Similarity-based Multi-source Transfer Learning Approach for Time Series Classification

Ayantha Senanayaka[1], Abdullah Al Mamun[1], Glenn Bond[4], Wenmeng Tian [1,3], Haifeng Wang[1], Sara Fuller[3], TC Falls[5], Shahram Rahimi[2], and Linkan Bian[1,3]

[1]*Department of Industrial and Systems Engineering, Mississippi State University, Starkville, MS 39762, United States*
aus20@msstate.edu
am4799@msstate.edu
tian@ise.msstate.edu
wang@ise.msstate.edu
bian@ise.msstate.edu

[2] *Department of Computer Science and Engineering, Mississippi State University, Starkville, MS 39762, United States*
rahimi@cse.msstate.edu

[3]*Center for Advanced Vehicular Systems (CAVS), Mississippi State University, Starkville, MS 39762, United States*
sfuller@cavs.msstate.edu

[4] *Information Technology Laboratory, US Army Engineer Research and Development Center, Vicksburg, MS, 39180, United States*
william.g.bond@erdc.dren.mil

[5]*Institute for Systems Engineering Research, Mississippi State University, Vicksburg, MS, 39180, United States*
tcfalls@iser.msstate.edu

ABSTRACT

This study aims to develop an effective method of classification concerning time series signals for machine state prediction to advance predictive maintenance (PdM). Conventional machine learning (ML) algorithms are widely adopted in PdM, however, most existing methods assume that the training (source) and testing (target) data follow the same distribution, and that labeled data are available in both source and target domains. For real-world PdM applications, the heterogeneity in machine original equipment manufacturers (OEMs), operating conditions, facility environment, and maintenance records collectively lead to heterogeneous distribution for data collected from different machines. This will significantly limit the performance of conventional ML algorithms in PdM. Moreover, labeling data is generally costly and time-consuming. Finally, industrial processes incorporate complex conditions, and unpredictable breakdown modes lead to extreme complexities for PdM. In this study, similarity-based multi-source transfer learning (SiMuS-TL) approach is proposed for real-time classification of time series signals. A new domain, called "mixed domain," is established to model the hidden similarities among the multiple sources and the target. The proposed SiMuS-TL model mainly includes three key steps: 1) learning group-based feature patterns, 2) developing group-based pre-trained models, and 3) weight transferring. The proposed SiMuS-TL model is validated by observing the state of the rotating machinery using a dataset collected on the Skill boss manufacturing system, publicly available standard bearing datasets, Case Western Reserve University (CWRU), and Paderborn University (PU) bearing datasets. The results of the performance comparison demonstrate that the proposed SiMuS-TL method outperformed conventional Support Vector Machine (SVM), Artificial Neural Network (ANN), and Transfer learning with neural networks (TLNN) without similarity-based transfer learning methods.

1. INTRODUCTION

The objective of this study is to develop an intelligent predictive maintenance (PdM) method for real-time classification of time series signals based on knowledge transfer from the states of multiple machines to the states of

https://doi.org/10.36001/IJPHM.2021.v13i2.3267

an unknown operating machine. Machine failures become a more challenging problem to solve in various phases of maintenance. In terms of maintenance, how to utilize an effective maintenance technique is a more intensive focus in the past few years (De Felice, Petrillo, and Austorino, 2014; Selcuk, 2017; Jin, Weiss, Siegel, and Lee 2016).

The state prediction of machinery is a significant problem. It provides the baseline information for fault detection, decision making, and PdM prior to the machine breakdowns (Xu, Liu, Jiang, Shen, and Huang, 2019). Recently, big data collected from the internet of things (IoT) motivates the research adopting innovative data-driven machine intelligence. Therefore, an intelligent state prediction model for operating machinery handling multiple machinery conditions can significantly advance PdM practices. As a result, the productivity and availability of machines can be increased, and the maintenance cost and downtime can also be reduced (Li & Zhang, 2020).

