Renewable Energy 197 (2022) 331–341

Contents lists available at ScienceDirect

Renewable Energy

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

A deep boosted transfer learning method for wind turbine gearbox fault detection

Faras Jamil [a,b,*], Timothy Verstraeten [a,b], Ann Nowé [b], Cédric Peeters [a], Jan Helsen [a]

[a] Acoustics & Vibration Research Group/OWI-Lab, Vrije Universiteit Brussel, Belgium
[b] Artificial Intelligence Lab, Vrije Universiteit Brussel, Belgium

ARTICLE INFO

Keywords:
Fault detection
Deep transfer learning
Vibrations
Gearbox

ABSTRACT

Deep learning methods have become popular among researchers in the field of fault detection. However, their performance depends on the availability of big datasets. To overcome this problem researchers started applying transfer learning to achieve good performance from small available datasets, by leveraging multiple prediction models over similar machines and working conditions. However, the influence of negative transfer limits their application. Negative transfer among prediction models increases when the environment and working conditions are changing continuously. To overcome the effect of negative transfer, we propose a novel deep transfer learning method, coined deep boosted transfer learning, for wind turbine gearbox fault detection that prevents negative transfer and only focuses on relevant information from the source machine. The proposed method is an instance-based deep transfer learning method that updates the weights of the source and the target machine training samples separately. The weights of different source training samples are gradually decreased to reduce the impact on the final model. The proposed method is verified by the Case Western Reserve University bearing and real field wind farm datasets. The results show that the proposed method ignores negative transfer and achieves higher accuracy compared to standard deep learning and deep transfer learning methods.

1. Introduction

In the past two decades the impact of global warming and climate change has significantly influenced society and popularized the belief that a transition to renewables as an alternative energy source to fossil fuels is a must rather than a choice. As a consequence of this belief and thanks to technological advances, renewable energy production is growing increasingly fast with it being predicted to not slow down anytime soon. Wind energy is one of these booming technologies, having a 53% year-over-year growth in the global wind industry in 2020 despite the disrupted supply chain due to the global pandemic [1]. Apart from the fastest growth, renewable energy sources produce 15% of world total energy consumption. To ensure sustained growth, the production of renewable energy must be lucrative and reliable. Europe has a significant interest in offshore wind farms because the ideal on-shore sites are already populated [2]. It shares 75% of the entire world offshore installation [3]. The remote and challenging location of offshore wind farms increases the energy cost due to the extra logistic cost for installation and maintenance [4], which is expected to account for 30% of the total cost [5]. Moreover, they also face more environmental resistance from strong wind and waves [6,7] and thus require more maintenance. Therefore, the maintenance cost should be reduced to make offshore wind farms more expedient [8]. Cost can be reduced by early fault detection based on available information and then planning maintenance [9].

The recent advancements in deep learning (DL) achieve enormous success in many fields [10], ranging from computer vision [11,12] to speech recognition [13]. Therefore, it is also attracting researchers in the field of fault detection [14–17]. Compared to other fields however, fault detection faces a data imbalance problem where most available data are considered healthy cases with only a very small minority being data of fault cases [18]. In this study, a method was developed to build fault detection models over an offshore wind farm by sharing the learned knowledge among the wind turbines (WTs). Even though the WTs are practically identical, their behavior differs from each other due to different environments and working conditions (WCs) [19,20]. The recent progress in the Internet of Things (IoT) allows collecting data from remote sources. The offshore wind farm operators can collect different types of data from multiple sensors on higher frequencies.

* Correspondence to: Vrije University Brussel, Pleinlaan 2, 1050 Brussel, Belgium.
E-mail addresses: faras.jamil@vub.be (F. Jamil), timothy.verstraeten@vub.be (T. Verstraeten), ann.nowe@vub.be (A. Nowé), cedric.peeters@vub.be (C. Peeters), jan.helsen@vub.be (J. Helsen).

https://doi.org/10.1016/j.renene.2022.07.117
Received 10 December 2021; Received in revised form 10 May 2022; Accepted 25 July 2022
Available online 28 July 2022
0960-1481/

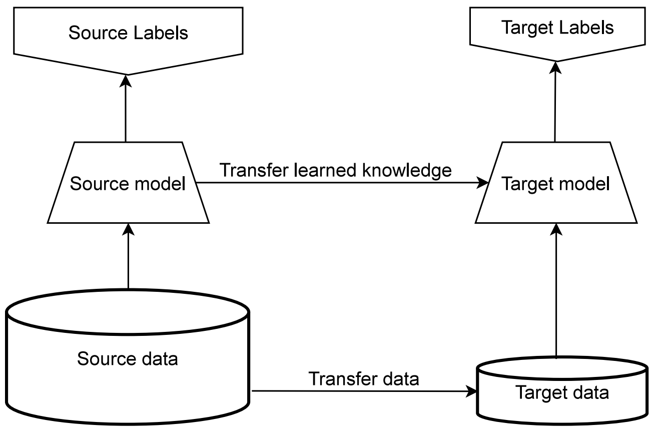


Fig. 1. Transfer Learning allows transferring knowledge on model or data level to improve the learning of target task.

This research is focused on transfer learning (TL) that allow combining data over multiple machines. To deal with large datasets and capture the non-linear trends from different sensors, we need a method like deep learning instead of using shallow machine learning models. TL with DL is employed to transfer learned knowledge from extensive fault history WT to the scarce fault history WT, which is not sufficient to train traditional machine learning models. [20].

1.1. Transfer learning

The goal of TL is to improve the performance of learners by transferring knowledge from another related domain whereas, the traditional machine learning models are based on the assumption that both the training and testing data belong to the same data distribution. The idea of TL is inspired by human learning behavior, where humans use previous related knowledge while learning to solve new problems. For example, a person learns to drive a car quickly if he knows how to ride a bike as compared to learning from scratch without any road experience. As shown in Fig. 1, TL enables machine learning models to transfer learned knowledge from source domains to a target domain to improve the performance of the target learning function, while both the source and target domain have different data distributions [21]. Moreover, the source domain data samples can be transferred to improve the learning of the target model [22].

TL is getting popular, and it has been applied in many fields including fault detection, but there are still some research gaps that need to be filled out. We will only discuss deep transfer learning (DTL) fault detection methods because this research is focused on DTL methods. First, we will give an overview of already available DTL fault detection methods, and then we will explain our proposed method to fill the research gaps. DTL fault detection methods transfer the learned knowledge about faults from a source machine to a target machine. The currently available methods only use one machine as a source. Lu et al. [23] propose domain adaption in fault diagnosis to learn features combined from the source and the target domain, and then a support vector machine classifier is used to predict faults. A network-based DTL method was used to predict bearing inner race, ball, and outer race faults in changing WCs [24]. Li et al. [25] proposed an improved deep neural network (DNN) optimized by a particle swarm optimization algorithm [26] and L2 regularization method to classify gear pitting faults. Data labeling is a difficult task, and fault data is imbalanced, however, it is also possible to transfer the knowledge to a target without the labeled data. The maximum mean discrepancy was used to minimize the difference between the source and the target domains when the labeled data was not available for the target. Along with the domain adaptation using maximum mean discrepancy, different DL models are used for condition recognition like sparse autoencoder [27] or convolutional neural network (CNN) [28,29]. Yang et al. [30] propose a feature-based CNN to extract transferable features from the raw vibration signals. Then multi-layer domain adaptation is added to reduce the distribution discrepancy of learned transferable features, and pseudo label learning [31] is used to train from unlabeled target domain samples. ImageNet [11] pre-trained network is used to train a deep learning network for fault classification [32]. The sensors data is converted to image data by plotting [33] or using wavelet transformation [34] to obtain a time-frequency distribution to fine-tune high-level network layers, and the low-level features are extracted from the pre-trained Imagenet network. The research using the DTL fault detection methods was validated on lab-generated data like the Case Western Reserve University(CWRU) bearing dataset [35].

The current state-of-the-art methods do not focus on the aspect that there are some dissimilarities between the source and the target machines. The discrepancies referred to as negative transfer may hinder the performance of the target model. In contrast with the negative transfer, the valuable information from the source is positive transfer. Therefore, we can improve the performance of the target machine DTL fault detection model by only allowing positive transfer while ignoring the negative transfer [36]. The negative transfer is a topic of interest in the TL research community, but it is not addressed in fault detection literature. It is possible to avoid the negative transfer by identifying the relevant properties in the source machine.

