. 6 in Section 4.1) allows a detection of disturbance events even if they are not directly observed by a PMU. Thus, the disturbance class $y$ is characterized as follows

$$y = \left[y_{\mathrm{Loc}}, y_{\mathrm{Type}}\right]. \tag{4}$$

These events are directly or indirectly observed by PMUs. All $K$ disturbance events $C$ are summarized with the class configuration $\overline{C}$

$$\overline{C} = [C_1 \cdots C_{\mathrm{K}}]. \tag{5}$$

### 3.2. Two-stage denoising autoencoder and disturbance classifier training

To increase the robustness of the disturbance classification based on simulated training data, a feature representation is required, which is invariant to PMU induced measurement errors. For this, a denoising recurrent autoencoder is trained on synthesized, erroneous PMU signals $\hat{\boldsymbol{X}}_{\mathrm{P}}$ to gain a robust feature representation. Afterwards, the learned encoder part is transferred to the classification model to apply a finetuning using the simulated PMU signals $\boldsymbol{X}_{\mathrm{S}}$. In the application phase, the disturbance classifier predicts the disturbance event $\hat{y}$ based on the measured PMU signals $\boldsymbol{X}_{\mathrm{P}}$. The general working principle is given in Fig. 3.

The *first stage* includes the training of a denoising recurrent autoencoder (dRAE). By the use of the optimization error model (see paragraph 2), approximated PMU measurements $\hat{\boldsymbol{X}}_{\mathrm{P}}$ are fed into the dRAE model to reconstruct the clean simulation data $\boldsymbol{X}_{\mathrm{S}}$. The dRAE consists of a GRU based encoder (E) and decoder (D) module, whose parameters $\underline{\theta}_{\mathrm{E}}$ and $\underline{\theta}_{\mathrm{D}}$ are learned together via backpropagation through time. Inside the GRU encoder, a hidden state vector $\underline{h}^{t}_{\mathrm{E}}$ is computed for each time step $t$ based on the normalized observation matrix $\hat{\boldsymbol{X}}^{\mathrm{N}}_{\mathrm{P}}$. These hidden state information are summarized with an attention-based embedding module (introduced in [1]) to create a compact feature representation $\underline{\mathrm{f}}$ as follows

$$\underline{\mathrm{f}} = f_{\mathrm{E}}\left(\hat{\boldsymbol{X}}^{\mathrm{N}}_{\mathrm{P}}, \underline{\theta}_{\mathrm{E}}\right) \text{ with } \underline{\mathrm{f}} \in \mathbb{R}^{\mathrm{R}}. \tag{6}$$

The GRU decoder takes this feature vector as input at each time step to compute a hidden state vector $\underline{h}^{t}_{\mathrm{D}}$. These vectors are passed to a final dense layer with linear activation function to estimate the normalized, clean PMU signals $\hat{\boldsymbol{X}}^{\mathrm{N}}_{\mathrm{S}}$ with

$$\hat{\boldsymbol{X}}^{\mathrm{N}}_{\mathrm{S}} = f_{\mathrm{D}}\left(\underline{\mathrm{f}}, \underline{\theta}_{\mathrm{D}}\right) \text{ with } \hat{\boldsymbol{X}}^{\mathrm{N}}_{\mathrm{S}} \in \mathbb{R}^{\mathrm{T}\times \mathrm{M}}. \tag{7}$$

The GRU encoder and decoder parameters $\underline{\theta}_{\mathrm{E}}$ and $\underline{\theta}_{\mathrm{D}}$ are learned by minimizing the mean absolute difference between the estimated and true simulation values as follows

$$\min_{\underline{\theta}_{\mathrm{E}}, \underline{\theta}_{\mathrm{D}}} \left(\left|\boldsymbol{X}^{\mathrm{N}}_{\mathrm{S}} - \hat{\boldsymbol{X}}^{\mathrm{N}}_{\mathrm{S}}\right|\right). \tag{8}$$

Further modifications of the GRU decoder are implemented to improve the reconstruction results. The feature vector is concatenated with the previous target observation $\underline{x}^{\mathrm{N},t-1}_{\mathrm{S}}$ to compute the hidden state vector $\underline{h}^{t}_{\mathrm{D}}$ within each GRU cell $f_{\mathrm{GRU}}$ with its trainable parameters $\underline{\theta}_{\mathrm{GRU}}$, such that

$$\underline{h}^{t}_{\mathrm{D}} = f_{\mathrm{GRU}}\left(\left[\underline{\mathrm{f}}, \underline{x}^{\mathrm{N},t-1}_{\mathrm{S}}\right], \underline{h}^{t-1}_{\mathrm{D}}, \underline{\theta}_{\mathrm{GRU}}\right). \tag{9}$$