Two types of modeling techniques have identified system failures/breakdowns: physics-based models and data-driven models. Data-driven techniques are the most attractive methodology in PdM applications because many such methods have been developed to leverage sensor data for machine state prediction. Broadly, data-driven state/condition monitoring methods associated with PdM consist of a specific combination of steps, beginning with data acquisition. In industrial-level applications, different types of data can be collected by placing sensors to monitor the state of the operating machines. Vibration, acoustic, current, and temperature are the most common sensor signals available to establish PdM models. After completing the data acquisition process, the obtained signals/data are pre-processed (including de-noising and standardization) to extract critical process features. Then, machine learning (ML) techniques are utilized to build reliable models for fault recognition, fault classification, and maintenance related decision-making (Selcuk, 2017; Xu et al., 2019). With the development of artificial intelligence (AI) and ML practices such as deep learning (DL), researchers investigated how powerfully these techniques manifested in PdM (Li & Ma, 2020; Li et al., 2020; Li, Wang, and He, 2016).

In this study, **"sources"** are defined as the machines with sufficient data available for supervised modeling, which means that different states/conditions can be correctly identified from real-time signals. The **"target"** is the functioning machine not identical to the sources that is new to the ML model and for which we intend to identify the current state. Performance of the predictive models is highly dependent on the data distribution characteristics of the source and target (Li et al., 2016). While most conventional ML algorithms assume the process data collected from the source and target follows the same probability distribution, in real-world applications, the source and target domains may follow different distributions, and the assumption may not hold. Plenty of studies were developed by leveraging labeled data available in both domains. However, collecting labeled data or labeling existing data is highly expensive, inefficient, and time-consuming. Few studies on unsupervised clustering-based fault diagnosis have been conducted (Zhang, Yu, Chang, and Wang, 2015; Guo, Lei, Xing, Yan, and Li, 2019). The researchers considered clustering the training set and target separately to identify the different fault types in subsets. Afterward, learning techniques were utilized to learn feature patterns in the subsets. In one recently published study, Li and Zhang proposed a fault diagnosis method to address the partial domain adaptation problems using DL structures (Li & Zhang, 2020). In their method, they assumed that the label space of the target domain is a sub-space of source label space. Multiple classification models and conditional data alignment schemes were used to obtain domain-invariant features for the healthy state data in source and target domains. Then, the prediction consistency schemes were utilized to perform the partial domain adaptation. In another study, same group of the corresponding authors (Li et al., (2020)) extended their previous work by introducing "representation clustering," adopting autoencoder structures to address data sparsity issues with insufficient labeled data.

One of the promising techniques, called "transfer learning (TL)," which transfers pre-trained knowledge from the source to the target domain can resolve the problems mentioned earlier. The TL approach does not need to train predictive models from scratch, but it enables transferring the previously learned knowledge to initialize the target model(Pan & Yang, 2009). With TL-based advancement in condition monitoring, fault diagnosis, and status prediction, DL is a highly motivated research area in the last decade (Jin et al., 2016; Namuduri, Narayanan, Davuluru, Burton, and Bhansali, 2020; Niu, Liu, Wang, and Song, 2021).

Despite that, industrial-level applications cooperate with multiple sources with unpredictable failure modes, which leads to high computational cost, high complexity, and extra complications to the predictive analysis in various aspects. Also, existing TL-based machine state predictions mainly focused on supervised learning (labeled data available in targets) with one single source and one single target (Niu, Liu, Wang, and Song, 2020). However, this is not always the case in real-life state prediction problems, as it is possible to observe various machines as multiple sources and multiple different conditions. The challenges identified in multi-sources vs. one target problem are briefly summarized as follows:

- The multiple distribution properties/feature characteristics are involved in the analysis.
- Algorithms are highly complex and computationally intensive.

- Noisy data samples from various sensors may lead to performance deterioration in the predictive model.
- The relationship between multiple sources and the target samples is unknown.
- The current state of the target machine is unknown (No labels in the target).

Moreover, almost all multi-source studies with one target problem reveal that recognizable failure modes can be identified by adopting clustering techniques (Liu, Zhou, Xu, Zheng, Peng, and Jiang, 2018; Afridi, Ross, and Shapiro, 2016; Liu, Li, and Ma, 2016). However, unknown failure modes can arise in industrial-level applications, and it is still a challenge to account for them in such analysis.