The effect of negative transfer limits the applications of TL methods where the difference between the source and the target domain is substantial. The influence of negative transfer increases with the amount of data, i.e., when the source domain dataset becomes larger, the performance of the target model decreases. Due to this limitation, the current state-of-the-art methods only perform better when the number of samples in the target domain is somewhat equal to the number of samples in the source domain. The state-of-the-art TL fault detection methods use 50% [28,30] and 37% [24] target domain training data corresponding to the source domain data. The research of Zhang et al. [24] also discusses the impact of negative transfer and demonstrate that substantial differences between the source and the target machines deteriorate the performance of the DTL model.

It is also possible to combine DL with TL to train non-linear high-dimensional DTL models with a small dataset. DTL is classified into four categories: instance-based DTL, network-based DTL, mapping-based DTL, and adversarial-based DTL [22]. In order to address the negative transfer problem, we propose, an instance-based DTL method.

1.1.1. Instance-based deep transfer learning

Instance-based DTL is a weight adjustment DTL approach for both the source domain and the target domain instances. It is also possible to drop instances with negative influence instead of adjusting their weights. An instance-based DTL method identifies the impact of individual data samples on the model's performance and adjusts their weights accordingly. The weight adjustment of instances helps to avoid negative transfer in the DTL model. It is different from the most common network-based DTL in which only learned weights from the source model are transferred to the target model, and then the model is fine-tuned on target domain data samples. A modified version of network-based DTL is used to identify and remove the instances that contribute to the lower performance of the model [37,38]. However, the source domain introduces negative transfer and dropping the target domain instances can result in model overfitting. The proposed method transfer the learned knowledge while ignoring the negative transfer. Deep boosted transfer learning (DBTL) assigns weights to the instances of both the source and the target domains based on their relevance. The most related source domain data samples that improve the target model get high weights, and the least relevant data samples get low weights to avoid the negative transfer. In contrast, the weights of the target domain samples that were wrongly predicted are increased to

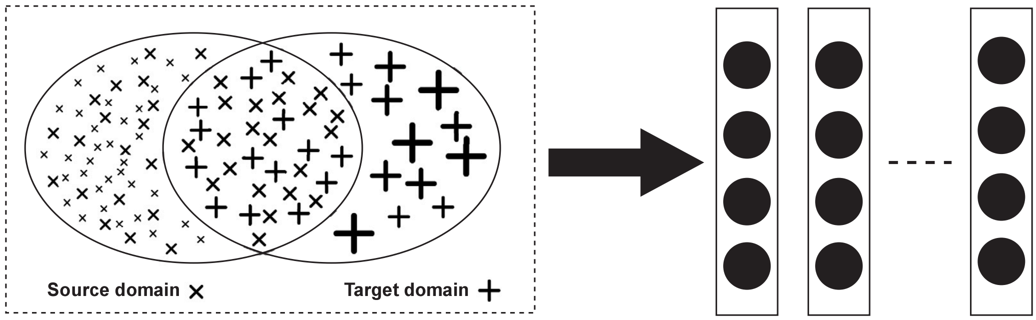


Fig. 2. Sketch map of instance-based DTL proposed method. The cross (x) and plus (+) signs respectively represent the source and the target domain instances. The size of instances corresponds to their weights that contributes toward the training of the deep learning model.

allow the model to learn new features different from the source domain. The sketch map of the instance-based DTL proposed method is shown in Fig. 2.

The contributions of our research are summarized below.

1. We propose a novel method that significantly prevents negative transfer in DL models when transferring fault detection knowledge from the source machine to the target machine.
2. The prediction accuracy of the proposed algorithm improves when the source domain data samples are increased whilst the target domain data is limited, it allows using all the available source domain data.
3. The proposed method shows improvements even in transferring fault detection knowledge between more dissimilar machines, where a DTL method shows deterioration as compared to the simple DL method.

2. Proposed transfer learning fault detection method

We need data from two different but related domains to train a DL model using TL. A source domain that has more data samples as compared to the target domain and the knowledge is transferred from the source domain to the target domain. Our proposed method is based on DL, and we use Instance-Based DTL to allow positive transfer while ignoring negative transfer. We propose deep boosted transfer learning (DBTL) that assigns weights to the source and the target domain instances. The DL model is required to train on the same dataset for multiple epochs. DBTL transfers knowledge from the source to the target domain by updating the weights of the same dataset's samples between each epoch. It uses the TrAdaBoost [39] weight updating mechanism to update the weights of the source and target domain data samples. DBTL has a low computational complexity compared to TrAdaBoost and allows for multi-class classification. It combines the source and the target domain data samples to train a DL model. Specifically, T_s and T_t are the training data that represent the source and target domains respectively. The instance space is $X = X_s(1,\dots,m) \cup X_t(1,\dots,n)$, and Y is a set of labels, m and n are the sizes of T_s and T_t respectively. After each batch update of the neural network, which we call an *epoch*, if a source training sample is predicted wrong, it is the most dissimilar from the target domain and it can cause negative transfer. Therefore, to reduce its impact in the subsequent epochs, its weight is decreased by applying a fixed multiplier $\beta = 1/(1+\sqrt{2\log n/E})$ where E is the number of epochs [39]:

$$w_i = w_i \beta^{|h(x_i)-y_i|} \tag{1}$$

where $h(x_i)$ is the predicted label and y_i is original label. In contrast, the weights of the target domain samples that were wrongly predicted are increased according to the weight updating mechanism of AdaBoost [40]:

$$w_i = w_i \beta_e^{-|h(x_i)-y_i|} \tag{2}$$

The β_e is calculated based on the error ϵ_e calculated during each epoch:

$$\beta_e = \frac{\epsilon_e}{1-\epsilon_e} \tag{3}$$

where ϵ_e is calculated as follows:

$$\epsilon_e = \Sigma_{i=n+1}^{n+m} \frac{w_i^e \cdot |h_e(x_i) - y_i|}{\Sigma_{i=n+1}^{n+m} w_i^e} \tag{4}$$

Algorithm 1: DBTL

Input: Source $T_s(1,\dots,n)$ and Target $T_t(1,\dots,m)$ training dataset.
Maximum number of epochs E.
DNN Learning algorithm.
Initialize: Initial weight vector $w^1 = (w_1^1,\dots,w_n^1,w_{n+1}^1,\dots,w_{n+m}^1)$

$$w_i^1 = \frac{1}{n}, \qquad 1 \le i \le n$$
$$w_i^1 = \frac{1}{m}, \qquad n+1 \le i \le n+m$$
$$\beta = \frac{1}{(1+\sqrt{2\log n/E})}$$

for $e = 1, \dots, E$ **do**

$p^e = \frac{w^e}{\Sigma_{i=1}^{n+m} w_i^e}$; `/* Weights normalization */`

Train the DNN on T_s and T_t with normalized weight distribution p^e and get the prediction on $X = (T_s, T_t)$ $h_e : X \to Y$

Calculate the error of h_e on T_t: $\epsilon_e = \Sigma_{i=1}^{n} \frac{w_i^e \cdot |h_e(x_i) - y_i|}{\Sigma_{i=1}^{n} w_i^e}$;

`/* where` $h_e(x_i)$ `is predicted label` y_i `is original class label */`

$\beta_e = \frac{\epsilon_e}{1-\epsilon_e}$

Update weights:

$w_i^{e+1} = w_i^e \beta^{|h_e(x_i)-y_i|}, \qquad 1 \le i \le m$

$w_i^{e+1} = w_i^e \beta_e^{-|h_e(x_i)-y_i|}, \qquad m+1 \le i \le m+n$

end

A DNN trained on the source and the target domain.

A formal description of the DBTL is given in algorithm 1. A combined training dataset $T \subseteq \{X, Y\}$ is used to train a DL model that predicts new unseen samples from the target domain. The training data T along with the number of epochs and a DL model is passed as an input to the algorithm. It requires initial weights to start training, and they can be initialized based on the problem. It is also possible to start with equal weights for both the source and the target domains training samples. However, we give high weights to the target domain training samples over the source domain training samples. Moreover, the sample weights can be initialized using prior knowledge about the source and the target domains. After initializing the weights, the following steps are repeated for each epoch:

Table 1
The DNN network detail structure with the configuration of each layer with respective output for the CWRU and real-world wind farm data.