This is principle is also known as "teacher forcing" or "conditioned decoder" [22] . Additionally, the hidden state vector of the GRU encoder at $t = 0$ is initialized with the last hidden state vector of the GRU encoder $\underline{h}^{t=0}_{\mathrm{D}} = \underline{h}^{\mathrm{T}}_{\mathrm{E}}$ and the output signals are reconstructed in reverse order.

In the *second stage* the learned GRU encoder module $f_{\mathrm{E}}$ is combined with a classifier module $f_{\mathrm{C}}$ to create the disturbance classification model based on the simulated signals $\boldsymbol{X}_{\mathrm{S}}$. With that, the encoder feature representation is first passed to a dense layer with a Sigmoid activation function (number of hidden units equals the number of classes) and afterwards passed to a dense

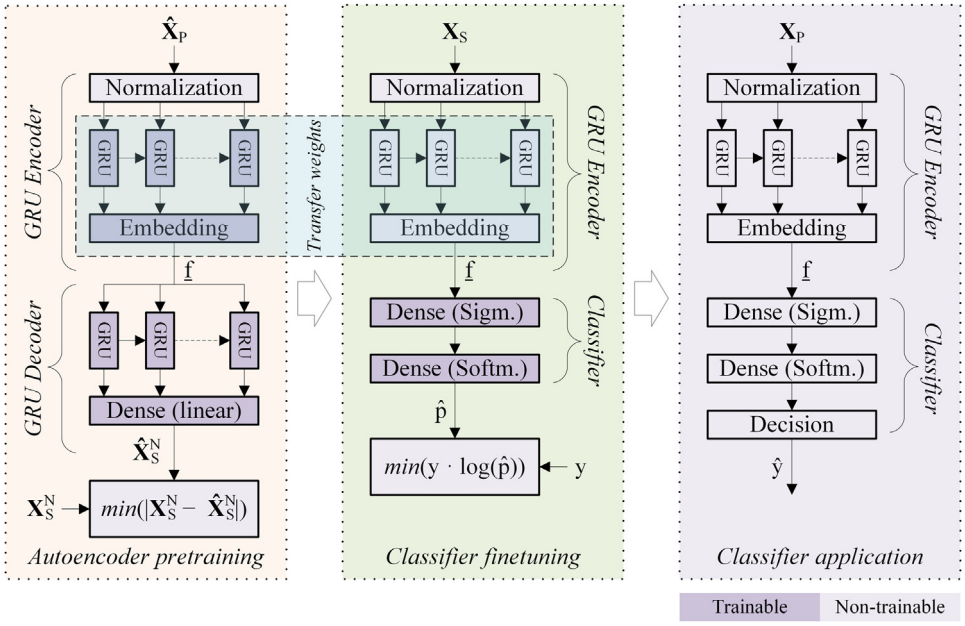


**Fig. 3.** Working principle of the two-stage disturbance classifier training and application incorporating denoising recurrent autoencoders.

layer with a Softmax activation function to estimate the posterior probabilities $\hat{\underline{p}}$ for all $K$ disturbance classes with

$$\hat{\underline{p}} = f_C\left(f_E\left(\boldsymbol{X}_S, \underline{\theta}_E\right), \underline{\theta}_C\right) \text{ with } \hat{\underline{p}} \in \mathbb{R}^K. \tag{10}$$

During the training of the classification model, the GRU encoder parameters $\underline{\theta}_E$ are kept fixed such that only the classifier parameters $\underline{\theta}_C$ are learned by minimizing the following cross-entropy loss formulation as follows

$$\min_{\underline{\theta}_C}\left(\sum_C y_C \log \hat{p}_C\right). \tag{11}$$

In the application phase, the trained classification model is faced with measured PMU signals $\boldsymbol{X}_P$, which can differ from the approximated signals $\hat{\boldsymbol{X}}_P$ with regard to the underlying error characteristics. Corresponding experiments are presented in Section 4.4.

