



Controlled generation of unseen faults for *Partial* and *Open-Partial* domain adaptation

Katharina Rombach^a, Gabriel Michau^b, Olga Fink^{c,*}

^a Intelligent Maintenance Systems, ETH Zurich, Stefano-Franscini-Platz 5, 8093 Zurich, Switzerland

^b Maintenance Systems & Technologies, Stadler Service AG, Ernst-Stadler-Strasse 4, 9565 Bussnang, Switzerland

^c Intelligent Maintenance and Operations Systems, EPFL, Station 18, 1015 Lausanne, Switzerland

ARTICLE INFO

Keywords:

Unseen fault generation

Controlled generation

Partial domain adaptation

Open-Partial domain adaptation

ABSTRACT

New operating conditions can result in a significant performance drop of fault diagnostics models due to the domain shift between the training and the testing data distributions. While several domain adaptation approaches have been proposed to overcome such domain shifts, their application is limited if the fault classes represented in the two domains are not the same. To enable a better transferability between two different domains, particularly in setups where only the healthy data class is shared between the two domains, we propose a new framework for *Partial* and *Open-Partial* domain adaptation based on generating distinct fault signatures with a Wasserstein GAN. The main contribution of the proposed framework is the controlled data generation with two characteristics. Firstly, previously unobserved target faults can be generated by having only access to healthy target and faulty source samples. Secondly, distinct fault types and severity levels can be generated precisely. The proposed method is especially suited for extreme domain adaption settings that are particularly relevant in the context of complex and safety-critical systems, where only one class is shared between the two domains. We evaluate the proposed framework on *Partial* as well as *Open-Partial* domain adaptation tasks on two bearing fault diagnostics case studies. In the evaluated case studies the proposed methodology demonstrated superior results compared to other methods, particularly in the presence of large domain gaps. The experiments conducted in different label space settings (*Partial* and *Open-Partial*) showcase the versatility of the proposed framework.

1. Introduction

A reliable operation of complex (safety-critical) assets can be achieved by monitoring the condition of the assets in real time, detecting the faults in an early stage and distinguishing between the different fault types and fault severity levels to enable an informed schedule of the recovery maintenance or fault mitigation actions. Data-driven models based on real-time condition monitoring (CM) data have shown a great potential for fault detection and diagnostics [1]. However, CM data is often affected by distributional shifts (referred to as domain shifts), that can significantly decrease the performance of data-driven models. For example, changing operating conditions (OCs) can cause such a distributional shift [2] or CM data of two units of a fleet can differ significantly due to differences in their configurations and operating regimes [3]. To enable the transfer of data-driven models under domain shifts, domain adaptation (DA) methods have been successfully applied in fault diagnostics [4,5]. Most of the proposed approaches, however, require a *ClosedSet* DA setting, where the same

fault classes are represented in both domains (see Fig. 1). However, in real-world datasets, the classes represented in the two domains are often not congruent. Due to the rareness of faults in safety-critical industrial assets, observing each possible fault in all assets of a fleet and/or under all possible OCs is often infeasible. This results in label space discrepancies in CM datasets of different domains.

In the literature, different DA settings have been distinguished by their type of discrepancy in the label space [6–8]. This is illustrated in Fig. 1. In the *Partial* DA setting, the target domain covers only a subset of the source classes (source domain has private classes), whereas in the *OpenSet* DA setting, the source domain covers only a subset of the target domain classes (target domain has private classes). The *Open-Partial* DA setup is a combination of both previous settings where both domains have private classes that are not represented in the other domain. Most existing DA methods are designed for only one of the above mentioned DA settings (mainly *ClosedSet* DA) and are often not transferable to other DA scenarios. This poses a challenge for successful

* Corresponding author.

E-mail address: olga.fink@epfl.ch (O. Fink).

<https://doi.org/10.1016/j.ress.2022.108857>

Received 26 March 2022; Received in revised form 20 September 2022; Accepted 24 September 2022

Available online 10 October 2022

0951-8320/© 2022 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

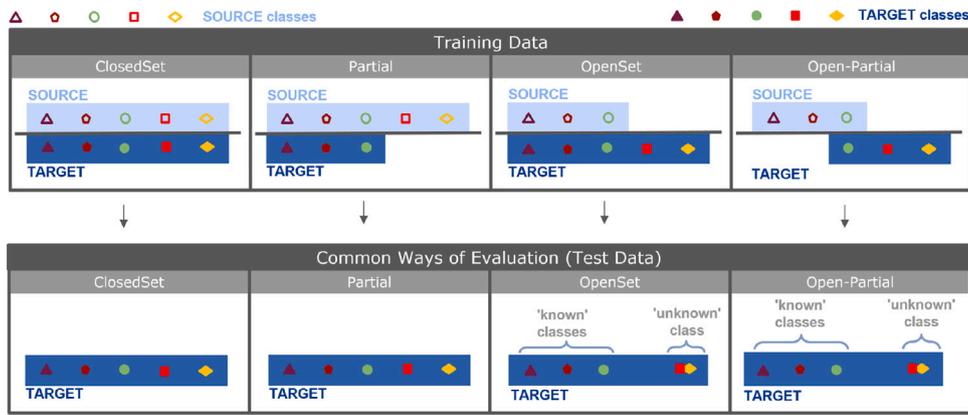


Fig. 1. Four DA configurations according to label space discrepancies: (a) *ClosedSet*; (b) *Partial*, (c) *OpenSet*, (d) both *Open-Partial*.

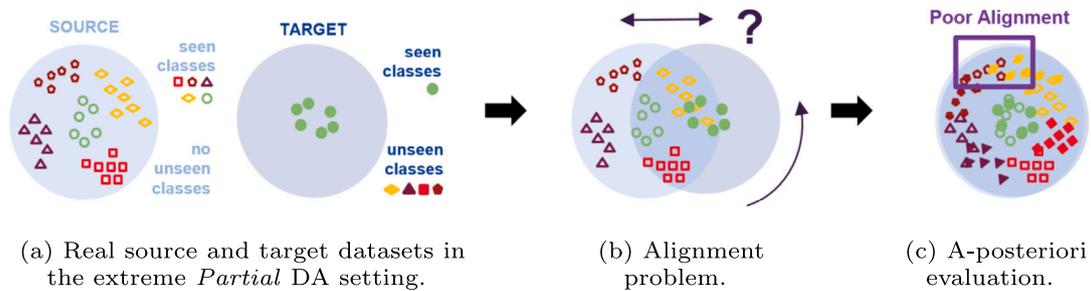


Fig. 2. Illustration of the source and target alignment challenge when only one class is shared between the domains on the example of the *Partial* DA setting: The source and the target datasets are shown in Fig. 2(a) whereby only one class (green class) is represented in the target domain. The alignment step based on one class only is shown in Fig. 2(b), whereby the challenge of finding the optimal alignment is indicated. The quality of chosen alignment method can only be tested during the a-posteriori evaluation, when the target classes have been observed (see Fig. 2(c)).

DA in real applications, where different types of discrepancies in the label space are common. Moreover, since faults occur rarely in either of the domains [9], data from specific faults types are typically only represented in one of the domains. This presents a DA task with an extreme case of label space discrepancy, where only one class, the healthy one, is shared between the two domains [10]. For example, if a system starts operating under a new OC, only data of the assets' current condition will be available. For safety-critical systems, this is usually the healthy condition, meaning that only the healthy class is shared between datasets from various OCs (an extreme case of *Partial* DA). As illustrated in Fig. 2, such an extreme case of label space discrepancy between two domains can pose a significant challenge for *Partial* DA methods based on feature alignment. With only one class shared between the two domains, there exist many possible alignment solutions (see Fig. 2(b)) and their performance can only be evaluated after the model is employed and the real target faults have been observed (see Fig. 2(c)). Extreme discrepancies in the label space of training datasets can also arise if two units of a fleet are experiencing different fault types during the data collection (and model development) period. Then, in the available training dataset, the only common health class experienced so far by both units may be the healthy class (an extreme case of *Open-Partial* DA). Previous studies have shown that, generally, the less classes are shared between the domains, the harder the DA task becomes [10,11]. Despite the relevance to PHM applications, there is hardly any work tackling the extreme cases of discrepancies in the label spaces (with only one shared class between two domains) for fault diagnostics in different DA settings. These extreme scenarios are in the focus of the research in this paper.

We propose to address the challenge raised by label space discrepancies for DA by enabling the generation of domain- and class-specific data from fault conditions that have not been observed before in the target domain. The generated fault data can compensate for unseen

domain-specific fault classes and, thereby, transform the given *Partial* or *Open-Partial* DA setting into a *ClosedSet* DA setting if it fulfills two requirements. Firstly, the generated data should be adapted to the specificities of the desired domain and secondly, the faults should be specific to a desired class in the label space. This is, on the one hand, a particularly challenging task since no fault data in the target domain is available i.e. the data generation is unsupervised. On the other hand, it is a particularly relevant case for real safety-critical applications where a representative dataset of fault data is typically not available. To enable unsupervised data generation, the proposed work is based on the hypothesis that the Fourier spectrum from faulty data can be disentangled in data signatures that represent (1) solely fault class characteristics and (2) domain-specific characteristics within the data. Building on this hypothesis, the main contribution of this research is a novel framework *FaultSignatureGAN* based on a Wasserstein GAN [12] that enables to generate domain-independent fault class signatures that are transferable to any new domain, given only healthy data of that domain. *FaultSignatureGAN* enables a controlled way to generate physically plausible faults of previously unobserved distinct classes in the target domain and thereby, enables to complement label spaces with different types of class discrepancies for DA tasks. Since the proposed framework relies solely on the availability of source faults and healthy target data, its benefits are particularly pronounced for targeting the extreme case of DA that is particularly relevant in the context of safety-critical systems, where only one class (the healthy class) is shared between the two domains. However, it is applicable to any number of shared and missing fault classes in the two domains. The proposed framework *FaultSignatureGAN* is not limited to only one type of label space discrepancy since it is applicable in *Partial* as well as *Open-Partial* setups.

The remainder of the paper is organized as follows. First, relevant related work is summarized in Section 2. The proposed framework is

explained in Section 3. The case studies are introduced in Section 4 and the exact setup of the conducted experiments is stated in Section 5. The results of the conducted experiments for *Partial* DA are shown in Section 6.1 and for *Open-Partial* settings in Section 6.2. The findings are discussed in Section 7 and conclusions are drawn in Section 8.