Layer (type)	CWRU		Wind Farm	
	Configurations	Output	Configurations	Output
Input		[32, 1, 400]		[8, 1, 40]
Conv1d-1	ch:16, k:3	[32, 16, 398]	ch:16, k:3	[8, 16, 38]
BatchNorm1d-1	ch:16	[32, 16, 398]	ch:16	[8, 16, 38]
ReLU-1		[32, 16, 398]		[8, 16, 38]
MaxPool1d-1	k:8, s:4	[32, 16, 98]	k:4, s:2	[8, 16, 18]
Dropout-1	d:0.5	[32, 16, 98]	d:0.5	[8, 16, 18]
Conv1d-2	ch:32, k:3	[32, 32, 96]	ch:32, k:3	[8, 32, 16]
BatchNorm1d-2	ch:32	[32, 32, 96]	ch:32	[8, 32, 16]
ReLU-2		[32, 32, 96]		[8, 32, 16]
MaxPool1d-2	k:8, s:4	[32, 32, 23]	k:4, s:2	[8, 32, 7]
Dropout-2	d:0.5	[32, 32, 23]	d:0.5	[8, 32, 7]
Conv1d-3	ch:64, k:3	[32, 64, 21]	ch:64, k:3	[8, 64, 5]
BatchNorm1d-3	ch:64	[32, 64, 21]	ch:64	[8, 64, 5]
ReLU-3		[32, 64, 21]		[8, 64, 5]
MaxPool1d-3	k:4, s:2	[32, 64, 9]	k:2, s:1	[8, 64, 4]
Dropout-3	d:0.5	[32, 64, 9]	d:0.5	[8, 64, 4]
Conv1d-4	ch:128, k:3	[32, 128, 7]	ch:128, k:3	[8, 128, 2]
BatchNorm1d-4	ch:128	[32, 128, 7]	ch:128	[8, 128, 2]
ReLU-4		[32, 128, 7]		[8, 128, 2]
MaxPool1d-4	k:4, s:2	[32, 128, 2]	k:2, s:1	[8, 128, 1]
Dropout-4	d:0.5	[32, 128, 2]	d:0.5	[8, 128, 1]
Flattening		[32, 256]		[8, 128]
Linear-1	n:256	[32, 128]	n:128	[8, 64]
ReLU-5		[32, 128]		[8, 64]
Dropout-5	d:0.5	[32, 128]	d:0.5	[8, 64]
Linear-2	n:128	[32, 64]	n:64	[8, 32]
ReLU-6		[32, 64]		[8, 32]
Dropout-6	d:0.5	[32, 64]	d:0.5	[8, 32]
Linear-3	n:64	[32, 6]	n:32	[8, 2]
Output	n:6	[32, 6]	n:2	[8, 2]

- Normalize weights before passing to the DL model.
- Train the DL model on instance space T sampled from both the source and the target domains and then predict using the trained model to identify wrongly predicted instances.
- Calculate the epoch error ϵ_e as given by Eq. (4) from the predicted and actual labels of the target domain.
- Calculate β_e from ϵ_e as given in Eq. (3)
- Update the instance space T weights using Eqs. (1) and (2) for the source and the target domains training samples.

The output is a DL model trained from the source and the target domains that can predict new unseen samples from the target domain while ignoring negative transfer. Whilst DL works better when trained on large datasets, this research focuses on achieving good results from DL models by using a small dataset using TL. The implementation of DBTL is similar to a simple DL model, but it requires calculating the weights of the source and the target domain data samples after each epoch. The difference between the computation time of the DL and the DBTL model is negligible if trained on the same number of data samples. DBTL does not require a specific DL architecture, and it can be adopted by any DL architecture. We use a simple feedforward CNN based architecture. The training samples are 1-dimensional therefore, we use 1-dimensional convolutional (Conv1D) layers. The Conv1D-based DL models showed better results when compared with other DL architectures on an open-source benchmark study [41].

The DNN structure used to train on the CWRU dataset for DL, DTL, and DBTL is shown in Fig. 3. We used the same Conv1D based DNN structure for wind farm data but due to different input and output sizes, some parameters are different. The detailed DNN structure for both the CWRU bearing dataset and real-world wind farm data is provided in Table 1.

Table 2
DL Hyperparameters the batch size 32 and 08 was used for the CWRU and wind farm datasets.

Hyperparameters	Value
Epochs	100
Learning rate (LR)	0.001
LR decay	0.00001
Batch size	32 & 08

Table 3
CWRU bearing dataset source and target domains WCs for transfer learning. WC1 is the source domain and WC2, WC3, and WC4 are the target domains.

	WC1	WC2	WC3	WC4
Fault diameter (inches)	0.007	0.021	0.021	0.021
Motor speed (rpm)	1797	1797	1730	1730
Frequency (kHz)	48	48	48	12
Training samples	930/37226	375	375	375
Testing samples	–	38109	144855	36021

3. Experiments

The proposed method is validated on the Case Western Reserve University (CWRU) bearing dataset, which is a standard benchmark in the context of fault detection [35]. Furthermore, the method is also demonstrated on annotated failure data from 2 wind turbines in the field. To compare the performance of the proposed method and identify the effect of negative transfer we use a simple DL method and network-based DTL method. The network-based DTL method is inspired by the fault detection method in Zhang et al. [24], where trained weights of the source Deep Neural Network (DNN) model are passed to the target DNN model. The used DL method is the baseline traditional machine learning method without TL and DNN trained on target machine data without any prior knowledge. Furthermore, the state-of-the-art TL methods were validated on datasets with a limited number of source domain training samples because they were lacking the ability to avoid the negative transfer. To validate DBTL on the standard of the current state-of-the-art methods and demonstrate the true potential of the proposed method we performed two types of experiments with scarce and extensive source domain training samples respectively.

The DNN hyperparameters are provided in Table 2. The same hyperparameters are used for all DL, DTL, and DBTL experiments except the batch size. The batch size was 32 and 8 respectively for the CWRU bearing dataset and the real-world wind farm dataset experiments. Moreover, each experiment was performed for 10 different seed values, and the results are calculated based on the mean and standard deviation over 10 repeated experiments. The performance analysis of DL methods is based on model prediction accuracy on testing data and training loss [42].

4. CWRU bearing fault case study

4.1. CWRU experiment setup

The CWRU experiment equipment is shown in Fig. 4. A 2 hp Reliance Electric motor and SKF bearings were used in the experiments. To observe the faulty behavior in the bearings single point fault with 7 mils, 14 mils, and 21 mils (1 mil equals 0,001 inches) diameters were introduced using electro-discharge machining. Accelerometers attached on the 12 o'clock position at both the drive and fan end of the motor housing using magnetic bases were used to collect the vibration data. Table 3 depicts the detail of the CWRU dataset. We use WC1 as a source domain and WC2, WC3, and WC4 are used as target domains where the difference increases from WC2 to WC4. The source domain WC1 (fault diameter 0.007 inches, motor speed 1797 rpm, and the sample frequency 48 kHz) is distinct from WC2 based on the fault

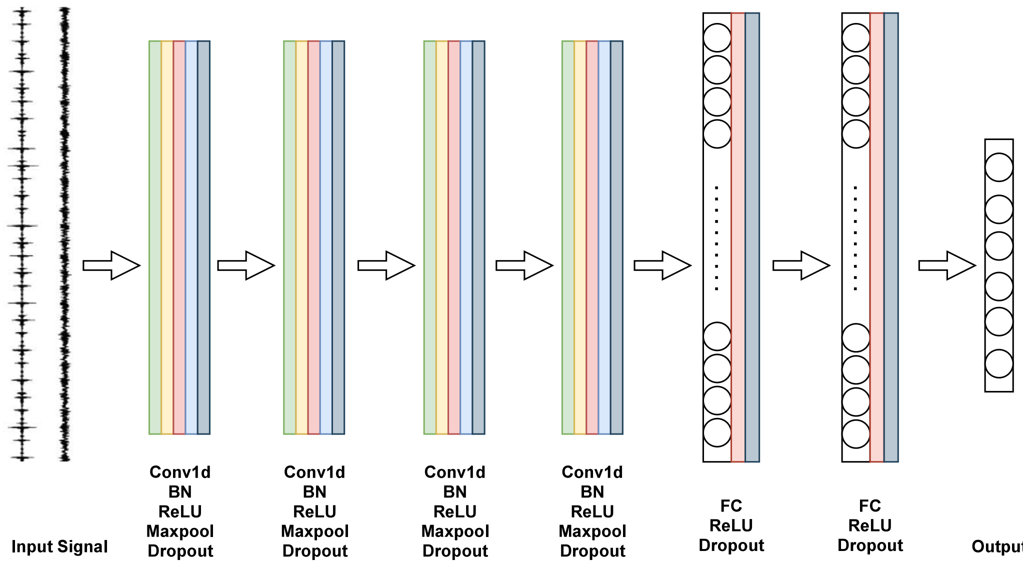


Fig. 3. DNN structure trained on the CWRU dataset for DL, DTL and DBTL, each Conv1D layer is followed by Batch normalization, relu activation function, max-pooling, and dropout(50%) respectively. Moreover, each linear layer is also followed by an activation function relu and dropout(50%).