### 3.3. Gated recurrent units based encoding of phasor measurements

The GRU encoder in the dRAE and classification model plays a vital role to compute informative and robust feature representations from multivariate PMU frequency and voltage signals. The architecture of the GRU encoder module used in this paper is given in Fig. 4.

At the input side, standardized frequency $x_F^t$ and voltage magnitude values $x_U^t$ from $M$ PMUs are passed to the GRU layer, such that the observation vector $\underline{x}^t$ can be described as

$$\underline{x}^t = \left[x_{F(1)}^t, x_{U(1)}^t, \ldots, x_{F(M)}^t, x_{U(M)}^t\right]. \tag{12}$$

Inside the GRU cells the hidden state vectors $\underline{h}^t$ are computed for each time step $t$ to capture the necessary information from the last recent observations and hidden states. As already introduced in [1], an attention model computes the feature vector as weighted sum of all hidden state vectors, such that

$$\underline{f} = \sum_t \alpha^t \cdot \underline{h}^t \text{ with } \underline{h}^t \in \mathbb{R}^R. \tag{13}$$

With that, the number of hidden units of the GRU layer $R$ corresponds to the dimension of the final feature representation.

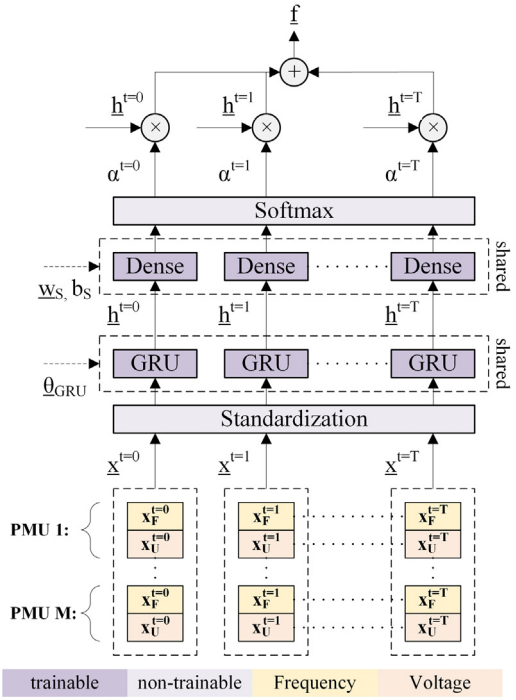


**Fig. 4.** Architecture of the GRU encoder with attention model for multivariate PMU signals.

The attention model computes the attention weights $\alpha^t$ by assigning a score value $s^t$ for each hidden state using a single-layer feedforward neural network with trainable weights $\underline{w}_S$ and bias $b_S$ as well as a tangent hyperbolic activation function. Afterwards, these score values are squashed into a range [0, 1] by applying a Softmax function, such that

$$\alpha^t = \frac{exp\left(s^t\right)}{\sum_t exp\left(s^t\right)} \text{ and } s^t = \tanh\left(\underline{w}_S \underline{h}^t + b_S\right). \tag{14}$$

With that, the neural network networks gains access to all information from the hidden states and can focus on specific time

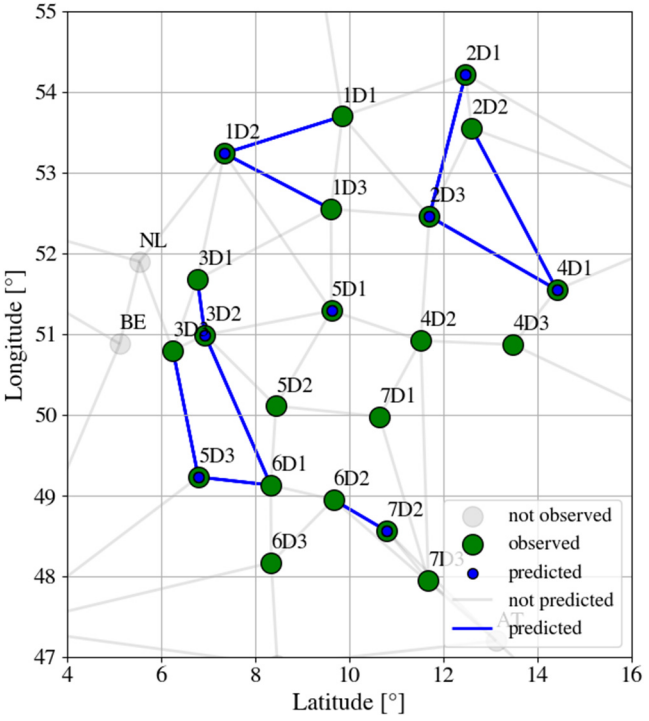


**Fig. 5.** Grid topology of the generic power transmission system.

ranges or time steps of the input signals, which improves both the reconstruction and classification performance.