2. Related work

Domain Adaptation has been intensively studied in recent years in the context of PHM applications [13]. Most of the proposed approaches, however, have been exclusively developed for the *ClosedSet* DA setting where the source and target domain cover the same classes — see left column in Fig. 1. As exemplified in Section 1, the assumption of a *ClosedSet* setting is not realistic in many practical applications. Hence, *ClosedSet* DA methods do not meet the requirements of industrial applications. There has been an increasing interest to develop methods that address more realistic DA scenarios with label space discrepancies. Approaches for *Partial* [6], *OpenSet* [7] as well as *Open-Partial* [8] DA have been mainly developed in the field of computer vision. Recently, several research studies have developed the ideas further to adapt them to the challenges of real CM data. In the context of fault diagnostics, adversarial approaches have been proposed for *OpenSet* DA with different degrees of label space discrepancies i.e. with a different number of shared classes between the two domains. For example, in [14], an auxiliary domain discriminator was introduced to attribute less weight to private target samples and fault diagnostics experiments were conducted on three bearing datasets with two shared classes between the two domains. In another study on *OpenSet* DA for fault diagnostics, Zhang et al. [11] used an instance-level weighted mechanism to identify private target classes and tested the proposed method i.a. with three (out of ten) shared classes between the domains on two rotating machinery datasets. The results demonstrated that, generally, the less classes are shared between the domains, the harder the DA task becomes. Another method proposed a source class-wise and target instance-wise weighting mechanism combined with an additional outlier identifier for *OpenSet* fault diagnostics on two rotating machine datasets. The proposed method has even been applied on multiple label space discrepancy settings [15]. Despite its relevance to fault diagnostics in safety-critical complex technological system, none of the above mentioned studies has tackled the extreme case of the *OpenSet* or *Open-Partial* DA setting, where only one class (the healthy class) is shared between the domains. Another major limitations of the previously proposed approaches on *OpenSet* DA is that they only aim to classify known classes (i.e. source classes) and do not enable to distinguish private target samples into different classes. In safety-critical applications, however, it is important to distinguish between different health conditions within the private target samples to plan appropriate maintenance actions.

Methods targeting the *Partial* DA setting have also been developed for PHM applications. For example, a class-weighted adversarial DA method was proposed that uses the domain discriminator's output to detect private classes [16]. The output of two classifiers has also been employed to estimate the target distribution and train domain-invariant representations [17]. Also, randomly selected source data is used to augment the target domain to align the conditional distributions combined with a class-wise adaptation [18]. Some research studies have even dealt with the extreme case of *Partial* DA where only the healthy state is shared between the two domains [10,19,20]. For example, Li and Zhang [19] proposed a conditional data alignment step (using the maximum mean discrepancy) that is only applied to the healthy data from the source and target domain to prevent misalignment due to the label space discrepancy. In addition to the conditional alignment, the authors proposed prediction consistency schemes using multiple classifier models for fault diagnostics in *Partial* DA settings. Wang et al. [10] proposed a unilateral alignment approach (*Unilateral*) for *Partial* DA with extreme label space discrepancy. The proposed method made

use of the inter-class relationships of the source domain and aligned the target features to the pre-trained source domain features. Although the results of previous studies using different feature source and target alignment techniques in extreme *Partial* DA settings are promising, the methods have mainly been tested on CM datasets with small domain gaps (indicated by the high Baseline classification performance). The employed methods may fail under large domain shifts, where the inter-class relationships might have changed significantly.

One fault data generation approach was investigated in the extreme case of *Partial* DA combined with an additional alignment step [20]. However, the proposed target data generation method required extrapolation abilities of the generative model. Given the limited extrapolation abilities of deep models, it is not to be expected that the generated data resembles realistic target faults — especially given large domain gaps. Instead of generating target data as performed in [20], Zhao et al. [18] adapted the idea of Liang et al. [6] and proposed to augment the target data with source data to compensate the missing class data and performed adversarial feature alignment on the augmented and class-weighted datasets combined with a class-center-alignment loss. While the source data augmentation stabilized the alignment process, the proposed method may fail in settings where the inter-class relationships might have changed significantly. Further, the above mentioned approaches tackling different settings of label space discrepancy in DA have usually been developed for one specific DA setting, either *Partial* or *OpenSet*, and are typically not applicable in other settings. Furthermore, large domain gaps have not been tackled so far in the extreme case of label space discrepancy where only the healthy class is shared between the domains. Another limiting factor in applying the above mentioned DA methods based on feature alignment to new safety-critical assets is finding an optimal hyperparameter setting. With only one class being shared between the domains, there exist multiple possible alignment solutions and their quality can only be evaluated a posteriori, posing a safety risk in industrial assets. Therefore, previous works used, e.g., data and labels from target faults for one validation domain shift to tune the hyperparameters [10]. This solution to find the optimal hyperparameter settings is, however, not possible in real applications where data from unobserved target faults is not available.

In this work, we aim to develop a framework that performs well in the extreme case of DA under different label space discrepancy settings with a particular focus on the *Partial* and *Open-Partial* setting. We aim to develop a framework that enables DA also in the cases where the domain gaps are large. Further, we aim to achieve this without relying on target validation data to tune our methodology, as this is one of the limiting factors in existing DA methods to new safety-critical assets.

Domain generalization addresses the challenge of fault diagnostics under unforeseen domain shifts (contrary to one explicit shift between two domains). Different techniques have been proposed in the context of fault diagnostics [21,22]. However, these methods generally require access to multiple source domains, mainly with shared labeled spaces. This is often not given in industrial applications and therefore, domain generalization approaches are not applicable to the challenges addressed in this research.

Controlled Synthetic Data Generation has raised a lot of attention in recent years [23]. In the context of DA for computer vision tasks, for example, conditional generative models have been employed for domain mapping i.e. to translate a source input image to an image that closely resembles the target distribution [24]. However, these approaches require a *ClosedSet* DA setting since the target domain typically inherits the labels from the source domain. In the context of PHM applications, generative models have mainly been applied to balance imbalanced datasets, whereby e.g. conditional GANs have been used to control the generation process to generate desired distinct classes [25]. However, those approaches are solely suited to generate data from classes that have been observed before and not to generate previously unobserved classes in a specific domain. The latter is the focus of our research.

Contrary to using generative models, Wang et al. [26] proposed to use expert knowledge about different fault type patterns to generate synthetic fault data without access to any real fault data. This enabled to address fault diagnostics if only little real target fault data is available by performing DA in a subsequent step to close the synthetic-to-real domain gap. The expert knowledge enables to transfer different fault types to different types of bearings. However, this approach requires a substantial domain knowledge. Furthermore, patterns of different fault types (as in [26]) are typically easier to distinguish compared to different severity levels of the same fault type, as addressed in this research. To distinguish between different fault severities as well as types, synthetic data representing also different fault severities is required.

The concept of **disentanglement**, which is based on the hypothesis that real-world data is generated by a few independent explanatory factors of variation [27], has also been studied in the context of controlled data generation. However, although disentangled representations should be general and are expected to be generalizable to new domains, recent studies found that disentanglement does not guarantee combinatorial generalization (understand and produce novel combinations of familiar elements) [28]. To mitigate the lack of generalizability of disentangled representations, it is possible to constrain the disentanglement using a-priori knowledge on the data structure. For example, Yang and Soatto [29] assumed that, in image datasets, the domain-specific information is solely represented in the low frequency range whereas the semantic information is reflected only in the high frequency range. This assumption allowed the authors to generate unseen target data simply by swapping the domain-specific low-frequency block of the source and target images and perform DA with the synthetically generated data. This block-wise distinction into a domain-specific and a semantic-specific frequency ranges can be considered as a disentanglement in the Fourier space. Unfortunately, such a block-wise distinction of fixed frequency ranges representing either solely the domain-specific or semantic-specific components is not possible for CM data from mechanical systems with complex dynamic behavior. Instead, we expect a fault as well as a domain shift to affect the entire frequency spectrum. Based on the intuition that the OCs are independent of defects, we can assume that faults create disturbances on top of existing signals. We, therefore, assume that the Fourier spectrum can be expressed as the sum of domain-specific components and fault-specific components. This assumption, that both information content (domain-specific and semantic-specific) does not only affect constrained frequency ranges but rather impact the entire spectrum, generalizes the work of [29] to data from other application domains, in particular to CM data from complex industrial systems. Further, the assumptions enables the generation of unseen data while neither relying on combinatorial generalization of disentangled features nor relying on extrapolation abilities of the generative model.

In this work, we aim to develop a framework that enables a controlled generation of novel distinct fault classes in a target domain where the fault condition has not been observed before. Thereby, the **contribution of our proposed framework** is the generation of unseen domain-specific fault data, that enables DA with extreme label space discrepancies, also under large domain gaps. Contrary to other generative approaches, we do not only control the class being generated but also the specific domain of the data. Further, the data generation is unsupervised since the respective target fault has not been observed before in the specific target domain i.e. we enable the controlled generation of out-of-distribution data. Although the developed framework enables the generation of previously unseen data, it does not rely on extrapolation abilities of the generative model but instead, it relies on a disentanglement assumption. This assumption enables to transfer the fault information between different domains and, ultimately, to generate physically plausible data of unseen fault types and fault severities. The proposed framework, therefore, enables to generate data that can substitute for missing domain-specific class data for DA problems

with label space discrepancies. Our methodology is especially suited for DA in the extreme case of label space discrepancy, where only one class is shared between the domains and thereby, addresses an important requirement of reliable fault diagnostics in complex industrial (safety-critical) assets. However, it is also applicable to DA setups with any number of missing classes. Furthermore, contrary to other DA approaches, the proposed methodology is universally applicable to both *Partial*, *Open-Partial* DA setups.

3. Methodology

We propose a framework, referred to as *FaultSignatureGAN*, that enables to generate distinct domain-independent fault signatures based on the hypothesis that the faulty signal can be represented as the sum of domain-specific components and fault-specific components. These fault signatures can be transferred to new target domains such that the transferred data is representative of distinct fault classes in a target domain where they have not been observed before. The generated data is then used in a subsequent step to substitute for missing class data in different DA settings with label space discrepancies: *Partial* (see Fig. 3) and *Open-Partial* DA (see Fig. 4). Finally, a classification model is trained on the augmented datasets.

3.1. Training *FaultSignatureGAN*

FaultSignatureGAN comprises three parts (A–C) as illustrated in Fig. 5: (A) The first part ensures that generated fault signatures are easily transferable to a specific domain; (B) the second part ensures that the transformed fault signatures represent plausible domain data; and (C) the last part ensures that the transformed fault signatures are representative of the desired fault classes. Part (A) of the framework is tackled by a generative network that generates domain-independent fault signatures from distinct classes in the Fourier domain. These fault signatures are then transferred to a specific domain by adding them to randomly sampled data from the domain's healthy class. The ability of the generated data to represent true domain fault data is imposed in part (B) by an adversarial discriminator. The semantic plausibility of the generated data to represent a desired fault class (as sampled from the sampling module) is tackled by a classifier in part (C). The different parts of the framework are detailed below.