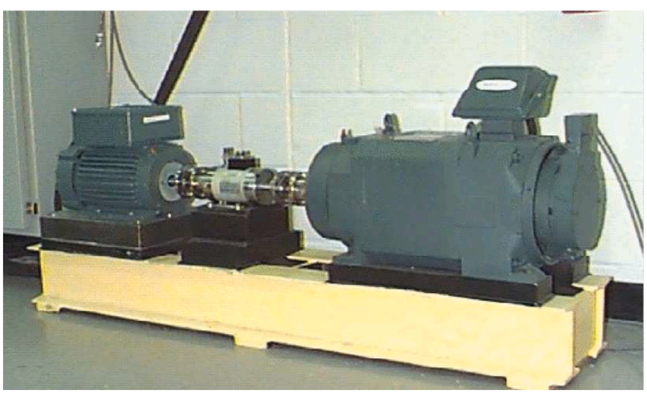


Fig. 4. CWRU experiment equipment [35].

Table 4
CWRU bearing dataset mean accuracy with the standard deviation of DL, DTL, and DBTL methods when knowledge is transferred from a scarce source domain. The improvement of DTL and DBTL from the base case DL method is given in parentheses of the respective cell.

Model	WC1 to WC2	WC1 to WC3	WC1 to WC4
DL	84.55 ± 0.81	82.35 ± 1.02	89.45 ± 1.03
DTL	87.66 ± 1.12 (3.11)	85.24 ± 0.96 (2.88)	90.29 ± 1.06 (0.85)
DBTL	89.71 ± 1.49 (5.16)	87.32 ± 1.88 (4.96)	94.48 ± 1.20 (5.04)

Table 5
CWRU bearing dataset mean accuracy with the standard deviation of DL, DTL, and DBTL methods when knowledge is transferred from an extensive source domain. The improvement and deterioration of DTL and DBTL from the base case DL method is given in parentheses in respective cells.

Model	WC1 to WC2	WC1 to WC3	WC1 to WC4
DL	84.55 ± 0.81	82.35 ± 1.02	89.45 ± 1.03
DTL	91.56 ± 0.44 (7.01)	83.12 ± 0.98 (0.77)	88.23 ± 1.06 (−1.22)
DBTL	96.64 ± 0.38 (12.09)	92.70 ± 0.43 (10.34)	99.24 ± 0.09 (9.79)

diameter (0.021 inches). WC3 is different based on motor speed (1730 rpm) including fault diameter (0.021 inches), and WC4 is the same as WC3 with a change in sample collection rate (sample frequency 12 kHz). The size of the training dataset is increased using the data augmentation technique as it helps DL models to avoid overfitting [43]. A sliding window of length 400 with a step size of 20 [24] is used to create multiple overlapping training samples from the available raw signal as shown in Fig. 5. Two different experiments are performed. The first experiment uses 930 training samples of the source domain, while the second experiment uses all (37226) samples. However, the target domain training samples are fixed at 375 for all experiments to assess the impact of the size of source domain datasets.

4.2. CWRU bearing dataset results

Two experiments are performed using scarce and extensive source domain samples on CWRU bearing dataset. The target domains (WC2, WC3, WC4) variations increase with the source domain (WC1) from WC2 to WC4, which increases the potential of having negative transfer.

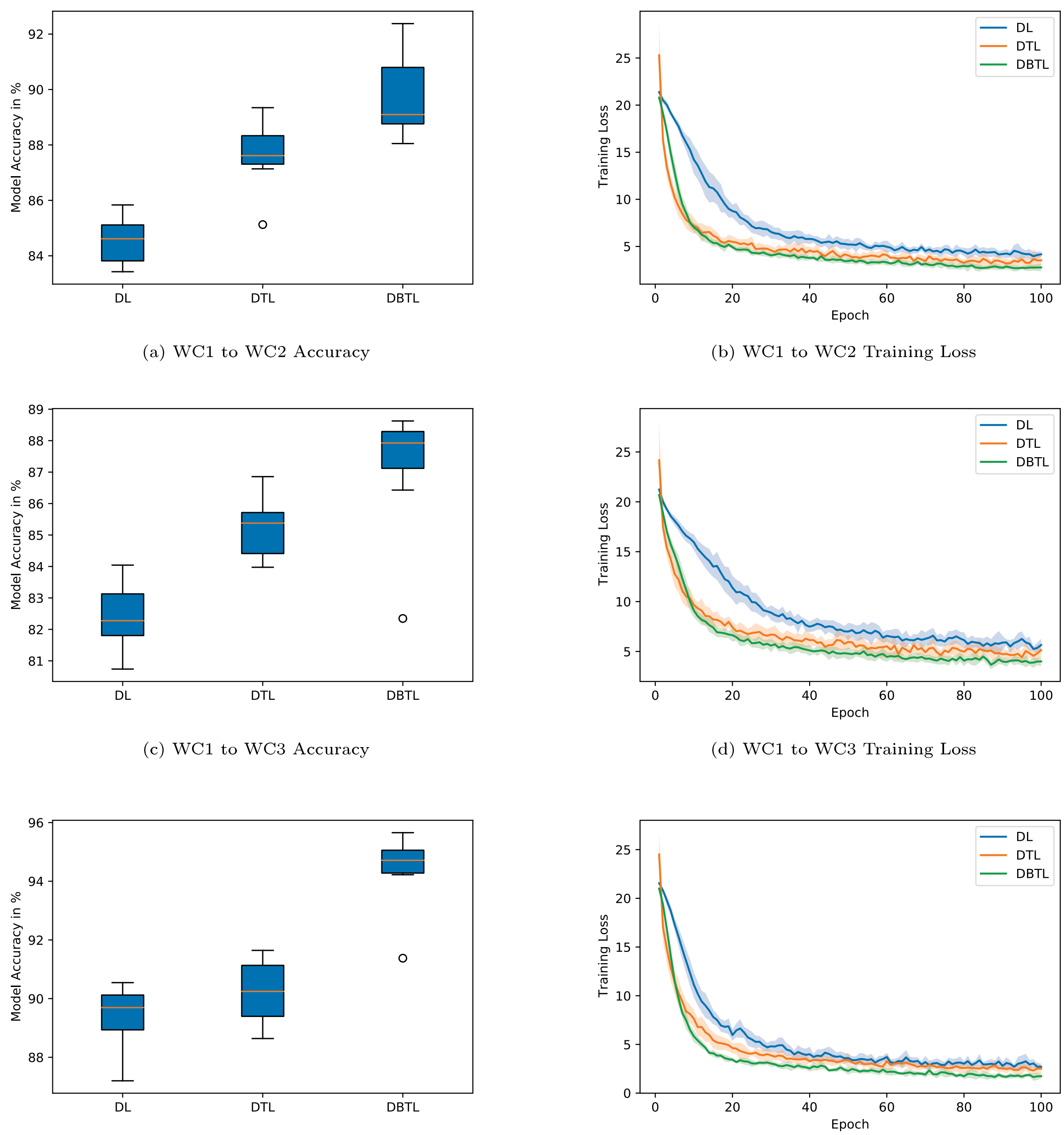
4.2.1. Transferring from scarce source samples

The results of all three methods are shown in Fig. 6 when the knowledge is transferred from scarce source samples. The detailed results for all methods are provided in Table 4. The mean and standard deviation are calculated over 10 repeated experiments, and the improvement is observed in DTL and DBTL from the base case DL. The DTL method observes notable deterioration when the source domain is significantly different from the target domain and shows better results when there are more similarities between the source and the target domains. WC2, which is most similar to WC1, observes a 3.11% improvement from the base case DL, whereas WC3 and WC4 which are less similar, observe a 2.88% and 0.85% improvement with respect to the base case, respectively. However, the DBTL shows more consistent results by reducing the negative transfer from the source domain. The improvement from the base case DL method is significant when comparing with the DTL method. The improvement from the base case DL method for WC2, WC3, and WC4 is 5.16%, 4.96%, and 5.04% respectively.