## 4. Case studies and discussion

### 4.1. Grid topology and basic assumptions

The dynamic simulation model is a generic power transmission system, which consists of 33 substations and 172 transmission lines as well as various conventional and renewable energy resources (e.g. photovoltaic power plants). The system size is comparable to the IEEE 118 bus system. The grid is equipped with additional automatic voltage regulators (AVR) and power system stabilizers (PSS) to maintain a stable system operation. The DIgSILENT® Programming Language (DPL) is used to simulate disturbance events (e.g. power plant outages, line outages, short circuits) for different load and generation conditions. For more details see [1] .

The PMU frequency and voltage signals at the substation busbars are derived as average values from the RMS simulation results assuming a reporting rate of 25 f.p.s. or 40 ms. These PMU signals are extracted from the dynamic simulations until 10 s after the triggering of the disturbance event. For this study, a class configuration of 54 disturbance events is chosen including 8 power plant outages, 8 load changes, 8 PV power losses, 10 line trips and 20 three-phase short circuits at different line positions. These disturbance events are concentrated at or near to 8 stations of the electrical grid. The grid topology along with the predicted disturbance events and observed PMU stations is shown in Fig. 5.

These disturbance events are simulated for 7 operational points, which represent different load and generation conditions in the grid. 5 operational points are selected for the model training and validation as well as the 2 remaining operational points for the model testing. Some exemplary PMU frequency and voltage magnitude signals for a generator trip for 10 s post-disturbance time is given in Fig. 6.

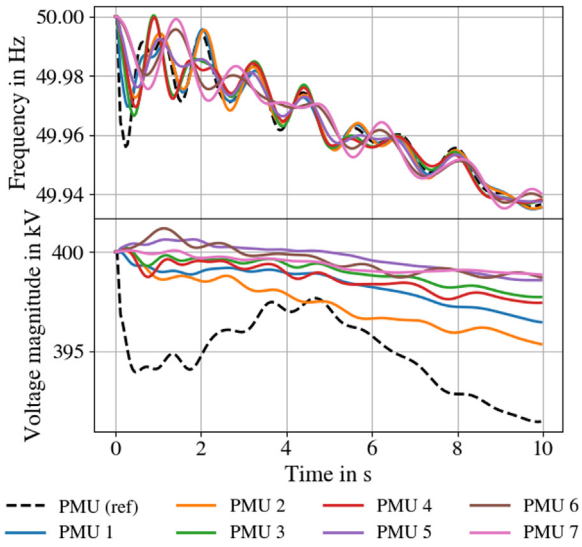


**Fig. 6.** Simulated frequency and voltage magnitude signals of a generator trip over 10 s for 8 PMUs (ref: PMU directly located at the disturbance).

**Table 2**
Optimal hyperparameters of the dRAEClass and ClassOnly models.

| Hyperparameter | dRAEClass | ClassOnly |
|---|---|---|
| Hidden units (Encoder) | 280 | 150 |
| Hidden units (Decoder) | 280 | – |
| Learning rate | 1st stage: 0.001<br>2nd stage: 0.0007 | 0.01 |
| Batch size | 1st stage: 10<br>2nd stage: 50 | 50 |

**Table 3**
Overview of the Gaussian noise scenarios.

| Noise scenario | $r_{SNR}$ | $r_Z$ |
|---|---|---|
| G1 | 10 dB | 4% |
| G2 | 25 dB | 2% |

### 4.2. Selection of hyperparameters

As a starting point, the optimal hyperparameters for the GRU encoder, decoder and classifier modules are identified assuming a Gaussian distributed error signal ($r_{SNR} = 10$ dB and $r_Z = 4\%$) in the test data. For this, different training runs are performed with early stopping and a validation set size of 25% compared to the training set. The results are computed for the combined dRAE and classifier model using the two-stage training approach as described in Section 3.2 (*dRAEClass*) and for a sole training of the classifier model (*ClassOnly*). During the experiments, significant differences can be observed when training both model variants. As an example Fig. 7 shows the effect of the learning rate on the reconstruction error of the dRAE model and on the accuracy of the classifier model in case of the two-stage training (*dRAEClass*).