The underlying hypothesis: The proposed approach is based on the hypothesis that the Fourier spectrum of fault data can be expressed as the sum of (1) domain-specific components (the spectrum of a signal from normal operation) and (2) of fault-specific components representing the specific fault characteristics. Further, we assume that the latter (spectrum of a signal representing the specific faulty condition) corresponds to a general domain-independent fault signature that is adjusted to new domains simply by linear scaling. In other words, this hypothesis allows us to express Fourier coefficients [30] of the fault data of a certain class c from a specific domain \mathbb{X} ($x_{fault,\mathbb{X}}^{c,FFT}$) as a sum of (1) domain-specific characteristics that are represented by the healthy class data of that domain $x_{\mathbb{S}}^{FFT}$ and (2) the fault class specific characteristics that are domain-independent $x_{fault}^{c,FFT}$ and scaled by a factor w — see Eq. (1).

$$x_{fault,\mathbb{X}}^{c,FFT} = x_{\mathbb{X}}^{FFT} + w * x_{fault}^{c,FFT} \quad (1)$$

The linear scaling with w is performed to account for the fact that the fault signature is affected by operational changes and, therefore, we alleviate the strong assumption that the fault-specific variations of real fault data are independent from OCs. Between a source domain \mathbb{S} and target domain \mathbb{T} , the weight factor w is defined as in Eq. (2) based on the healthy data.

$$w = E(x_{h,\mathbb{S}}/x_{h,\mathbb{T}}) \quad (2)$$

(Part A) The generative model G_{θ} : The final goal is to generate faults in a target domain that have not been observed before based

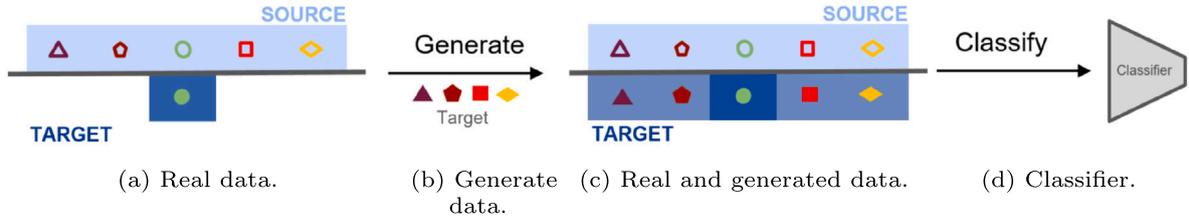


Fig. 3. FaultSignatureGAN in the Partial DA settings: the original data setting is depicted in Fig. 3(a); the missing target classes are generated in Fig. 3(b); the target dataset is augmented with synthetically generated data in Fig. 3(c) and a classifier is trained on the augmented dataset in Fig. 3(d).

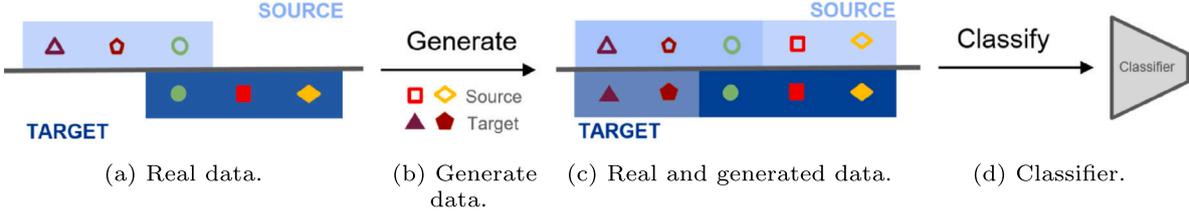


Fig. 4. FaultSignatureGAN in the Open-Partial DA settings: the original data setting is depicted in Fig. 4(a); the missing source and target classes are generated in Fig. 4(b); the source and target dataset is augmented with synthetically generated data in Fig. 4(c) and a classifier is trained on the augmented dataset in Fig. 4(d).

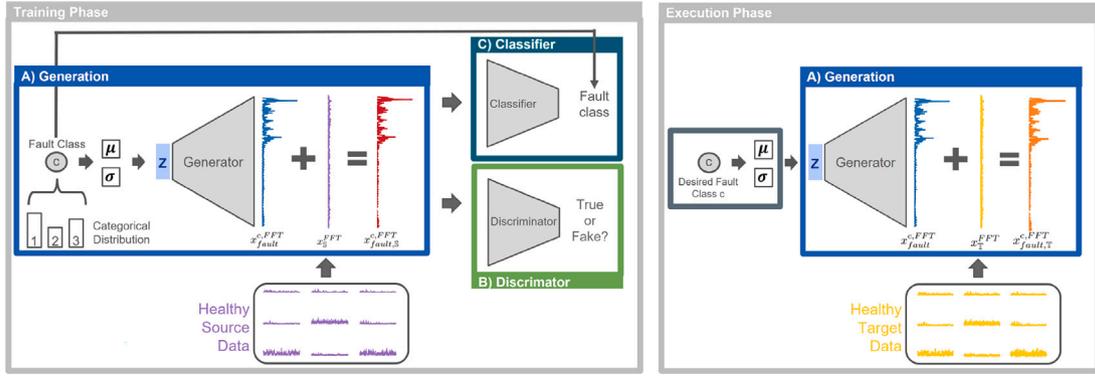


Fig. 5. FaultSignatureGAN: **Training Phase:** Training the (A) generative model to generate domain independent fault characteristics while imposing (B) plausibility with the discriminator in the source domain and (C) semantic consistency with the classifier. **Execution Phase:** The generation of unseen target data.

on Eq. (1). From the two components in Eq. (1), we only have access to the healthy data representing the domain-specific variations (x_T^{FFT}) in the target domain \mathbb{T} . Therefore, to generate unseen faults in the target domain $x_{fault,\mathbb{T}}^{c,FFT}$, we need to design a framework that enables the generation of the domain-independent characteristics of a fault class $x_{fault}^{c,FFT}$. We train the proposed architecture on the data from the source domain, where we do not only have access to the healthy data $x_{h,\mathbb{S}}^{FFT}$ but also to true fault data $x_{fault,\mathbb{S}}^{c,FFT}$. In the source domain \mathbb{S} , the scaling factor w equals to 1. Due to the variability in the healthy class, simply subtracting the individual healthy samples from faulty ones (the reverse operation) is not sufficient to retrieve a domain independent fault signature. Therefore, we propose a generative model. The generative model is trained such that its output (blue signal in Fig. 5) can be transformed in a real source fault by adding it to a healthy source sample (according to Eq. (1)). Thus, the generated signal can be transformed to real domain faults with any of the samples from the healthy data distribution. This process is depicted in Fig. 5. To ensure plausibility of the generated signals in the specific domain, the generator is trained to fool a discriminator D_w (see below). To ensure semantic or class consistency, we condition the generative model on the desired fault class by simply sampling the distinct desired fault class from a categorical distribution. Each of the discrete values drawn from the categorical distribution corresponds to a specific fault class. The probability of each category is defined based on the class distribution in the training dataset $T_{f,\mathbb{S}}$, from which the fault signatures should

be learned. In other words, the probability of category i is defined by Eq. (3). The value sampled from the uniform distribution is then passed to two vectors (μ and σ in Fig. 5) that parameterize a Gaussian distribution (mean and deviation), from which we sample using the reparametrization trick [31]. The generative model is updated based on the consistency of the desired class with the classifier's prediction (see below).

$$p_i = \frac{|\{(x_j, y_j) | (x_j, y_j) \in T_{f,\mathbb{S}} \& (y_j = i)\}|}{|T_{f,\mathbb{S}}|} \quad (3)$$

To enable a better distinction, we will refer to the signal representing the domain independent fault characteristics (depicted in blue in Fig. 5) as the *generated fault signature*, and to the signal representing the domain-specific fault data (depicted in red or orange in Fig. 5) as the *generated data sample* throughout the paper. Further, in the following, we will consider the data always in the Fourier domain without emphasizing it specifically.

(Part B) The discriminator D_w : We need to ensure that the generated data represents plausible domain data. Due to the absence of real target faults, the discriminator is trained to discriminate between real fault data of the source domain and the generated source data. We implement a Wasserstein GAN that is optimized with gradient penalty since its training has proven to be more stable compared to other GAN implementations, mitigating mode collapse [12]. The adversarial loss function is defined by Eq. (4).

$$L_D = \mathbb{E}_{\tilde{x}^c, x_{h,\mathbb{S}}} [D_w(\tilde{x}^c + x_{h,\mathbb{S}})] - \mathbb{E}_{x_{f,\mathbb{S}}^c} [D_w(x_{f,\mathbb{S}}^c)]$$

$$+ \lambda_{GP} \mathbb{E}_{\tilde{x} \sim \mathbb{P}_{\tilde{x}}} [(\|\nabla_{\tilde{x}} D_w(\tilde{x})\|_2 - 1)^2], \quad (4)$$

where D_w is the discriminator model, \tilde{x} is a generated fault signature, $x_{h,\mathbb{S}}$ is a healthy source sample, $x_{f,\mathbb{S}}$ is a faulty source sample and \hat{x} is drawn from $\mathbb{P}_{\tilde{x}}$, a newly defined data distribution used to impose the gradient penalty [12].

(Part C) The classifier C_γ : A classifier is added to the framework to ensure semantic consistency of the generated data to a desired class. The classifier is optimized with the semi-hard triplet loss [32] on real source data. In Eq. (5), the corresponding loss function is shown, where C_γ is the classifier network, α is a fixed margin, x_a is the anchor sample, x_p the positive sample and x_n the negative one.

$$L_C = \max(\|C_\gamma(x_a) - C_\gamma(x_p)\|^2 - \|C_\gamma(x_a) - C_\gamma(x_n)\|^2 + \alpha, 0) \quad (5)$$

For updating the generative model G_θ , the semi-hard triplet loss is calculated using only synthetic data ($\tilde{x} + x_{h,\mathbb{S}}$) as anchors and real fault data $x_{f,\mathbb{S}}$ as positive resp. negative samples. The pseudo algorithm of the **Training Phase** is shown in Algorithm 1.

3.2. The generation of unseen data in the execution phase

After training the generative model G_θ in the **Training Phase**, the generation of target faults in the **Execution Phase** is straight forward: First, we sample the input of the generative model from a categorical distribution, which determines the desired fault classes to generate. The number of generated samples per class can be chosen freely. This input is then passed to the generative model to generate the respective fault class signatures. The fault signature is then transferred into the target domain (instead of the source domain) by (1) linearly scaling the fault signature (with w) and (2) adding it to the healthy data of the target domain (yellow data in **Execution Phase** of Fig. 5). The scaling of the fault signature is defined as the ratio between the mean signal of the healthy source data and the mean of the healthy target data per frequency component (as defined in Eq. (2)). Hence, the unseen target data is generated as defined in Eq. (6).

$$x_{fault,\mathbb{T}}^{c,FFT} = x_{\mathbb{T}}^{FFT} + w * x_{fault}^{c,FFT} \quad (6)$$

Algorithm 1 Training Phase of *FaultSignatureGAN*

Require: $T_{\mathbb{S}}$ (Source Dataset); $\lambda_{GP}, \lambda_D, \lambda_E, \alpha$ (Loss Function Parameter); n_{critic} , es (Early Stopping Criteria), m (Batch Size)

Ensure: G_θ

▷ **Prepare Dataset**

$T_{h,\mathbb{S}} = \{(x, y) \in T_{\mathbb{S}} \mid y \text{ is healthy}\}; T_{f,\mathbb{S}} = \{(x, y) \in T_{\mathbb{S}} \mid y \text{ is a fault class}\}$