4.2.2. Transferring from extensive source samples

The following results help us understand the effect of negative transfer and potential of DBTL. The results when transferring from the extensive source samples are shown in Fig. 7, and the detailed results are provided in Table 5. The DTL method improved from 3.11% to 7.01% when the source domain training samples are increased because the target domain (WC2) is most similar to the source domain. However, it observes the deterioration due to negative transfer in the

Fig. 5. Data augmentation over raw vibration signal with frame length 400 and step size 20.



(a) WC1 to WC2 Accuracy

(b) WC1 to WC2 Training Loss

(c) WC1 to WC3 Accuracy

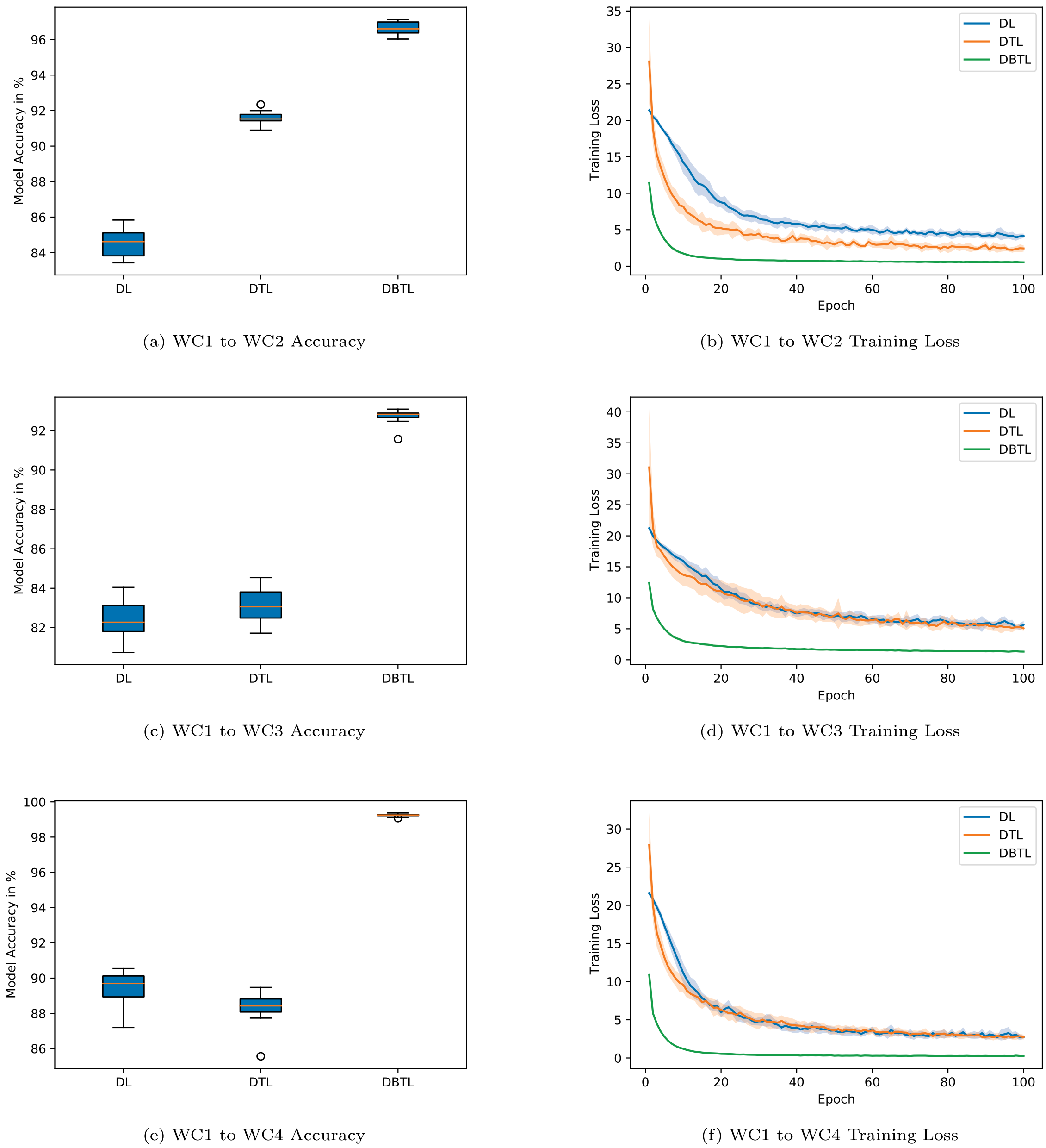
(d) WC1 to WC3 Training Loss

(e) WC1 to WC4 Accuracy

(f) WC1 to WC4 Training Loss

Fig. 6. The results of CWRU bearing dataset, when knowledge is transferred from the scarce source domain.

remaining two experiments. The DTL method observes deterioration from 2.88% to 0.77% when more dissimilarities are introduced in the target domain (WC3). Moreover, the most dissimilar target domain (WC4) observes deterioration from +0.85% to −1.22% and also shows deterioration from the base case DL method instead of showing any improvement. In contrast, the DBTL observes significant improvements in all cases from both the base case DL, DTL, and previous scarce source sample experiments due to the ability to reduce negative transfer from the source domain while allowing positive transfer. The proposed method, DBTL, improves 12.09% from base case DL when the target domain (WC2) is most similar, which is also a significant improvement from previous experiments by only increasing source domain training samples. When transferring to more distinct target domains WC3 and WC4, the DBTL observes a 10.34% and 9.79% improvement with respect to the base case DL respectively, which is a significant



(a) WC1 to WC2 Accuracy

(b) WC1 to WC2 Training Loss

(c) WC1 to WC3 Accuracy

(d) WC1 to WC3 Training Loss

(e) WC1 to WC4 Accuracy

(f) WC1 to WC4 Training Loss

Fig. 7. The results of CWRU bearing dataset, when knowledge transferring from the extensive source domain.

Table 6
CWRU bearing dataset mean accuracy with the standard deviation of DL, DTL, and DBTL methods when knowledge is transferred from WC1 to WC4 by gradually increasing source domain data samples. The improvement of DTL and DBTL from the base case DL method is given in parentheses of the respective cell.

Model	930	1875	3750	7500	37226 (all)
DL	89.45 ± 1.03	89.45 ± 1.03	89.45 ± 1.03	89.45 ± 1.03	89.45 ± 1.03
DTL	90.29 ± 1.06 (0.85)	91.10 ± 1.20 (2.05)	90.33 ± 1.10 (1.28)	89.53 ± 1.54 (0.48)	88.23 ± 1.06 (−1.22)
DBTL	94.48 ± 1.20 (5.04)	96.60 ±0.51 (7.55)	97.65 ± 0.57 (8.60)	98.34 ± 0.20 (9.29)	99.24 ± 0.09 (9.79)

improvement compared to the previous experiments with a smaller source dataset.

4.3. Sensitivity study

To further demonstrate the strength of the proposed method, a sensitivity study is performed on the CWRU bearing dataset when transferring from WC1 to WC4 to identify the performance of each of the methods on gradually increasing source domain data samples. WC4 is selected as a target domain for the sensitivity study because it is the most distinct and acquires the most negative influence from the source domain WC1. The results when transferring from WC1 to WC4 by gradually increasing the source domain data samples are shown in Fig. 8, and the detailed results are provided in Table 6. The performance of the DTL method gradually declines after increasing the number of source domain data samples due to the increase in negative influence from the source domain. The accuracy drops from 91.10% to 90.33% and 89.53% when the source domain data samples increase from 1875 to 3750 and 7500 respectively. The DTL model observes further deterioration from the base case DL method by 1.22 when

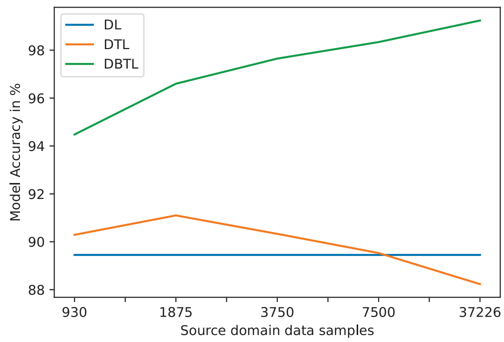


Fig. 8. The sensitivity study results of CWRU bearing dataset, when transferring from WC1 to WC4 by gradually increasing the data samples of the source domain.