Especially the learning rate and batch size have to be chosen carefully in both training stages to achieve optimal results. The best hyperparameter combinations for both model variations are given in Table 2.

Due to the additional reconstruction task, the number of trainable parameters of the *dRAEClass* combination is much higher than the *ClassOnly* model.

### 4.3. Gaussian distributed error signals

The first investigations focus on the model performance assuming Gaussian distributed error signals with different amounts

**Table 4**
Results for the Gaussian noise scenarios G1 and G2.

| Training data | Model | Accuracy (Training) | Accuracy (Validation) | Accuracy (Test) |
|---|---|---|---|---|
| *Experiment 1: Tested with noisy training data (G1)* | | | | |
| dRAE: Noisy (G1) Classifier: Clean | dRAEClass | 100.00 | 91.07 | 96.60 |
| Clean | ClassOnly | 96.63 | 84.99 | 92.65 |
| Noisy (G1) | ClassOnly | 97.27 | 81.92 | 93.43 |
| *Experiment 2: Tested with noisy test data (G1)* | | | | |
| dRAE: Noisy (G1) Classifier: Clean | dRAEClass | 100.00 | 91.23 | 82.44 |
| Clean | ClassOnly | 95.99 | 83.40 | 73.08 |
| *Experiment 3: Tested with noisy test data (G2)* | | | | |
| dRAE: Noisy (G1) Classifier: Clean | dRAEClass | 99.98 | 91.12 | 83.57 |
| Clean | ClassOnly | 96.19 | 83.40 | 74.83 |

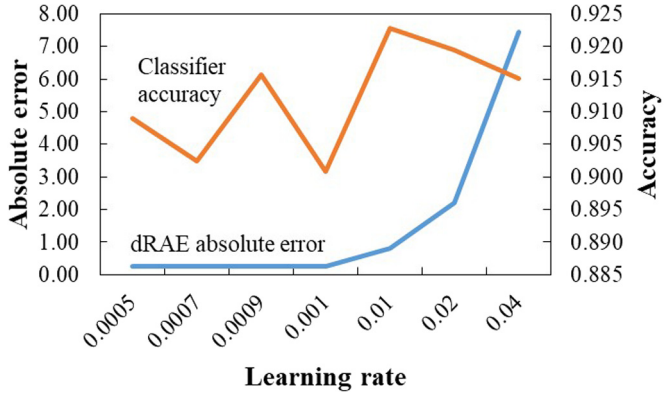


**Fig. 7.** Effects of the learning rate on the training performance of the dRAE and classifier models.

**Table 5**
Overview of the Laplacian noise scenarios.

| Noise scenario | $b$ | $r_{SNR}$ | $r_Z$ |
|---|---|---|---|
| L1 | 0.001 | 10 dB | 4% |
| L2 | 0.001 | 25 dB | 2% |

of noise ($r_{SNR}$) and missing values ($r_Z$) in the test data. The following two noise scenarios G1 (high amount of noise components) and G2 (low amount of noise components) are considered — see Table 3.

The investigations comprise of three experiments with different training and test datasets as well as noise scenarios. The corresponding accuracy results for the *dRAEClass* and *ClassOnly* model with the optimal hyperparameter configurations (see Table 2) is given in Table 4.

In experiment 1 the noise scenario G1 is considered for the training and application phase. Additionally, the models are tested on the noisy training data, such that $\hat{\boldsymbol{X}}_P = \boldsymbol{X}_P$. The *dRAEClass* model achieves the highest test accuracy with 96.60% compared to the *ClassOnly* model with 92.65%. Even if the *ClassOnly* model is trained with the noisy training data, the test accuracy is still lower with 93.43% compared to the *dRAEClass* model. As one major reason, the *dRAEClass* model provides a higher recognition and generalization capability, which improves the training and validation accuracies in all experiments. In experiment 2 the models are tested with the noisy test data, such that $\hat{\boldsymbol{X}}_P \neq \boldsymbol{X}_P$. The error distributions between training and test data are still the same. In that case, the differences between the model variants increase with a test accuracy for the *dRAEClass* model of 82.44% compared to a test accuracy for the *ClassOnly* model of 73.08%. In experiment 3 the error distribution of the test data corresponds to G2 whereas the error distribution of the training data is kept unchanged. Here too, the *dRAEClass* model provides the highest test accuracies with 83.57% compared to the *ClassOnly* model with 74.83%.