$Cat(T_{f,\mathbb{S}})$ Categorical Distribution of the classes in $T_{f,\mathbb{S}}$

while $es == \text{False}$ **do**

for $t = 1, \dots, n_{critic}$ **do**

▷ **Sample data batches**

$\{z^{(i)}\}_{i=0}^m \sim Cat(T_{f,\mathbb{S}})$

$\{(x_{f,\mathbb{S}}, y_{f,\mathbb{S}})\}_{i=0}^m \sim T_{f,\mathbb{S}}; \{x_{h,\mathbb{S}}\}_{i=0}^m \sim T_{h,\mathbb{S}}; \{(x_{\mathbb{S}}, y_{\mathbb{S}})\}_{i=0}^m \sim T_{\mathbb{S}}$

$\epsilon \sim U[0, 1]$

▷ **Generate data**

$\tilde{x} \leftarrow G_\theta(z)$

$\tilde{x}_f \leftarrow \tilde{x} + x_{h,\mathbb{S}}$

$\hat{x} \leftarrow \epsilon x_{f,\mathbb{S}} + (1 - \epsilon)\tilde{x}_f$

▷ **Update discriminator D**

$L_D^i \leftarrow D_w(\tilde{x}_f) - D_w(x_{f,\mathbb{S}}) + \lambda_{GP}((\|\nabla_{\tilde{x}} D_w(\hat{x})\|_2 - 1)^2)$

$w \leftarrow Adam(\nabla_w 1/m \sum_{i=1}^m L_D^i, w)$

end for

▷ **Update classifier C**

 From $\{x_{h,\mathbb{S}}\}_{i=0}^m$ form triplets [32] according to label $x_{\mathbb{S},a}$, $x_{\mathbb{S},p}$ and $x_{\mathbb{S},n}$

$L_C^i \leftarrow \max(\|C_\gamma(x_{\mathbb{S},a}) - C_\gamma(x_{\mathbb{S},p})\|^2 - \|C_\gamma(x_{\mathbb{S},a}) - C_\gamma(x_{\mathbb{S},n})\|^2 + \alpha, 0)$

$\gamma \leftarrow Adam(\nabla_\gamma 1/m \sum_{i=1}^m L_C^i, \gamma)$

▷ **Update generator G**

$L_D^i \leftarrow -D_w(G_\theta(z))$

$L_C^i \leftarrow \max(\|C(\tilde{x}_f) - C(x_{\mathbb{S},p})\|^2 - \|C(\tilde{x}_f) - C(x_{\mathbb{S},n})\|^2 + \alpha, 0)$

$L_G^i \leftarrow \lambda_D * L_D^i + \lambda_C * L_C^i$

$\theta \leftarrow Adam(\nabla_\theta 1/m \sum_{i=1}^m L_G^i, \theta)$

end while

return G_θ

3.3. Alternative approaches used for comparison

In this work, we address two DA settings with label space discrepancies: *Partial* DA and *Open-Partial* DA, with a particular focus on the extreme case where only one class is shared between two domains. While for *Partial* DA, some approaches have been proposed, only few are suitable for this extreme scenario. These few approaches are used for comparison for the *Partial* DA experiments. (1) First, we report the *Baseline* results, where we train a classifier on real source domain data only. It shows the minimal achievable performance if no adaptation is performed. (2) The adversarial feature alignment approach *Unilateral* DA [10] is chosen as a comparison method as it has been evaluated in a *Partial* DA setting before (as elaborated in Section 2). It aims to achieve the same goals but uses a different strategy (feature alignment vs. data generation in our proposed framework). While originally proposed as a completely unsupervised DA method, the authors also conducted experiments on the extreme scenario (where only the healthy class is shared between the two domains). For these experiments, the healthy data label from the target domain was used for alignment [10]. We compare our method to both implementations and denote the completely unsupervised implementation as *Unilateral* and the one using the target's healthy label for alignment as *Unilateral**. (3) The adversarial approach *BA3US* balances each batch of target data with randomly sampled source data. It, therefore, presents an interesting comparison method to the proposed *FaultSignatureGAN*, where we balance the target domain with generated data that has been mapped to the target domain in an unsupervised manner. (4) Last, the data generation approach *GenAlign* is used for comparison [20] (see Section 2), where target data is generated by passing novel input to the generative model. This approach is used to challenge the hypothesis that generative models are limited in their extrapolation abilities and therefore, the novel target data generation should not rely on extrapolation abilities of the model (as we do in our work).

For the *Open-Partial* domain experiments, however, there is no other suitable comparison method that is applicable to the same extreme case scenario as we consider here where only the healthy class is shared between the two domains. Therefore, only the *Baseline* is used for comparison in these experiments.

4. Case studies

The proposed approach is tested on two bearing datasets that have been applied for DA tasks in fault diagnostics. Our proposed framework is evaluated on both datasets in *Partial* and *Open-Partial* DA experiments. Both datasets are adjusted to the problem formulation of the respective DA setup.

4.1. CWRU

The CWRU dataset is a publicly available benchmark bearing dataset (bearing type SKF 6205) provided by the Case Western Reserve University Bearing Data Center (CWRU dataset) [33]. The data was collected on a test rig in laboratory conditions. It contains data recorded under four different loads (referred to as domain 0,1,2 and 3). The different load settings resulting in different rotational speeds are shown in Table 1. Data under healthy and nine different faulty conditions is available: Three fault types – Ball, Inner Race and Outer Race – with three severity levels each. An overview of the fault types and severity levels is shown in Table 2. The CWRU dataset has been extensively used to demonstrate *ClosedSet* DA methods under different OCs as well as for *Partial* DA setups [10,19].

Table 1

OCs under which the two case studies (CWRU and Paderborn) are recorded. Each setting corresponds to one domain.

| Domain | CWRU | Paderborn | | | |
|--------|------------------------|------------------------|------------------|------------------|--------------|
| | Rotational speed [rpm] | Rotational speed [rpm] | Load torque [Nm] | Radial force [N] | Setting name |
| 0 | 1797 | 1500 | 0.7 | 1000 | N15_M07_F10 |
| 1 | 1772 | 900 | 0.7 | 1000 | N09_M07_F10 |
| 2 | 1750 | 1500 | 0.1 | 1000 | N15_M01_F10 |
| 3 | 1730 | 1500 | 0.7 | 400 | N15_M07_F04 |

Table 2

Health conditions represented in the case studies (CWRU and Paderborn).

| | | Healthy | Outer race (OR) | Inner race (IR) | Ball (B) |
|-----------|------------------|---------|-----------------|-----------------|----------|
| CWRU | Size | – | 7 14 21 | 7 14 21 | 7 14 21 |
| | Class | 0 | 1 2 3 | 4 5 6 | 7 8 9 |
| Paderborn | Extent of damage | – | 1 2 – | 1 2 3 | – – – |
| | Class | 0 | 1 2 – | 3 4 6 | – – – |

4.2. Paderborn

The Paderborn dataset is a publicly available bearing dataset (bearing type SKF 6203) provided by the Chair of Design and Drive Technology from Paderborn University [34]. It incorporates both artificially induced bearing faults and realistic damages caused by accelerated lifetime tests under different OCs (rotational speed, load torque and radial force). In this study, we only consider real fault data and not the artificially induced one. The represented health conditions in the dataset are healthy conditions, Inner Race faults (three severity levels) as well as Outer Race faults (two severity levels). The different OCs are shown in Table 1 and the different classes in Table 2. The data was collected on a test rig under laboratory conditions and was also previously used in different DA studies [10]. Previous publications mainly focused on 3-class classification of the different fault types [10] and suggested that the domain gaps in the Paderborn dataset are larger compared to the CWRU dataset. Further, previous publications typically neglected the domain 1, since the domain gap to the other domains is considerably large compared to the other domain gaps. In this research, we focus on the type and severity classification (6-class classification) and also aim to bridge large domain gaps. Contrary to previous works, we, therefore, report results on all possible domain shifts with and without domain 1 in our DA evaluation.

Moreover, we use less data compared to previous publications, such as e.g. [10], for our evaluation (only using the datasets K002-5; KA04, KA15-16 and KI16,18,21 whereas KA22, KA30, KI04 and KI14 have not been used in this study). This enables us to evaluate if we can extract transferable fault characteristics from only limited fault data.

5. Experimental setup

To test if *FaultSignatureGAN* is capable of generating unseen domain faults, *Partial* and *Open-Partial* DA experiments are conducted, whereby the different domains correspond to the different OCs in the case studies. The experimental setups are shown in Table 3 (*Partial*) and Table 4 (*Open-Partial*) on an exemplary domain shift from a source domain X to a target domain Y ($X \rightarrow Y$).

The experiments are conducted as follows: First, a generative model is trained on data from one domain (as described in Section 3 and depicted in **Training Phase** in Fig. 5). In this work, we train one generative model to generate all severity levels of one fault type. Second, the label space of the target domain is completed by generating synthetic target fault data as depicted in **Execution Phase** in Fig. 5 based on healthy target data. The number of generated data samples

Table 3

Experimental setup for *Partial* DA on an exemplary domain shift $X \rightarrow Y$.

| <i>Partial</i> | | | | | |
|----------------|-------------------|---------------|---------------------|---------------|--------------------------------|
| Dataset | Domain shift | Source domain | Source classes | Target domain | Target classes during training |
| CWRU | $X \rightarrow Y$ | X | 0,1,2,3,4,5,6,7,8,9 | Y | 0 |
| Paderborn | $X \rightarrow Y$ | X | 0,1,2,3,4,5 | Y | 0 |

Table 4

Experimental setup for *Open-Partial* DA on an exemplary domain shift $X \rightarrow Y$ on the Paderborn dataset.

| <i>Open-Partial</i> | | | | | |
|---------------------|-------------------|---------------|----------------|---------------|--------------------------------|
| Task | Domain shift | Source domain | Source classes | Target domain | Target classes during training |
| Source (IR) | $X \rightarrow Y$ | X | 0,3,4,5 | Y | 0,1,2 |
| \neq | | | | | |
| Target (OR) | | | | | |
| Source (OR) | $X \rightarrow Y$ | X | 0,1,2 | Y | 0,3,4,5 |
| \neq | | | | | |
| Target (IR) | | | | | |

per class is chosen to match the mean number of samples per class in the source domain. In the third step, a new training dataset is composed of the generated and real data from both domains and used to train a classification model. The performance of the classifier is then evaluated on a test dataset composed of all unseen faults and 30% of the data from each of the classes, from which the conditions have been observed before.