Fig. 9. Offshore wind farm.

using all 37226 source domain data samples. On the other hand, the proposed DBTL method consistently improves when the source domain data samples increase. The accuracy improves from 94.48% to 99.24% when the source domain data samples increase from 930 to 37226. This sensitivity study thus clearly shows the limitations of the DTL method by not avoiding negative transfer and not fully utilizing the available source domain datasets. The proposed DBTL can circumvent negative transfer and increases its accuracy when using the complete source domain dataset. It has an improved model accuracy for both the scarce and extensive source domain data samples when compared to DL and DTL.

5. Wind farm case study

5.1. Real-world wind farm dataset

The proposed method is evaluated on an offshore wind farm's dataset as depicted in Fig. 9. The data was collected from the gearbox of two WTs. However, instead of using raw vibration signals to train, the DL models are trained using statistical features calculated from the raw vibration signals because the raw signals are highly complex containing high levels of process noise and multiple interfering vibration sources whilst also being strongly non-stationary due to continuously changing operating conditions. Moreover, each component has different behavior that produces different signals, bearing signals are stochastic, whereas gears or shafts signals are deterministic [44,45]. The statistical features are based on the indicators detailed in Peeters et al. [46] research.

In the first step, the noise of other components was removed from the raw vibration signal. In the following step, six statistical indicators are calculated on the filtered signals: Root Mean Square, Crest factor, Kurtosis, Moors kurtosis, Peak-to-Peak and Peak Energy Index.

Table 7
Wind farm dataset overview.

	WT1	WT2
Healthy period (days)	517	1069
Unhealthy period (days)	407	226
Sample Frequency (kHz)	20	20
Sensor Sensitivity (mV/g)	100	100
Source training samples	82/214	82
Target training samples	24	24
Testing samples	190	58

Table 8
Real-world wind farm dataset mean accuracy with the standard deviation of DL, DTL, and DBTL methods when knowledge is transferred from a scarce source domain. The improvement and deterioration of DTL and DBTL from the base case DL method is given in parentheses in respective cells.

Model	WT1 to WT2	WT2 to WT1
DL	90.75 ± 0.23	89.66 ± 0.25
DTL	91.19 ± 0.23 (0.44)	89.52 ± 0.24 (−0.13)
DBTL	94.58 ± 0.16 (3.84)	92.67 ± 0.09 (3.02)

WT bearings data have been used for fault detection experiments, as bearings are the most critical components. [47].

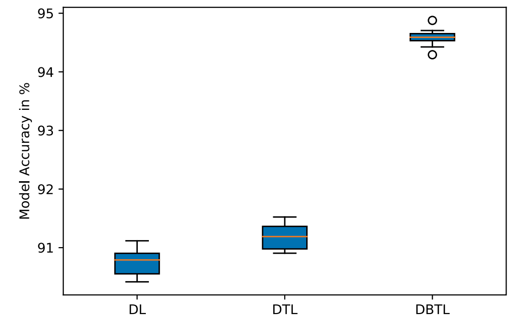
Both WTs observed an inner raceway generator bearing fault at the drive-end side. The history of the WTs is provided in Table 7, and it depicts that the unhealthy period is low. The healthy and unhealthy classes are used for the wind farm dataset. It is different from the CWRU bearing dataset that has six classes, five types of bearing faults and one healthy behavior class. A balanced dataset of healthy and unhealthy samples was selected for experiments. The raw vibration signal sampling rate is 20 kHz. A single statistical feature is calculated from 10 s of raw signal every 2 days, and therefore, there are approximately 150–200 measurements for each WT in one year. A sliding window of length 40 and step size 1 is used to get training samples from the statistical features by applying the data augmentation. A small window for the wind farm dataset is used because the statistical features are aggregated over multiple timestamps from the raw signal. The DL, DTL and DBTL models are trained separately on each feature of a sensor, and then the average is calculated over the features results. Two types of experiment were performed on the wind farm dataset, firstly, using 82 and then all (214) available training samples of the source domain. The target domain training samples are fixed at 24 for all experiments.

5.2. Wind farm dataset results

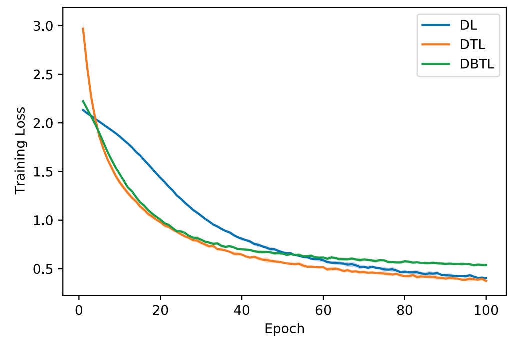
The proposed method is validated on a real-world wind farm dataset of two different WTs. The WTs are distinct based on different environments and WCs. However, the exact difference between the WTs is not known, in contrast to the experiments on the CWRU bearing dataset. This is particularly interesting, as in reality, it is generally unclear how different WTs in fact are in terms of operating conditions and failure modes. Therefore, it is necessary for transfer learning methods to be able to deal with this uncertainty. The following results include both the scarce and extensive source domain training samples experiments. For this case, no sensitivity study is performed since the actual difference between wind turbines is unknown in a real-life wind farm dataset.

5.2.1. Transferring from scarce source samples

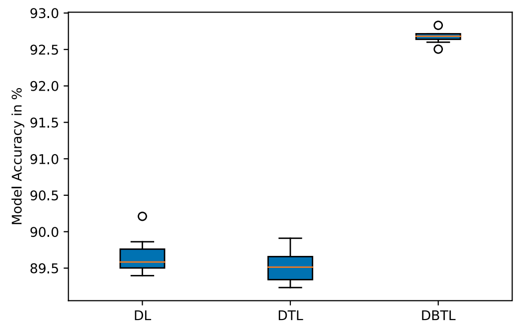
In the first experiment, we consider a small training dataset for the source of 82 samples (see Table 7). The results are shown in Fig. 10. The detailed results and the improvement and deterioration is provided in Table 8. The DTL observes a 0.44% improvement when transferring from WT1 to WT2, and it deteriorates with 0.13% when transferring



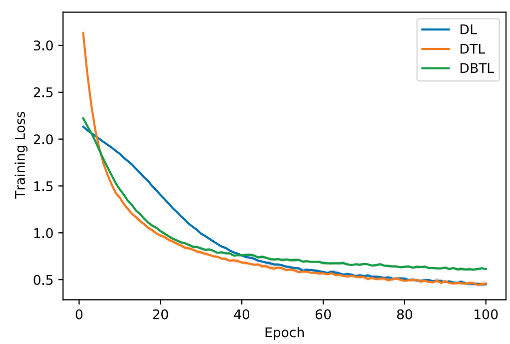
(a) WT1 to WT2 Accuracy



(b) WT1 to WT2 Training Loss

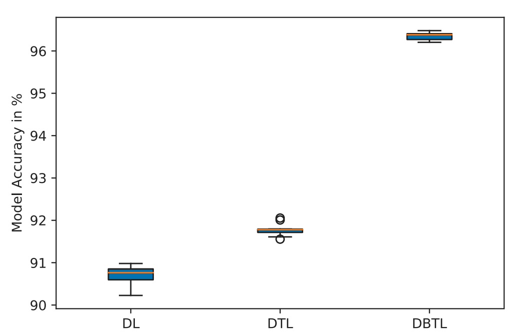


(c) WT2 to WT1 Accuracy

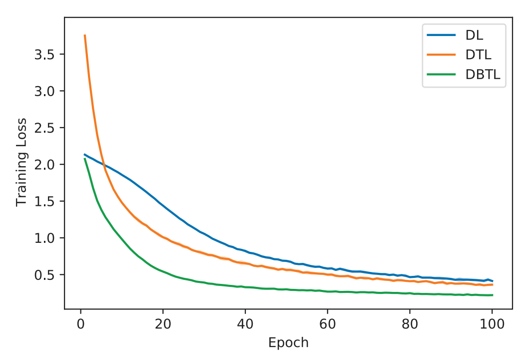


(d) WT2 to WT1 Training Loss

Fig. 10. The results of real-life wind farm dataset, when knowledge transferring from the scarce source domain.