### 4.4. Laplacian distributed error signals

For the investigation of Laplacian distributed measurement errors, the noise scenarios L1 (high amount of noise components) and L2 (low amount of noise components) are defined with a fixed scale parameter $b$ and different values for $r_{SNR}$ and $r_Z$ – see Table 5.

Four experiments are performed with different training and test datasets as well as noise scenarios. The corresponding accuracy results for the *dRAEClass* and *ClassOnly* model with the optimal hyperparameter configurations (see Table 2) is given in Table 6.

Similar to experiment 1 from Section 4.3, experiment 4 considers an equal noise scenario (L1) for the training and test data and both models are tested using the noisy training data. As to be expected, the *dRAEClass* model shows the highest test accuracy with 96.28% compared to the *ClassOnly* model with 92.47%. In experiment 5, the models are tested with the noisy test data assuming the L1 noise distribution. If the *dRAEClass* model is trained assuming L1 distributed measurement errors, the test accuracy with 81.89% is higher compared to the *ClassOnly* model with 75.60%. Even if *dRAEClass* model is trained assuming G1 distributed measurement errors, the test accuracy does not change significantly with 81.57%. Similar findings can be observed from the results of experiment 6 (both models are tested with L2 distributed errors) and 7 (both models are tested with G1 distributed errors). As a consequence, even if the noise distributions between training and test data are different from each other, the *dRAEClass* model can still achieve significantly better test accuracies compared to the *ClassOnly* model. Also in case of Laplacian distributed errors, the training and validation accuracies of the *dRAEClass* model are much higher compared to the *ClassOnly* model. This has to be considered, when evaluating the benefit of the two-stage training approach.

## 5. Conclusions and outlook

From the results in Sections 4.3 and 4.4, the two-stage training approach incorporating denoising recurrent autoencoders (*dRAEClass* model) shows superior classification accuracies compared

**Table 6**
Results for the Laplacian noise scenarios L1 and L2.

| Training data | Model | Accuracy (Training) | Accuracy (Validation) | Accuracy (Test) |
|---|---|---|---|---|
| *Experiment 4: Tested with noisy training data (L1)* | | | | |
| dRAE: Noisy (L1) Classifier: Clean | dRAEClass | 99.98 | 89.59 | 96.28 |
| Clean | ClassOnly | 96.21 | 85.37 | 92.47 |
| *Experiment 5: Tested with noisy test data (L1)* | | | | |
| dRAE: Noisy (L1) Classifier: Clean | dRAEClass | 99.98 | 91.89 | 81.89 |
| dRAE: Noisy (G1) Classifier: Clean | dRAEClass | 100.00 | 90.25 | 81.57 |
| Clean | ClassOnly | 97.49 | 84.60 | 75.60 |
| *Experiment 6: Tested with noisy test data (L2)* | | | | |
| dRAE: Noisy (L1) Classifier: Clean | dRAEClass | 99.95 | 89.42 | 82.46 |
| Clean | ClassOnly | 97.20 | 85.70 | 74.84 |
| *Experiment 7: Tested with noisy test data (G1)* | | | | |
| dRAE: Noisy (L1) Classifier: Clean | dRAEClass | 99.87 | 91.62 | 81.34 |
| Clean | ClassOnly | 95.99 | 83.40 | 73.08 |

**Table 7**
Accuracies of the proposed classification approach against a benchmark model.

| Model | Training | Validation |
|---|---|---|
| dRAEClass | 96.63–100.00 | 84.99–91.89 |
| ClassOnly | 95.99–97.49 | 81.92–85.70 |
| Benchmark | 88.01 | 68.40 |