Hyperparameter Tuning: Data-driven solutions based on neural networks come with many hyperparameters to tune including those of the network architecture. These choices strongly influence the performance of the final model including its generalizability to new data. There is no commonly accepted procedure for optimizing the hyperparameters for an unknown target domain [35]. Some works rely, therefore, on a target validation dataset [35] or validation tasks [10]. In the context of safety-critical systems where no target fault data is available, this is not possible. Hence, in this work, we do not make the assumption of having target data available for hyperparameter tuning, since it is a strong limitation of applying existing DA methods to real PHM applications. For training the **generative model (Training Phase** in Fig. 5), only criteria related to the source dataset are used: In addition to optimizing the loss functions (see Section 3) on the source dataset, a stopping criterion is implemented. The training is stopped if an auxiliary classifier trained on the synthetic source data returns an accuracy of at least 98%, evaluated on the real source data. Since this callback function is computationally expensive, it is only executed after each 50 epochs of training. Further, the hyperparameters of the final **classification model** need to be tuned as well. In absence of real target fault data, we used synthetically generated data as a validation dataset. To showcase and evaluate the impact that hyperparameter settings have on the ability of the model to generalize to an unseen domain, we trained three different model architectures: Model (1) equals the one used in previous publications [10,20], Model (2) equals Model (1) but has the ReLU activation function and Model (3) equals Model (1) except that the kernel size is set to 12 (compared to 3 used in [20]). Exemplary, we only evaluate the domain shifts from source domain 0 on the CWRU dataset for hyperparameter tuning. The final accuracies on a source validation dataset, a synthetic fault dataset as well as on the true target test dataset of the three models are shown in Fig. 6. The performance on the target dataset varies considerably depending on the architecture used. For example, on domain shift $0 \rightarrow 2$, the final performance on the target dataset varies by 10% depending on the model used. This evaluation shows clearly that even small changes in hyperparameters can impact the generalization ability strongly i.e. have a big effect on the performance in the target domain. The source

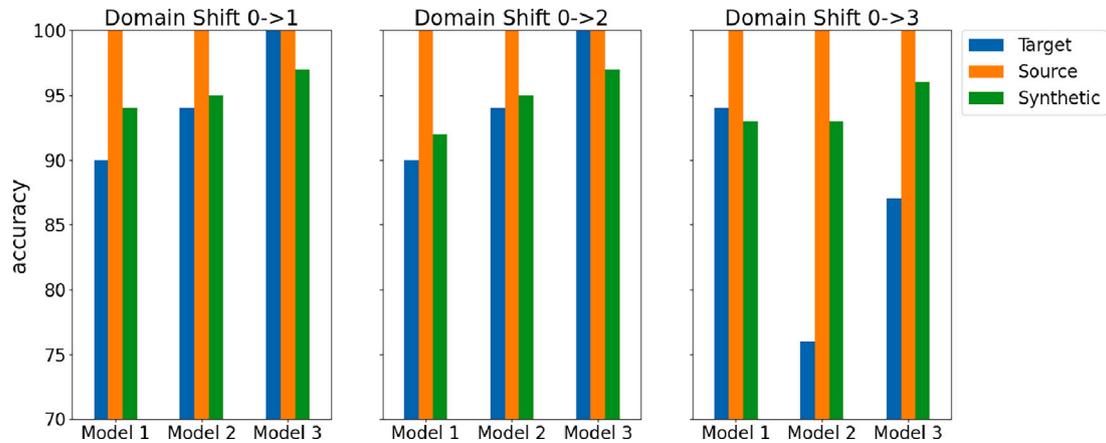


Fig. 6. Visualization of the effect of hyperparameter tuning on the generalizability of different model architectures on different domain shifts: Model 1 as in [10], Model 2 as in [10] but with the ReLU activation function, Model 3 as in [10] with the kernel size of 12 (compared to 3 used in [10]). Three different datasets are used for evaluation: (1) source validation dataset (orange); (2) dataset with synthetic faults (green), and (3) the real target dataset (blue).

validation dataset does not provide a good indication which model to choose since the performance on the source dataset always results in 100%. The accuracy of the synthetically generated dataset does not correlate strongly with the target accuracy in all instances. However, it gives a clear indication to choose Model 3 in all instances. This is also the best choice for the highest accuracy on the target dataset in the first two domain shifts. Only for domain shift $0 \rightarrow 3$ this is not an ideal choice. Although not ideal, we want to emphasize that the synthetic validation dataset provides information on which model to choose compared to the source validation dataset. On average, that information results in the best final model choice. Therefore, we conduct our experiments on Model (3) for the CWRU dataset. We train the classification model for 2000 epochs (since the source as well as the synthetic validation datasets suggest that no considerable change happens after 2000 epochs). To enable a better comparison, we use only one model architecture for all domain shifts per case study.

Apart from using a synthetic validation dataset, we propose to use the following strategy for certain hyperparameters: (1) Applying a heavy regularization (since it leads to better generalization); (2) running the optimization for multiple epochs — more than indicated by the validation dataset. The latter choice is motivated by the findings of learning theory that hypothesize that there are two phases of deep learning: a fitting and a compression phase. It is indicated that the latter is responsible for the excellent generalization performance of deep networks [36]. The final model architectures being used are elaborated in Appendix.

Label Availability: Similarly to previous studies conducted on the extreme case of label discrepancy [10,19], we assume to know the label of the healthy data in the target domain for the *Partial* and *Open-Partial* DA experiments. Since healthy data is ubiquitous, this is considered to be a realistic assumption. In addition, for the *Open-Partial* setup, we assume that if at training data acquisition time fault classes have been observed, we also have the labels for the fault classes, both for source and target domains. This is a particularly realistic setup for fleets of complex systems with different ages for each of the units (each of the domains). Some units will have experienced one subset of fault types and other units will have experienced another subset of fault types. However, at testing time, we would like to be able to diagnose all of the fault types for all of the units.

Data Pre-Processing: To enable a fair comparison, the datasets are pre-processed in the same way as in previous publications [10,20]. The CWRU datasets are first truncated (at 12000 timesteps) and divided into 200 sequences of 1024 points. After applying the Fast Fourier Transform [30], only the first 512 coefficients are used (excluding the first one). The same process is applied to the Paderborn dataset. However, the data is not truncated and the 1024 long samples are sliced with a stride of 4096.

6. Evaluation and results

Partial (see Section 6.1) as well as *Open-Partial* DA experiments are conducted (see Section 6.2) to test the ability of the generated data to bridge domain gaps in different DA settings. To evaluate the physical plausibility of the generated data qualitatively, we visualize the generated data (see Section 6.3).

6.1. Partial DA

First, we conduct experiments in the extreme case of *Partial* DA where only the healthy class is shared between the domains. The experimental setup is exemplified in Fig. 7 for the Paderborn dataset, where only the healthy target data is available. First, the missing fault data in the target domain is generated (darker blocks in Fig. 7) to train a classifier model, which is evaluated on a target test dataset.

CWRU: We compare the results of the proposed methodology with the methods outlined in Section 3.3. If available, the exact results from previous publications are shown. If not, we re-implement the methods while using the same setup as in the original publications. Only for *Ba3US*, a new architecture needed to be tuned as elaborated in Appendix since this method has not yet been applied to any of the presented case studies. Since hyperparameter tuning without fault data did not lead to satisfying results, we followed the protocol of Wang et al. [10] and used the domain shift experiment $0 \rightarrow 3$ as a validation task. For the source-only experiments, we report, on the one hand, the previously reported results based on the originally proposed classifier architecture [10] with a kernel size of 3 (referred to as *Baseline*). On the other hand, we report the results of the source-only experiments conducted with the classifier architecture optimized based on the synthetic data as reported in Section 5. This architecture has a kernel size of 12 (referred to as *Baseline^{syn}*). The balanced accuracy of all experiments is shown in Table 5. The overall performance of all approaches is high — even for the two source-only *Baselines*, suggesting that the domain gaps are small in this dataset. This also leaves only limited room for improvement. The baseline model with optimized hyperparameters (*Baseline^{syn}*), however, outperforms the existing *Baseline* by 1.71%, showcasing that the generated data is beneficial for hyperparameter tuning. Moreover, adding the data generated by *FaultSignatureGAN* to the training dataset, results in an additional improvement of 0.96% — resulting in a total improvement of 2.67% compared to the previously reported baseline method. This shows that the generated data is beneficial in order to bridge domain gaps. The method based on data generation used for comparison, (*GenAlign*), where the generative model is conditioned on novel input data performs worse than *FaultSignatureGAN*. These results suggest

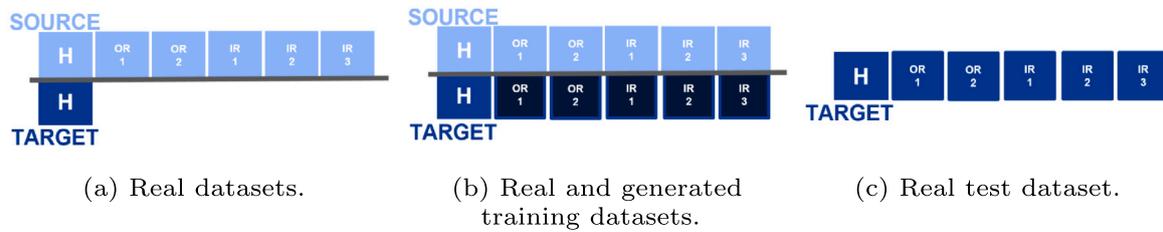


Fig. 7. Example of an extreme experimental *Partial DA* Settings on the Paderborn case study. In Fig. 7(a) The real datasets are shown where only the healthy class is shared between the source and the target dataset and the source dataset has five private classes. In Fig. 7(b) The training dataset is shown where the missing fault classes in the target domain are synthetically generated and in Fig. 7(c) the test dataset consisting of real target data is shown.

Table 5

Extreme *Partial DA* results on the CWRU dataset (10-class classification) under all domain shifts.

| Domain shift | Baseline ^a | Baseline ^{S_{syn}} | Unilateral ^b | Unilateral* ^a | BA3US | Gen Align [20] ^a | <i>Fault Signature GAN</i> |
|--------------|-----------------------|-------------------------------------|-------------------------|--------------------------|--------------|-----------------------------|----------------------------|
| 0 → 1 | 93.49 ± 1.75 | 99.49 ± 0.06 | 97.04 ± 0.86 | 98.08 ± 0.16 | 91.07 ± 3.98 | 97.81 | 99.87 ± 0.07 |
| 0 → 2 | 93.65 ± 0.96 | 99.96 ± 0.02 | 96.38 ± 2.34 | 99.56 ± 0.18 | 91.12 ± 1.28 | 96.02 | 99.36 ± 0.36 |
| 0 → 3 | 91.02 ± 0.02 | 90.27 ± 0.69 | 94.14 ± 0.56 | 98.22 ± 0.65 | 96.33 ± 1.71 | 94.24 | 94.50 ± 1.10 |
| 1 → 0 | 97.93 ± 0.93 | 96.79 ± 0.45 | 97.48 ± 0.45 | 98.08 ± 0.32 | 96.98 ± 1.02 | 97.27 | 97.62 ± 0.19 |
| 1 → 2 | 100.00 ± 0.00 | 100.00 ± 0.00 | 100.00 ± 0.00 | 100.00 ± 0.00 | 99.95 ± 0.04 | 96.32 | 99.95 ± 0.00 |
| 1 → 3 | 98.26 ± 1.63 | 99.46 ± 0.19 | 98.40 ± 0.91 | 99.20 ± 0.19 | 99.23 ± 0.66 | 94.59 | 99.35 ± 0.11 |
| 2 → 0 | 91.63 ± 1.63 | 96.15 ± 0.15 | 90.13 ± 3.66 | 96.43 ± 0.43 | 93.7 ± 3.59 | 95.44 | 96.50 ± 0.16 |
| 2 → 1 | 97.09 ± 0.09 | 97.78 ± 0.09 | 97.84 ± 0.26 | 97.48 ± 0.40 | 95.58 ± 1.93 | 96.55 | 97.06 ± 0.09 |
| 2 → 3 | 99.78 ± 0.17 | 99.63 ± 0.12 | 99.71 ± 0.10 | 98.97 ± 0.21 | 99.75 ± 0.21 | 96.13 | 99.63 ± 0.09 |
| 3 → 0 | 87.96 ± 0.18 | 88.58 ± 0.19 | 86.50 ± 4.56 | 94.85 ± 2.16 | 86.75 ± 0.21 | 92.82 | 92.81 ± 0.92 |
| 3 → 1 | 89.42 ± 0.42 | 92.68 ± 0.64 | 93.22 ± 0.97 | 96.18 ± 0.50 | 85.53 ± 3.19 | 93.04 | 95.41 ± 0.21 |
| 3 → 2 | 99.65 ± 0.17 | 99.68 ± 0.44 | 99.82 ± 0.04 | 99.78 ± 0.09 | 98.68 ± 2.23 | 95.49 | 99.99 ± 0.01 |
| Mean | 94.99 | 96.70 | 95.88 | 98.07 | 94.56 | 95.49 | 97.66 |

^aResults as reported in the original publication.