(a) WT1 to WT2 Accuracy



(b) WT1 to WT2 Training Loss

Fig. 11. The results of real-life wind farm dataset, when knowledge transferring from the extensive source domain.

from WT2 to WT1 from base case DL. The influence of negative transfer is more notable in real-world data, which results in significant deterioration even when transferring knowledge from scarce source domain training samples. Nevertheless, the DBTL observes 3.84% and 3.02% improvement in both cases when transferring from WT1 to WT2 and WT2 to WT1 respectively.

Table 9
Real-world wind farm dataset mean accuracy with the standard deviation of DL, DTL, and DBTL methods when knowledge is transferred from a extensive source domain. The improvement of DTL and DBTL from the base case DL method is given in parentheses in respective cells.

Model	WT1 to WT2
DL	90.69 ± 0.25
DTL	91.78 ± 0.15 (1.10)
DBTL	96.35 ± 0.10 (5.67)

5.2.2. Transferring from extensive source samples

Due to the limitation of available data, only one experiment is performed when transferring from WT1 to WT2, not vice versa. The results when transferring from extensive source domain training samples are shown in Fig. 11, and the detailed overview is provided in Table 9. Both TL-based methods improve from the base case DL method and previous experiments with scarce source domain training samples. The DTL and DBTL respectively improve from 0.44% to 1.10% and 3.84% to 5.67% by increasing the training samples of the source domain.

6. Discussion

The results of the experimental analysis indicate an excellent performance of the proposed DBTL method. It outperforms conventional deep learning and deep transfer learning on two different datasets, the public dataset of the Case Western Reserve University (CWRU) and an experimental real-world wind farm dataset. The current state-of-the-art fault detection DTL methods are mostly validated on the CWRU dataset.

The proposed method significantly reduces the impact of negative transfer by leveraging the data over multiple similar machines. It makes the DBTL more robust for real-world data. In general fault detection methods are particularly developed for raw signals or the derived features from raw signals. The implementation of DBTL is similar to deep learning models and flexible to train on any dataset. The difference is that it requires updating data samples weight after each epoch which makes the difference between the DL and DBTL computation time negligible if trained on the same size dataset. Furthermore, it can be adapted to any deep learning architecture. Therefore, DBTL is validated on both the raw signals and the features. The ability of DBTL to identify the data samples with negative transfer and reduce their weights allows utilizing all available data from the source machine. To exhibit this strength two different experiments are conducted on each dataset with scarce and extensive source domain data samples. The proposed method depicts better fault prediction accuracy over the base case DL and the state-of-the-art DTL method for all experiments. The DTL method deteriorates when transferring from a more distinctive source machine with extensive data samples however, the DBTL depicts enhancement in all cases by reducing negative transfer. DBTL observes a 9.79% improvement from base case DL when transferring from WC1 to WC4 most distinctive source in CWRU bearing fault dataset where the normal DTL deteriorates by 1.22%. Similarly for the wind farm dataset when transferring from WT2 to WT1 the DBTL improves 3.02%, whereas the DTL method deteriorates by 0.13% from base case DL. The DBTL utilizes all the available data from the source by distributing the weights from high to low for the most significant to least significant data samples. It enables the proposed method to perform better in every case by avoiding overfitting. The proposed method is not applicable when transferring from already trained models as in the network-based DTL method, however, if the data is available a new model can be trained by combining the source and the target machine data. Researchers identified the importance of TL in the fault detection field, and the impact of negative transfer is addressed in this research. However, transferring from multiple sources is still an interesting topic for future research.

7. Conclusion

In this work, a new deep transfer learning method is proposed, coined deep boosted transfer learning (DBTL), for automated fault detection. It is a transfer learning approach for deep learning (DL) models to avoid negative transfer from the source domain. Furthermore, it allows for transferring learned knowledge to more machines with distinct behavior, thereby better utilizing all available source domain data. The effectiveness of the proposed method is validated on two different datasets. The CWRU bearing dataset and a real-life wind farm dataset. Moreover, two types of experiments on the CWRU and the wind farm datasets are performed by using scarce and extensive source domain training samples. The results show that the proposed method exhibit significant improvements for each test case compared to the base case DL and deep transfer learning (DTL) methods. In some experiments, the DTL method deteriorates from the simple DL method when the source domain is too different from the target domain, which the DBTL did not suffer from.

CRediT authorship contribution statement

Faras Jamil: Conceptualization, Methodology, Software, Formal analysis, Writing – original draft, Writing – review & editing. **Timothy Verstraeten:** Conceptualization, Supervision, Software, Writing – review & editing. **Ann Nowé:** Conceptualization, Supervision. **Cédric Peeters:** Conceptualization, Supervision, Writing – review & editing. **Jan Helsen:** Conceptualization, Supervision, Writing – review & editing, Funding acquisition.

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.

Acknowledgments

The authors would like to acknowledge FWO (Fonds Wetenschappelijk Onderzoek) for their support through the postdoctoral grant of Cedric Peeters (#1282221N) and the SBO Robustify project (#S006119N). This research was supported by funding from the Flemish Government under the “Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen” programme and under the VLAIO/Blue Cluster Supersized 4.0 ICON project. This work was also financed by the European Union under the H2020 PLATOON project (#872592).

References

[1] Global Wind Report 2021, Global Wind Energy Council, 2021, URL: https://gwec.net/global-wind-report-2021/.
[2] P.H. Alfredsson, A. Segalini, Introduction Wind Farms in Complex Terrains: An Introduction, The Royal Society Publishing, 2017, http://dx.doi.org/10.1098/rsta.2016.0096.
[3] Global Offshore Wind Report 2020, Global Wind Energy Council, 2020, URL: https://gwec.net/global-offshore-wind-report-2020/.
[4] J. Helsen, Review of research on condition monitoring for improved O&M of offshore wind turbine drivetrains, Acoust. Aust. (2021) http://dx.doi.org/10.1007/s40857-021-00237-2.
[5] B. Maples, G. Saur, M. Hand, R. van de Pietermen, T. Obdam, Installation, operation, and maintenance strategies to reduce the cost of offshore wind energy, 2013, http://dx.doi.org/10.2172/1220079.
[6] J. Helsen, Y. Guo, J. Keller, P. Guillaume, Experimental investigation of bearing slip in a wind turbine gearbox during a transient grid loss event, Wind Energy 19 (12) (2016) 2255–2269, http://dx.doi.org/10.1002/we.1979.
[7] B. Byrne, G. Houlsby, Foundations for offshore wind turbines, Phil. Trans. R. Soc. A 361 (1813) (2003) 2909–2930, http://dx.doi.org/10.1098/rsta.2003.1286.
[8] C.A. Irawan, D. Ouelhadj, D. Jones, M. Stålhane, I.B. Sperstad, Optimisation of maintenance routing and scheduling for offshore wind farms, European J. Oper. Res. 256 (1) (2017) 76–89, http://dx.doi.org/10.1016/j.ejor.2016.05.059.
[9] A.K. Jardine, D. Lin, D. Banjevic, A review on machinery diagnostics and prognostics implementing condition-based maintenance, Mech. Syst. Signal Process. 20 (7) (2006) 1483–1510, http://dx.doi.org/10.1016/j.ymssp.2005.09.012.
[10] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444, http://dx.doi.org/10.1038/nature14539.
[11] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, Vol. 25, Curran Associates, Inc., 2012, pp. 1097–1105, URL: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
[12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015.
[13] G. Hinton, L. Deng, D. Yu, G.E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups, IEEE Signal Process. Mag. 29 (6) (2012) 82–97, http://dx.doi.org/10.1109/MSP.2012.2205597.
[14] F. Jia, Y. Lei, J. Lin, X. Zhou, N. Lu, Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data, Mech. Syst. Signal Process. 72–73 (2016) 303–315, http://dx.doi.org/10.1016/j.ymssp.2015.10.025.
[15] Y. Lei, F. Jia, J. Lin, S. Xing, S.X. Ding, An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data, IEEE Trans. Ind. Electron. 63 (5) (2016) 3137–3147, http://dx.doi.org/10.1109/TIE.2016.2519325.
[16] W. Zhang, C. Li, G. Peng, Y. Chen, Z. Zhang, A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load, Mech. Syst. Signal Process. 100 (2018) 439–453, http://dx.doi.org/10.1016/j.ymssp.2017.06.022.
[17] V.T. Tran, F. AlThobiani, A. Ball, An approach to fault diagnosis of reciprocating compressor valves using Teager–Kaiser energy operator and deep belief networks, Expert Syst. Appl. 41 (9) (2014) 4113–4122, http://dx.doi.org/10.1016/j.eswa.2013.12.026.
[18] L. Chen, G. Xu, Q. Zhang, X. Zhang, Learning deep representation of imbalanced SCADA data for fault detection of wind turbines, Measurement 139 (2019) 370–379, http://dx.doi.org/10.1016/j.measurement.2019.03.029.