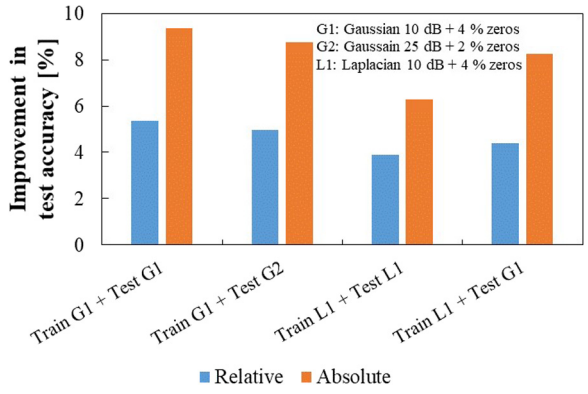


**Fig. 8.** Absolute and relative improvements of the test accuracies for different noise scenarios.

to the one-stage training (*ClassOnly* model) for different noise scenarios. The improved training and validation accuracies of the *dRAEClass* model suggest a better recognition and generalization capability as well and has to be considered when comparing the test accuracies. To sum up the results, Fig. 8 shows the absolute improvements of the test accuracies between both models $\eta_{\mathrm{ACC,te}}^{\mathrm{dRAEClass}} - \eta_{\mathrm{ACC,te}}^{\mathrm{ClassOnly}}$ as well as the relative improvements of the test accuracies with additional correction of the different training accuracies $\left(\eta_{\mathrm{ACC,te}}^{\mathrm{dRAEClass}} - \eta_{\mathrm{ACC,te}}^{\mathrm{ClassOnly}}\right) - \left(\eta_{\mathrm{ACC,tr}}^{\mathrm{dRAEClass}} - \eta_{\mathrm{ACC,tr}}^{\mathrm{ClassOnly}}\right)$.

As it can be seen, the absolute improvement of the test accuracies of the *dRAEClass* model is roughly between 6% and 9% compared to the *ClassOnly* model whereas the relative improvement reduced to a range between 4% and 5%. It should be highlighted, that the two-stage training approach even works, if the error distribution of the approximated noisy training data $\hat{\boldsymbol{X}}_{\mathrm{P}}$ does not match the true error distribution of the test data $\boldsymbol{X}_{\mathrm{P}}$. For a better ranking of the *dRAEClass* and *ClassOnly* model results, Table 7 compares the training and validation accuracies from Sections 4.3 and 4.4 with the classification accuracies of the benchmark approach from [3] . For this, a full PMU observability of the grid is assumed as well as noise-free training and validation datasets.

It can be seen, that among all case studies the recurrent neural network based *dRAEClass* and *ClassOnly* models can surpass the benchmark model results.

Still, further evaluation studies are required to justify the significance of the proposed model approach including the integration of larger power transmission systems, additional benchmark solutions and real PMU measurement datasets. Additional experiments should also focus on the generalization capability of the *dRAEClass* model to denoise arbitrary measurement errors even in case of large deviations between the training and test data (e.g. using techniques from domain adaptation). Also, an efficient methodology should be developed for extracting noise parameters from real PMU measurements to adapt the optimization based error model and to refine the classification model during the application phase.

## CRediT authorship contribution statement

**André Kummerow:** Conceptualization, Investigation, Methodology, Software, Writing – original draft. **Mohammad Dirbas:** Investigation, Methodology, Software, Writing – original draft. **Cristian Monsalve:** Data curation, Resources. **Steffen Nicolai:** Supervision, Validation, Writing – review & editing, Funding acquisition. **Peter Bretschneider:** Project administration, Supervision, Validation, Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

This publication is a partly result of the project HyLITE (0350034) funded by the German Federal Ministry for Economic Affairs and Energy (BMWi).

## References


[1] A. Kummerow, C. Monsalve, C. Brosinsky, S. Nicolai, D. Westermann, A novel framework for synchrophasor based online recognition and efficient post-mortem analysis of disturbances in power systems, Appl. Sci. 10 (15) (2020) http://dx.doi.org/10.3390/app10155209.

[2] A. Kummerow, C. Monsalve, S. Nicolai, P. Bretschneider, Simultaneous online identification and localization of disturbances in power transmission systems, in: 2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe), Bucharest, Romania, Sep. 2019 - Oct. 2019, 2019, pp. 1–5.

[3] M. Biswal, S.M. Brahma, H. Cao, Supervisory protection and automated event diagnosis using PMU data, IEEE Trans. Power Deliv. 31 (4) (2016) 1855–1863, http://dx.doi.org/10.1109/TPWRD.2016.2520958.

[4] A.K. Singh, M. Fozdar, Supervisory framework for event detection and classification using wavelet transform, in institute of electrical and electronics engineers, power & energy society, etal, in: IEEE PES General Meeting, 2017, pp. 1–5.

[5] H. Jiang, J.J. Zhang, W. Gao, Z. Wu, Fault detection, identification, and location in smart grid based on data-driven computational methods, IEEE Trans. Smart Grid 5 (6) (2014) 2947–2956, http://dx.doi.org/10.1109/TSG.2014.2330624.