^bModels used as in original publication for reproducing results.

that it is not beneficial to condition the generative model on unseen input and rely on extrapolation abilities of the generative model as in *GenAlign*. Therefore, this approach is not used as comparison method in the following experiments on the Paderborn dataset. From the two unsupervised adversarial alignment approaches (*Unilateral* and *BA3US*), the *Unilateral* approach performed consistently better on all domain shifts. *Unilateral**, where the label of the healthy target data is used for alignment, results in the highest performance compared to all other approaches. On average, *FaultSignatureGAN* performs within the same performance range as *Unilateral**. In the following experiments on the Paderborn dataset, only *Unilateral** is used as a comparison method.

Paderborn: In this case study, we only use the best performing comparison DA method *Unilateral**. Contrary to other publications on the Paderborn case study that focus only on fault type classification (3-class classification), we focus on the task of fault type and severity classification in this work (6-class classification). Further, previous publications did not include domain 1 in their experiments [10], where the domain gap is large compared to the other domain gaps (as indicated by the *Baseline* results). In this work, we report the results for all DA tasks related to the domain 1 separately in the upper part of Table 6 to indicate the different domain gap sizes. The DA approaches (*Unilateral** and *FaultSignatureGAN*) outperform the *Baseline* on all domain shift experiments — see Table 6. While the performance gain is comparable between the two approaches on small domain gaps (*domains 0,2* and *3*), *FaultSignatureGAN* results in a considerably better performance on *domain 1*, where the domain gap is large. In all settings, there is a substantial relative gain. On *domains 0,2*, and *3*, an average improvement of 3.82% was achieved by *FaultSignatureGAN* compared to the *Baseline*. On *domain 1*, the absolute improvement is even 23.76%. If *domain 1* is the target domain, the absolute performance of all approaches is still rather low (<50%) despite the relatively high improvement. In the opposite direction, when *domain 1* is the source domain, higher absolute results were achieved (average performance of the three domain shift experiments with *FaultSignatureGAN* is 79.72%). Although the domain gap should be the same in both directions (domain as source or target),

Table 6

Extreme *Partial DA* results on the Paderborn dataset (6-class classification). In the upper part all results with domain shifts including *domains 0,2* and *3* are shown and in the lower part all domain shifts including *domain 1*.

| Domain shift | Baseline | Unilateral* | <i>FaultSignatureGAN</i> |
|--------------|---------------------|---------------------|--------------------------|
| 0 → 2 | 99.78 ± 0.06 | 99.92 ± 0.02 | 99.76 ± 0.24 |
| 0 → 3 | 69.49 ± 0.73 | 69.98 ± 1.73 | 72.08 ± 1.24 |
| 2 → 0 | 99.42 ± 0.01 | 99.67 ± 0.08 | 99.54 ± 0.16 |
| 2 → 3 | 74.66 ± 0.40 | 75.71 ± 1.69 | 75.49 ± 0.23 |
| 3 → 0 | 67.43 ± 0.6 | 71.29 ± 1.61 | 78.87 ± 0.65 |
| 3 → 2 | 68.37 ± 1.5 | 77.14 ± 1.42 | 76.35 ± 0.42 |
| Mean | 79.86 | 82.29 | 83.68 |
| 0 → 1 | 22.88 ± 1.51 | 29.37 ± 1.20 | 45.88 ± 2.21 |
| 1 → 0 | 58.66 ± 1.71 | 74.54 ± 0.53 | 84.34 ± 0.21 |
| 1 → 2 | 63.28 ± 1.77 | 75.16 ± 3.56 | 86.34 ± 0.77 |
| 1 → 3 | 47.99 ± 0.22 | 61.87 ± 1.77 | 68.50 ± 0.51 |
| 2 → 1 | 21.77 ± 0.53 | 29.96 ± 1.83 | 47.59 ± 1.66 |
| 3 → 1 | 22.77 ± 1.31 | 26.47 ± 0.24 | 45.30 ± 0.61 |
| Mean | 39.23 | 49.56 | 62.99 |

this difference in the performance could potentially be explained if the fault data in *domain 1* shows more variability compared to the other domains. This leads to better generalization on tasks from *domain 1* to other domains. Especially in these instances, *FaultSignatureGAN* shows a superior performance.

6.2. Open-partial DA experiments

To showcase the versatility of our framework, we conduct *Open-Partial DA* experiments in addition to the *Partial DA* experiments (Section 6.1). These experiments are only conducted on the Paderborn dataset since the domain gaps are larger compared to the CWRU dataset. The other DA methods used for comparison for the *Partial DA* setup are not directly applicable for the *Open-Partial* setup. Therefore, we only report the *Baseline* results for comparison, where the classifier

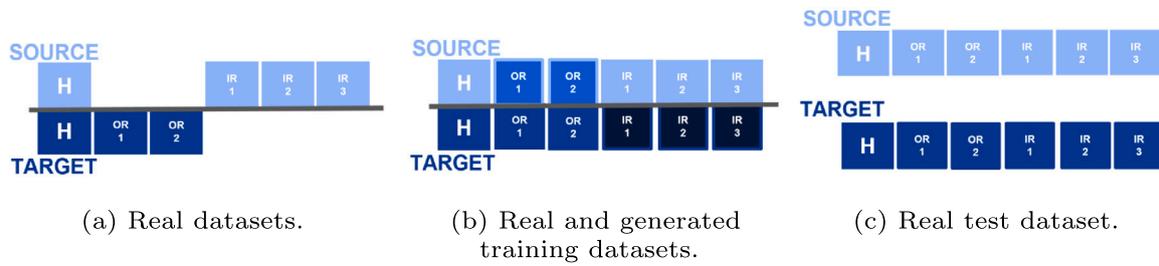


Fig. 8. Example of an extreme experimental *Open-Partial* DA Settings on the Paderborn case study. In Fig. 7(a) the real datasets are shown where only the healthy class is shared between the source and the target dataset and the source dataset has five private classes. In Fig. 8(b) The training dataset is shown where the missing fault classes in the target domain are synthetically generated and in Fig. 8(c), the two test datasets are shown where one consists of the real source and the other of the real target data.

Table 7

Open-Partial DA results on the Paderborn dataset (6-class classification). In the upper part all results with domain shifts including domains 0,2 and 3 are shown and in the lower part all domain shifts including domain 1. As shown in Fig. 8, the trained classification model is evaluated on two datasets: The source test dataset (S) and the target test dataset (T).

| | | Domain shift | Baseline | Proposed | | | Baseline | Proposed |
|---------------------------------------|-------------------|--------------|------------------|------------------------------------|---------------------------------------|--------------|------------------|------------------------------------|
| Source (IR) \Rightarrow Target (OR) | 0 \Rightarrow 2 | S | 99.41 \pm 0.15 | 99.81 \pm 0.02 | Source (OR) \Rightarrow Target (IR) | S | 99.12 \pm 0.42 | 99.96 \pm 0.05 |
| | | T | 99.88 \pm 0.01 | 99.97 \pm 0.02 | | T | 99.53 \pm 0.14 | 99.85 \pm 0.10 |
| | 0 \Rightarrow 3 | S | 73.65 \pm 0.13 | 91.74 \pm 0.05 | | S | 78.99 \pm 0.69 | 91.59 \pm 0.18 |
| | | T | 72.80 \pm 0.51 | 95.33 \pm 0.01 | | T | 75.41 \pm 0.48 | 96.18 \pm 0.75 |
| | 2 \Rightarrow 3 | S | 75.30 \pm 1.04 | 93.99 \pm 0.61 | | S | 78.10 \pm 1.17 | 97.13 \pm 0.70 |
| | | T | 72.93 \pm 1.16 | 94.82 \pm 0.23 | | T | 75.31 \pm 0.81 | 91.67 \pm 0.77 |
| Mean | S | 82.79 | 95.18 | S | 85.40 | 96.22 | | |
| | T | 81.87 | 96.71 | T | 83.42 | 95.90 | | |
| Source (IR) \Rightarrow Target (OR) | 0 \Rightarrow 1 | S | 56.91 \pm 1.44 | 66.55 \pm 0.61 | Source (OR) \Rightarrow Target (IR) | S | 51.53 \pm 1.43 | 53.90 \pm 1.33 |
| | | T | 65.53 \pm 0.09 | 71.04 \pm 0.22 | | T | 69.06 \pm 1.28 | 81.48 \pm 1.53 |
| | 1 \Rightarrow 2 | S | 53.62 \pm 0.23 | 56.21 \pm 1.18 | | S | 51.93 \pm 1.21 | 65.65 \pm 1.10 |
| | | T | 69.71 \pm 0.16 | 76.08 \pm 1.41 | | T | 65.76 \pm 0.26 | 67.54 \pm 0.34 |
| | 1 \Rightarrow 3 | S | 51.83 \pm 1.66 | 66.10 \pm 1.40 | | S | 63.66 \pm 0.56 | 67.64 \pm 0.15 |
| | | T | 66.50 \pm 0.02 | 74.75 \pm 0.65 | | T | 65.25 \pm 0.06 | 71.14 \pm 0.62 |
| Mean | S | 54.12 | 62.95 | S | 55.71 | 62.39 | | |
| | T | 67.24 | 73.96 | T | 66.69 | 73.39 | | |

was trained on all available real data from the source and target domain. Fig. 8 depicts an example for the experimental setup: For the *Open-Partial* DA experiments, we assume that in each of the two domains, one fault type occurred with different severities. For example, the outer race fault with severity 1 and 2 (OR1 and OR2) occurred in the target domain, whereas the inner race fault (severity 1, 2 and 3; IR1, IR2 and IR3) occurred in the source domain. In a first step, two generative models are trained. The fault signature of the outer race fault is trained on the target data, the fault signature of the inner race fault on the source data (see **Training Phase** in Section 3). In a second step, the missing fault data is generated: In the example of Fig. 8, the outer race fault is generated for the source domain and the inner race fault classes with severity 1,2 and 3 for the target domain. This generated and real data composes the training dataset. In our experimental setup for *Open-Partial* DA, there is missing data in each of the domains. Therefore, we evaluate the performance on two test datasets — the source S and the target T. The test datasets comprise the real missing fault data as well as of a 30% of known health conditions. The results on the 6-class classification task are reported in Table 7.