[19] T. Verstraeten, A. Nowé, J. Keller, Y. Guo, S. Sheng, J. Helsen, Fleetwide data-enabled reliability improvement of wind turbines, Renew. Sustain. Energy Rev. 109 (2019) 428–437, http://dx.doi.org/10.1016/j.rser.2019.03.019.

[20] C. Li, S. Zhang, Y. Qin, E. Estupinan, A systematic review of deep transfer learning for machinery fault diagnosis, Neurocomputing 407 (2020) 121–135, http://dx.doi.org/10.1016/j.neucom.2020.04.045.

[21] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359, http://dx.doi.org/10.1109/TKDE.2009.191.

[22] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, C. Liu, A survey on deep transfer learning, in: V. Kůrková, Y. Manolopoulos, B. Hammer, L. Iliadis, I. Maglogiannis (Eds.), Artificial Neural Networks and Machine Learning, ICANN 2018, Springer International Publishing, Cham, 2018, pp. 270–279, http://dx.doi.org/10.1007/978-3-030-01424-7_27.

[23] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, T. Zhang, Deep model based domain adaptation for fault diagnosis, IEEE Trans. Ind. Electron. 64 (3) (2017) 2296–2305, http://dx.doi.org/10.1109/TIE.2016.2627020.

[24] R. Zhang, H. Tao, L. Wu, Y. Guan, Transfer learning with neural networks for bearing fault diagnosis in changing working conditions, IEEE Access 5 (2017) 14347–14357, http://dx.doi.org/10.1109/ACCESS.2017.2720965.

[25] J. Li, X. Li, D. He, Y. Qu, A domain adaptation model for early gear pitting fault diagnosis based on deep transfer learning network, Proc. Inst. Mech. Eng. O 234 (1) (2020) 168–182, http://dx.doi.org/10.1177/1748006X19867776.

[26] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95 - International Conference on Neural Networks, Vol. 4, 1995, pp. 1942–1948, http://dx.doi.org/10.1109/ICNN.1995.488968.

[27] L. Wen, L. Gao, X. Li, A new deep transfer learning based on sparse auto-encoder for fault diagnosis, IEEE Trans. Syst. Man Cybern. A 49 (1) (2019) 136–144, http://dx.doi.org/10.1109/TSMC.2017.2754287.

[28] L. Guo, Y. Lei, S. Xing, T. Yan, N. Li, Deep convolutional transfer learning network: a new method for intelligent fault diagnosis of machines with unlabeled data, IEEE Trans. Ind. Electron. 66 (9) (2019) 7316–7325, http://dx.doi.org/10.1109/TIE.2018.2877090.

[29] Q. Wang, G. Michau, O. Fink, Domain adaptive transfer learning for fault diagnosis, in: 2019 Prognostics and System Health Management Conference, PHM-Paris, 2019, pp. 279–285, http://dx.doi.org/10.1109/PHM-Paris.2019.00054.

[30] B. Yang, Y. Lei, F. Jia, S. Xing, An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings, Mech. Syst. Signal Process. 122 (2019) 692–706, http://dx.doi.org/10.1016/j.ymssp.2018.12.051.

[31] D.-H. Lee, et al., Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks, in: Workshop on Challenges in Representation Learning, ICML, Vol. 3, (2) 2013, p. 896.

[32] S. Shao, S. McAleer, R. Yan, P. Baldi, Highly accurate machine fault diagnosis using deep transfer learning, IEEE Trans. Ind. Inf. 15 (4) (2019) 2446–2455, http://dx.doi.org/10.1109/TII.2018.2864759.

[33] P. Cao, S. Zhang, J. Tang, Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning, IEEE Access 6 (2018) 26241–26253, http://dx.doi.org/10.1109/ACCESS.2018.2837621.

[34] M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies, Image coding using wavelet transform, IEEE Trans. Image Process. 1 (2) (1992) 205–220.

[35] W.A. Smith, R.B. Randall, Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study, Mech. Syst. Signal Process. 64–65 (2015) 100–131, http://dx.doi.org/10.1016/j.ymssp.2015.04.021.

[36] M.T. Rosenstein, Z. Marx, L.P. Kaelbling, T.G. Dietterich, To transfer or not to transfer, in: Proc. Conf. Neural Information Processing Systems (NIPS '05) Workshop Inductive Transfer: 10 Years Later, Vol. 898, 2005, pp. 1–4.

[37] R.S. Perdana, Y. Ishida, Instance-based deep transfer learning on cross-domain image captioning, in: 2019 International Electronics Symposium, IES, 2019, pp. 24–30, http://dx.doi.org/10.1109/ELECSYM.2019.8901660.

[38] T. Wang, J. Huan, M. Zhu, Instance-based deep transfer learning, in: 2019 IEEE Winter Conference on Applications of Computer Vision, WACV, 2019, pp. 367–375, http://dx.doi.org/10.1109/WACV.2019.00045.

[39] W. Dai, Q. Yang, G.-R. Xue, Y. Yu, Boosting for transfer learning, in: Proceedings of the 24th International Conference on Machine Learning, ICML '07, Association for Computing Machinery, New York, NY, USA, 2007, pp. 193–200, http://dx.doi.org/10.1145/1273496.1273521.

[40] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, J. Comput. System Sci. 55 (1) (1997) 119–139, http://dx.doi.org/10.1006/jcss.1997.1504.

[41] Z. Zhao, T. Li, J. Wu, C. Sun, S. Wang, R. Yan, X. Chen, Deep learning algorithms for rotating machinery intelligent diagnosis: An open source benchmark study, ISA Trans. 107 (2020) 224–255, http://dx.doi.org/10.1016/j.isatra.2020.08.010.

[42] Z. Ye, J. Yu, Deep morphological convolutional network for feature learning of vibration signals and its applications to gearbox fault diagnosis, Mech. Syst. Signal Process. 161 (2021) 107984, http://dx.doi.org/10.1016/j.ymssp.2021.107984.

[43] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6 (1) (2019) 60, http://dx.doi.org/10.1186/s40537-019-0197-0.

[44] C. Peeters, P. Guillaume, J. Helsen, A comparison of cepstral editing methods as signal pre-processing techniques for vibration-based bearing fault detection, Mech. Syst. Signal Process. 91 (2017) 354–381, http://dx.doi.org/10.1016/j.ymssp.2016.12.036.

[45] C. Peeters, P. Guillaume, J. Helsen, Vibration-based bearing fault detection for operations and maintenance cost reduction in wind energy, Renew. Energy 116 (2018) 74–87, http://dx.doi.org/10.1016/j.renene.2017.01.056, Real-time monitoring, prognosis and resilient control for wind energy systems.

[46] C. Peeters, T. Verstraeten, A. Nowé, J. Helsen, Wind turbine planetary gear fault identification using statistical condition indicators and machine learning, in: International Conference on Offshore Mechanics and Arctic Engineering, Volume 10: Ocean Renewable Energy, American Society of Mechanical Engineers, 2019, http://dx.doi.org/10.1115/OMAE2019-96713.

[47] J. Helsen, C. Devriendt, W. Weijtjens, P. Guillaume, Experimental dynamic identification of modeshape driving wind turbine grid loss event on nacelle testrig, Renew. Energy 85 (2016) 259–272, http://dx.doi.org/10.1016/j.renene.2015.06.046.