[6] C. Brosinsky, A. Kummerow, A. Naumann, A. Kronig, S. Balischewski, D. Westermann, A new development platform for the next generation of power system control center functionalities for hybrid AC-HVDC transmission systems, in: 2017 IEEE Power & Energy Society General Meeting: 16-20, Chicago, IL, 2017, 2017, pp. 1–5.

[7] A. Kummerow, C. Brosinsky, C. Monsalve, S. Nicolai, P. Bretschneider, D. Westermann, Pmu-based online and offline applications for modern power system control centers in hybrid AC-HVDC transmission systems, in: Proceedings of International ETG Congress 2019: 08. – 09.05.2019 in Esslingen Am Neckar, VDE VERLAG GMBH Berlin Offenbach, 2019, pp. 405–410.

[8] M. Panteli, D.S. Kirschen, Situation awareness in power systems: Theory, challenges and applications, Electr. Power Syst. Res. 122 (2015) 140–151, http://dx.doi.org/10.1016/j.epsr.2015.01.008.

[9] C. Hannon, D. Deka, D. Jin, M. Vuffray, A.Y. Lokhov, Real-time anomaly detection and classification in streaming PMU data, 2019, [Online]. Available: https://arxiv.org/pdf/1911.06316.

[10] Y. Zhu, C. Liu, K. Sun, Image embedding of PMU data for deep learning towards transient disturbance classification, in: 2018 IEEE International Conference on Energy Internet, ICEI, Beijing, May. 2018 - May, 2018, pp. 169–174.

[11] IEEE/IEC International Standard - Measuring relays and protection equipment - Part 118-1, Synchrophasor for power systems - Measurements, Piscataway, NJ, USA.

[12] L. Shen, et al., A flexible ensemble algorithm for big data cleaning of PMUs, Front. Energy Res. 9 (2021) http://dx.doi.org/10.3389/fenrg.2021.695057.

[13] I. Idehen, W. Jang, T. Overbye, Pmu data feature considerations for realistic, synthetic data generation, in: 2019 North American Power Symposium, NAPS, Wichita, KS, USA, Oct, 2019, pp. 1–6.

[14] K.S. Sajan, et al., Realistic synchrophasor data generation for anomaly detection and event classification, in: 8th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES): Proceedings : Virtual Workshop, Online from April 21, 2020, Sydney, Australia, 2020, pp. 1–6.

[15] S. Wang, J. Zhao, Z. Huang, R. Diao, Assessing Gaussian assumption of PMU measurement error using field data, IEEE Trans. Power Delivery 33 (6) (2018) 3233–3236, http://dx.doi.org/10.1109/TPWRD.2017.2762927.

[16] Z. Li, H. Liu, J. Zhao, T. Bi, Q. Yang, A power system disturbance classification method robust to PMU data quality issues, IEEE Trans. Ind. Inf. (2021) 1, http://dx.doi.org/10.1109/TII.2021.3072397.

[17] Y. Chen, L. Xie, P.R. Kumar, Power system event classification via dimensionality reduction of synchrophasor data, in: 2014 IEEE 8th Sensor Array and Multichannel Signal Processing Workshop, SAM, a Coruna, Spain, 57–60.

[18] M. Biswal, Y. Hao, P. Chen, S. Brahma, H. Cao, P. de Leon, Signal features for classification of power system disturbances using PMU data, in: 19th Power Systems Computation Conference: PSCC 2016 Genova : June (2016) 20-24, Centro Congressi Porto Antico, Magazzini Del Cotone Conference Centre, Genova, Italy, Genoa, Italy, 2016, pp. 1–7.

[19] Le Xie, Y. Chen, P.R. Kumar, Dimensionality reduction of synchrophasor data for early event detection: Linearized analysis, IEEE Trans. Power Syst. 29 (6) (2014) 2784–2794, http://dx.doi.org/10.1109/TPWRS.2014.2316476.

[20] A. Kummerow, M. Dirbas, C. Monsalve, S. Nicolai, P. Bretschneider, Influence of autoregressive noise on phasor data based disturbance classification, in: 2021 International Conference on Smart Energy Systems and Technologies, SEST, Vaasa, Finland, Sep. 2021, 2021, pp. 1–6.

[21] R.P. Brent, Algorithms for Minimization Without Derivatives, Prentice-Hall, Englewood Cliffs, N.J., 1973.

[22] Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016.