The experiments show that the synthetically generated data enables to achieve a good classification performance above 90% on all shifts with small domain gaps, by far exceeding the performance of the *Baseline* method of 83.37% on average. On the DA task related to *domain 1*, the absolute performance of the classifier is lower, however, it still results in a large relative improvement in all instances compared to the *Baseline* method.

6.3. Qualitative evaluation

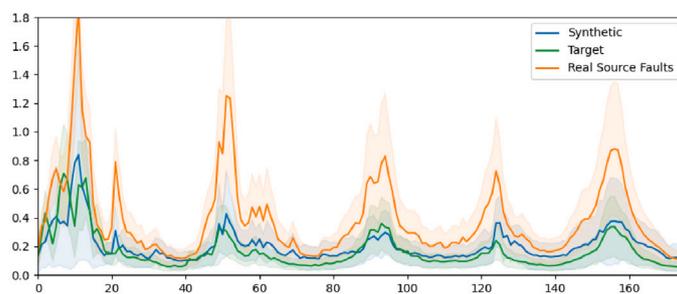
To evaluate the physical plausibility qualitatively, we visualize the mean and standard deviation of a batch of 1000 (1) generated signals

(blue line in Fig. 9(a)), (2) true faults in the target domain (orange line in Fig. 9(a)) and (3) true faults in the source domain (green line in Fig. 9(a)). Exemplary, we chose the *fault type* OR and *severity 1* on the domain shift 0 \rightarrow 1 on the Paderborn dataset. Additionally, to better visualize the differences, the residual of the generated target to the true target mean signal is visualized as well as the residual of the true source to the true target (which can be considered as the baseline) — see Fig. 9(b). The proposed framework generates the true target data well. Visually, it represents the target faults substantially better compared to the source data, especially in the higher frequency range.

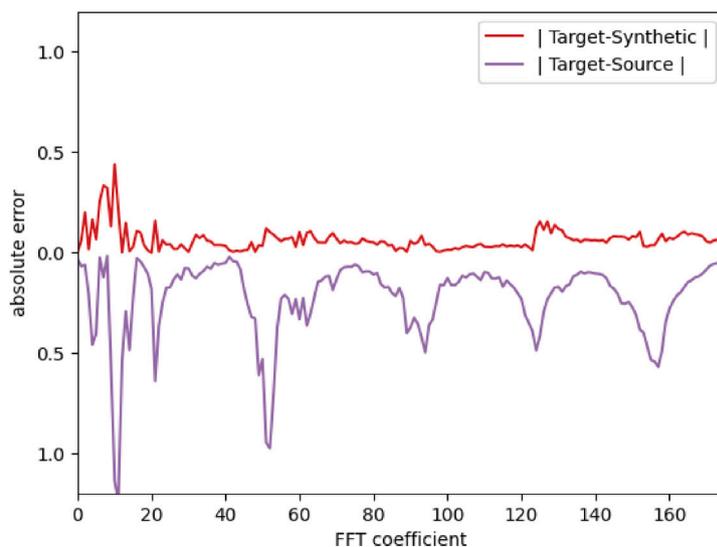
7. Discussion

The experiments performed in this research demonstrate the validity of the proposed framework *FaultSignatureGAN* to generate previously unobserved fault data, that can be used for DA with different types of extreme label discrepancies, where only the healthy class is shared between the domains. The obtained results open interesting points for discussion.

FaultSignatureGAN for DA with label space discrepancies: Given small domain gaps, *FaultSignatureGAN* outperforms most of the comparison methods, especially when comparing the results to *GenAlign*, the other generative approach. This particularly supports the assumption that the unsupervised generation of unseen data should not rely on extrapolation abilities of the generative model (as it does for the comparison methods). Instead, our approach enables the generation of unseen faults building on the hypothesis that domain-specific fault data can be disentangled in domain-specific characteristics and class specific ones and therefore, requires no extrapolation ability of the generative model. The comparison method *BA3US* outperforms *FaultSignatureGAN*



(a) Mean Fault Signal OR Severity 1 of the synthetic data (blue line), the true target fault class data (orange line) and the true source fault class data (green line).



(b) Mean Absolute Residual Signal OR Severity 1 of the Target to Synthetic (red line) and Target to Source (purple line).

Fig. 9. Paderborn data visualization of the OR severity 1 fault comparing real fault data with generated fault data.

solely on domain shift $0 \rightarrow 3$ (by 1.83%), where target data has been used to tune the hyperparameters of *BA3US*. In practical safety-critical applications, target data to tune the hyperparameters of a model is typically not available. Therefore, *FaultSignatureGAN* did not rely on this information and thus, satisfies more realistic requirements for PHM applications. The improvement that *BA3US* provided on that one domain shift could not be translated to other domain shifts. This, once again, showcases the importance of hyperparameter tuning for DA methods based on feature alignment and in particular the importance of the access to fault data for hyperparameter tuning. Only the feature alignment approach *Unilateral** provides a similar performance as *FaultSignatureGAN* under small domain gaps in *Partial* DA settings (see all results on CWRU in Section 6.1 and on Paderborn with domains 0,2 and 3). This is not surprising since synthetic data generation is never perfect. Therefore, when the domain gap is small, the source data represents the target data already quite well and one would expect little benefits in generating synthetic target specific data. On large domain gaps (those including domain 1 on the Paderborn dataset), however, the performance of the feature alignment method *Unilateral* drops. These are the scenarios where the proposed generative approach *FaultSignatureGAN* outperforms other approaches (see Sections 6.1 and 6.2). Therefore, if the size of the domain gap is unknown, *FaultSignatureGAN* is the best option to choose in safety-critical systems since

it provides a comparable performance under small domain gaps but a considerably better performance under large domain gaps.

Versatility of *FaultSignatureGAN*: Many different scenarios of label discrepancies are possible in real operations as exemplified in Section 1. Having one versatile method that can be applied in multiple of these scenarios is, therefore, utterly important for practical applications. The versatility of the proposed approach is demonstrated by applying it successfully to DA experiments with different types of label discrepancies (*Partial* and *Open-Partial*) (if the respective labels are known in the source domain), where it consistently outperforms other comparison methods under large domain gaps. We consider the versatility of the proposed approach as a one of the key benefits for practical PHM applications.

Plausibility of Unsupervised Data Generation and Validity of the Underlying Hypothesis: The generation of unseen target data is unsupervised and the plausibility of the target data cannot be directly imposed while training the generative model. Therefore, it is required to evaluate how realistic the target data generated by *FaultSignatureGAN* is; to which extent it can be used as a surrogate of real target data. The data visualization (see Section 6.3) shows that the generated data represents real target data well. In particular, it represents the target fault data substantially better compared to the source data. This finding is also supported by the findings in the DA experiments (both *Partial*

and *Open-Partial*) where *FaultSignatureGAN* consistently outperforms the *Baseline* method. This supports the validity of the underlying hypothesis to enable controlled generation of previously unobserved data. We can draw the following conclusions: (1) Eq. (1) serves as a good approximation of real fault data, (2) the disentanglement of domain-specific and fault-specific characteristics was successful and (3) that domain-invariant fault signatures can be extracted by *FaultSignatureGAN* given only one source domain. However, our assumption about the structure of domain-specific and fault-specific components composing real fault data as defined in Eq. (1) could be extended and further refined in future work, in particular, the assumption that the OCs impact the fault-specific components linearly. Moreover, the DA experiments show that the generative process succeeds in preserving the semantic meaning of the generated data. If this would not be the case, the generated data would introduce label noise to the training data and, quite likely, result in a performance drop in the target domain.

Synthetic Data for Hyperparameter Tuning: Our results have shown that the classification performance of the *Baseline* method in the target domain is highly dependent on the chosen classifier architecture (see evaluation in Section 5). This is also observed in the literature, where different *Baseline* results on exact same tasks are reported with different hyperparameters of classifier architectures. For example, Wang et al. [10] reported a mean performance 99.78% on the domain shift $2 \rightarrow 3$, whereas [37] reported a baseline performance of 92.2% using a different classifier architecture. The classification improvements in the target domain gained by an appropriate choice of hyperparameters is even larger compared to improvements gained by other DA methods. This emphasizes the importance of hyperparameter tuning including the choice of the network architecture for the task of DA with extreme label space discrepancies. Previous publications, therefore, rely, for example, on the availability of target data and labels for one domain shift experiment. This availability of any target faults is not realistic in real safety-critical applications, where faults did not occur. In absence of fault target data, there is no possibility to tune these hyperparameters with respect to the classification task in the target domain, which can pose a major risk in safety-critical assets. If, however, synthetic data is available that represents the real target data well, the data can be used for validation. In this study, we showed that synthetic fault data generated by *FaultSignatureGAN* can support selecting the optimal architecture without relying on real target fault data that is usually not available in real safety-critical applications. Herein lays one major benefit of the proposed data generative approach *FaultSignatureGAN*. Although a proof of optimality is impossible (as the real target data has not been observed), the synthetic data provides a better indication of which hyperparameters to choose compared to the hyperparameter choice based on the source dataset performance or even a random choice.

Decreasing Data Acquisition Time: In practice, a short data acquisition phase is essential to enable to start monitoring the condition of a new asset within a short period of time. However, faults are extremely rare in complex (safety-critical) systems. This lack of real fault data is a major limitation to applying data-driven solutions for fault diagnostics. *FaultSignatureGAN* allows to transfer fault patterns to a new target domain. Once a fault occurred in one domain providing sufficient fault data to train a generative model, the fault signature can be learned, which then can be used to generate new fault data for any newly emerging domain. This ultimately can speed up the data collection process significantly, enabling the application of data-driven solutions within a shorter time span.

8. Conclusion

In this research, we proposed the *FaultSignatureGAN* framework for controlled generation of unseen faults in the target domain. The resulting generated fault data is (1) specific to a desired domain and

(2) specific to a certain fault type and the severity level of the fault in that domain. Therefore, *FaultSignatureGAN* enables to start monitoring the condition of new assets without any faults observed in the target domain since plausible faulty data can be generated for all future target domains. While we considered different OCs as domains in this research, the proposed framework is also applicable to generate synthetic faults in new units of a fleet.

We demonstrated the potential of the *FaultSignatureGAN* to complement partial label spaces in different DA experiments — *Partial* as well as *Open-Partial* DA settings. The results show that the generated data represents true faults in the target domain considerably better than the source fault data, leading to an improved classification performance on the target domain. Our proposed method excels particularly on large domain gaps. *FaultSignatureGAN* also enabled hyperparameter tuning for unseen target domain which can be applied in combination with any other DA approach. Without any access to target faults, tuning existing methods optimally is not possible. This demonstrates one of the benefits of plausible data generation in the evaluated tasks.

For future work, an additional step integrating real but unlabeled target data in addition to the synthetically generated data is an interesting direction to explore. Additional unsupervised or semi-supervised DA approaches could be employed to bridge the synthetic to real gap. Furthermore, the transferability of the generated fault signatures between different bearing types is of high interest for future research. One further direction of future research would be to investigate the source data demand for *FaultSignatureGAN*, evaluating how many samples and how diverse they need to be in order to train a representative generative model. On a bigger scale, the integration of novel or evolving fault detection (those that have not been observed neither in the source nor in the target domain) in addition to the performed fault classification would be of a significant practical relevance.

CRedit authorship contribution statement

Katharina Rombach: Writing – original draft, Visualization, Validation, Software, Methodology, Conceptualization. **Gabriel Michau:** Writing – original draft, Supervision, Methodology, Conceptualization. **Olga Fink:** Writing – original draft, Supervision, Methodology, Funding acquisition, Conceptualization.

Data availability

Publicly available benchmark datasets have been used.

Acknowledgment

This research resulted from the “Integrated intelligent railway wheel condition prediction” (INTERACT) project, supported by the ETH Mobility Initiative.

Appendix. Models

Unless stated otherwise, the following model architectures were used:

Generation Model: The first layer of the generation model is a single neuron. The activation of this neuron is sampled from a categorical distribution corresponding to the number of fault classes (fault type severities). The second fully connected layer is the sampling layer (mean and variance), containing three units each, activated by LeakyReLU ($\alpha=0.001$).

The following fully connected layers successively increase the dimensionality to the desired final output shape. Each layer is activated by LeakyReLU ($\alpha=0.001$), using no bias and followed by a BatchNormalization (BN) layer.

Three 1D convolutional layers follow, each layer is activated by LeakyReLU ($\alpha=0.001$), and followed by a BatchNormalization (BN) layer.

At last the generated signal is added to a randomly drawn data point from the base dataset.

The Adam optimizer used with a learning rate of 0.0001, $\beta_{a_1} = 0.5$ and $\beta_{a_2} = 0.999$.

Triplet Encoder Model: The triplet encoder model consists of 6 fully connected layers, each activated with Leaky ReLu ($\alpha=0.1$) and followed by a dropout layer ($\text{rate}=0.4$). The final layer is 4 dimensional and is L2 normalized.

The Adam optimizer used with a learning rate of 0.0001, $\beta_{a_1} = 0.5$ and $\beta_{a_2} = 0.999$.

Discriminator Model: The discriminator model consists of six fully connected layers, each activated with Leaky ReLu ($\alpha=0.1$) and followed by a dropout layer ($\text{rate}=0.1$). The final layer is 1-dimensional. The Adam optimizer used with a learning rate of 0.0001, $\beta_{a_1} = 0.5$ and $\beta_{a_2} = 0.999$.

Classification Model for Early Stopping: It consists of 4 1-D convolutions layers (8 filters in each layer and kernel size is 3), each activated with Leaky ReLu ($\alpha=0.1$) and followed by a dropout layer ($\text{rate}=0.1$). Followed by a flattening layer and a fully connected layer with the appropriate number of units according to the number of classes in the dataset.

The Adam optimizer used with default parameters.

Classification Model for Evaluation: The classification model for evaluation is inspired by Wang et al. [10]. It consists of three 1D convolutional layers (10 filters in each layer, activated by ReLu) and dropout layers (0.4). The Adam optimizer used with default parameters.

For training the CWRU classification models a batch size of 64 is chosen, for Paderborn 2000.

Model Architecture for the comparison method BA3US: The comparison method BA3US has not yet been applied to any time-series data. Without using target fault data, the methodology could not be tuned to give satisfying results. Therefore, we followed the procedure of Wang et al. [10] and tuned BA3US on a validation task 0→3. We started using the exact same generator, discriminator and classifier architecture as well as optimizer setting as proposed by Wang et al. [10]. All hyperparameters (model architecture and weighting of the different loss terms) are then consecutively optimized on the validation task. Ultimately, the following architecture was used: The **feature extractor** consists of a 3 layer 1D-convolutional layer with kernel size 3 and 10 filters per layer. Each layer is batch normalized and activated by the sigmoid function, followed by a dropout layer ($\text{rate}=0.5$). Last, based on the flattened activations, a fully connected layer is added with 256 units. The **classifier** model consists of two fully connected layers. The first with 256 units is activated with the ReLU activation function and followed by a dropout layer ($\text{rate}=0.5$). The second contains 10 units (corresponding to the number of classes) and is activated by the softmax function. The **discriminator** contains three fully connected layer, each with 256 ReLU activated units. Only the last layer contains only one unit and is Sigmoid activated. The model is optimized on batches of 64 samples in the target and the source domain using the StochasticGradientDescent algorithm with a learning rate of 0.005. The initial ratio of augmented source samples is set to 1.0 ($\rho_0 = 1$), the test interval N_u is set to 50. The loss conditional entropy loss is weighted with a factor of 10^{-3} and the transfer loss with a factor of 10^{-1} . The weighted complement entropy loss is not considered since it did not lead to satisfying results ($w = 0$).

References

- [1] Fink O, et al. Potential, challenges and future directions for deep learning in prognostics and health management applications. *Eng Appl Artif Intell* 2020;92:103678.
- [2] Rombach K, Michau G, Fink O. Contrastive learning for fault detection and diagnostics in the context of changing operating conditions and novel fault types. *Sensors* 2021;21(10):3550.
- [3] Michau G, Fink O. Domain adaptation for one-class classification: Monitoring the health of critical systems under limited information. *Int J Prognost Health Manag* 2019;10(4).
- [4] Deng M, Deng A, Shi Y, Liu Y, Xu M. A novel sub-label learning mechanism for enhanced cross-domain fault diagnosis of rotating machinery. *Reliab Eng Syst Saf* 2022;108589.
- [5] Lee J, et al. Asymmetric inter-intra domain alignments (AIDA) method for intelligent fault diagnosis of rotating machinery. *Reliab Eng Syst Saf* 2022;218:108186.
- [6] Liang J, Wang Y, Hu D, He R, Feng J. A balanced and uncertainty-aware approach for partial domain adaptation. In: *European conference on computer vision*. Springer; 2020, p. 123–40.
- [7] Li G, Kang G, Zhu Y, Wei Y, Yang Y. Domain consensus clustering for universal domain adaptation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, p. 9757–66.
- [8] Saito K, Saenko K. Ovanet: One-vs-all network for universal domain adaptation. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, p. 9000–9.
- [9] Michau G, Fink O. Unsupervised transfer learning for anomaly detection: Application to complementary operating condition transfer. *Knowl-Based Syst* 2021;216:106816.
- [10] Wang Q, Michau G, Fink O. Missing-class-robust domain adaptation by unilateral alignment. *IEEE Trans Ind Electron* 2020;68(1):663–71.
- [11] Zhang W, Li X, Ma H, Luo Z, Li X. Open-set domain adaptation in machinery fault diagnostics using instance-level weighted adversarial learning. *IEEE Trans Ind Inf* 2021;17(11):7445–55.
- [12] Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville A. Improved training of wasserstein gans. 2017, arXiv preprint arXiv:1704.00028.
- [13] Li W, et al. A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: Theories, applications and challenges. *Mech Syst Signal Process* 2022;167:108487.
- [14] Zhao C, Shen W. Dual adversarial network for cross-domain open set fault diagnosis. *Reliab Eng Syst Saf* 2022;221:108358.
- [15] Zhang W, Li X, Ma H, Luo Z, Li X. Universal domain adaptation in fault diagnostics with hybrid weighted deep adversarial learning. *IEEE Trans Ind Inf* 2021;17(12):7957–67.
- [16] Li X, Zhang W, Ma H, Luo Z, Li X. Partial transfer learning in machinery cross-domain fault diagnostics using class-weighted adversarial networks. *Neural Netw* 2020;129:313–22.
- [17] Jiao J, Zhao M, Lin J, Ding C. Classifier inconsistency-based domain adaptation network for partial transfer intelligent diagnosis. *IEEE Trans Ind Inf* 2019;16(9):5965–74.
- [18] Zhao C, Liu G, Shen W. A balanced and weighted alignment network for partial transfer fault diagnosis. *ISA Trans* 2022.
- [19] Li X, Zhang W. Deep learning-based partial domain adaptation method on intelligent machinery fault diagnostics. *IEEE Trans Ind Electron* 2020;68(5):4351–61.
- [20] Li X, Zhang W, Ding Q. Cross-domain fault diagnosis of rolling element bearings using deep generative neural networks. *IEEE Trans Ind Electron* 2018;66(7):5525–34.
- [21] Zhou T, Han T, Droguett EL. Towards trustworthy machine fault diagnosis: A probabilistic Bayesian deep learning framework. *Reliab Eng Syst Saf* 2022;224:108525.
- [22] Zhao C, Shen W. Adaptive open set domain generalization network: Learning to diagnose unknown faults under unknown working conditions. *Reliab Eng Syst Saf* 2022;108672.
- [23] Gui J, Sun Z, Wen Y, Tao D, Ye J. A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE Trans Knowl Data Eng* 2021.
- [24] Wilson G, Cook DJ. A survey of unsupervised deep domain adaptation. *ACM Trans Intell Syst Technol (TIST)* 2020;11(5):1–46.
- [25] Luo J, Huang J, Li H. A case study of conditional deep convolutional generative adversarial networks in machine fault diagnosis. *J Intell Manuf* 2021;32(2):407–25.
- [26] Wang Q, Taal C, Fink O. Integrating expert knowledge with domain adaptation for unsupervised fault diagnosis. *IEEE Trans Instrum Meas* 2021.
- [27] Locatello F, et al. Challenging common assumptions in the unsupervised learning of disentangled representations. In: *International conference on machine learning*. PMLR; 2019, p. 4114–24.
- [28] Schott L, et al. Visual representation learning does not generalize strongly within the same domain. In: *Proc. International Conference on Learning Representations*. 2021.

- [29] Yang Y, Soatto S. Fda: Fourier domain adaptation for semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 4085–95.
- [30] Cooley JW, Tukey JW. An algorithm for the machine calculation of complex Fourier series. *Math Comp* 1965;19(90):297–301.
- [31] Kingma DP, Welling M. Auto-encoding variational bayes. In: International conference on learning representations. 2014.
- [32] Schroff F, Kalenichenko D, Philbin J. Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015, p. 815–23.
- [33] Smith WA, Randall RB. Rolling element bearing diagnostics using the case western reserve university data: A benchmark study. *Mech Syst Signal Process* 2015;64:100–31.
- [34] Lessmeier C, Kimotho JK, Zimmer D, Sextro W. Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification. In: PHM society european conference, Vol. 3. (1). 2016.
- [35] Bousmalis K, Silberman N, Dohan D, Erhan D, Krishnan D. Unsupervised pixel-level domain adaptation with generative adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 3722–31.
- [36] Saxe AM, et al. On the information bottleneck theory of deep learning. *J Stat Mech Theory Exp* 2019;2019(12):124020.
- [37] Li X, Zhang W, Ma H, Luo Z, Li X. Deep learning-based adversarial multi-classifier optimization for cross-domain machinery fault diagnostics. *J Manuf Syst* 2020;55:334–47.