This study aims to develop a powerful real-time operational state prediction method for the machinery using multiple sources by leveraging transfer learning and other statistical techniques (hierarchical clustering with Ward's linkage classification across the similarity groups). The proposed method is an extension of the fundamental one source vs. one target problem. A new domain called "mixed domain" can be established to determine the similarity across sources and the target. The mixed domain is classified into group clusters. A feed-forward multi-layer perceptron (MLP) model is introduced to develop group-based classification models. In this study, these models are called pre-trained models. In this way, the generalization error can be reduced. We assume that the optimized trained parameters could correctly predict the unknown targets partially gathered into the same cluster (dispersed in each cluster). Hence, we store the learned parameters (trained weights) from individual pre-trained models. Finally, these learned parameters are transferred to form the final predictive model to predict the unlabeled target. To our best knowledge, this is the first study to utilize multiple sources and targets together to characterize the relationship between them. The performance of the proposed methodology is validated on the application of unknown state prediction of rotating machinery. A computer-controlled machine, "Skill boss manufacturing system (AMATROL Inc., n.d.)," designed to demonstrate modern major functions in manufacturing and production systems, has been used for data collection. In addition, publicly available bearing datasets, CWRU(Case Western Reserve University, n.d.) and PU (Kimotho et al., 2016) are used to validate the proposed method. All classifier models were evaluated by standard performance measures, sensitivity-specificity, and precision-recall. Also, to balance them in one performance metric, balanced accuracy and F-measure have been observed. The performances of states prediction on unknown targets by SiMuS-TL compared with existing ML techniques. Well-known classifier models, SVM and ANN without transfer, and transfer learning method TLNN (R. A. N. Zhang et al., 2017) are utilized to explore the unknown state prediction of the target domain given multi-sources (with the same conditions and features extracted from the vibration signals).

The contribution of this work can be summarized as follows.

- A new ANN-based transfer learning approach for the multi-sources with one target problem is developed in this paper. Labeled multi-sources and unlabeled targets data are used in constructing a new domain called mixed domain. Extracted features from both sources and the target are divided into appropriate groups based on similarities and dissimilarities attributed by the hierarchical agglomerative grouping technique. Then, independent pre-trained models are developed to learn the task-specific group features. The final model was developed by transferring pre-trained parameters. The final training sample is a weighted sample chosen as per the ratio concerning the target sample distribution.
- This method guarantees less computational complexity on learning feature patterns as the pre-trained models were formed only on independent groups, which share the group-specific distribution characteristics. Here, we do not need any dense architectures to learn complicated feature patterns hidden in multiple sources and targets.
- The final model ensured unlabeled target prediction collectively in a simple setting because group-specific feature characteristics have transferred to build the final model.
- This method successfully addressed the difficulty of manipulating few or no labeled data in the target domain and various/mismatched distributions involved in the sources and the target (distribution discrepancy).
- The similarity-based transfer learning via ANN architecture provides the value-added advantage of leveraging pre-trained knowledge to construct a predictive model on unknown targets and efficiently make decisions.

The structure of this paper is organized in the following sections. A literature review and related studies in fault diagnosis, state prediction in rotating machinery, and PdM applications are introduced in Section 2. Methods and materials used in the proposed method, and learning architecture is discussed in Section 3. A case study based on applying the machine's state prediction of an unknown target is described in detail in Section 4, and finally, conclusions are discussed in Section 5.

2. LITERATURE REVIEW

In this section, an overview of the past literature on transfer learning deep neural networks, fault detection or machine's state prediction techniques, and applications related to predictive maintenance are discussed.

2.1. Fault diagnosis, state prediction and predictive maintenance (PdM)

Condition-based maintenance or predictive maintenance problem has been investigated vastly over the last decade (Selcuk, 2017). Predictive maintenance and fault diagnosis are under the field of prognostic and health management. Recent development in advanced data collecting technologies, big data, IoT, and cyber-physical systems motivates the efficient PdM techniques. In this framework, signal data acquisition and storage, data pre-processing, feature extraction, diagnosis or prognosis, decision making are the significant steps (Xu et al., 2019). The primary goal of these practices are to identify the failures of machines or equipment by monitoring the current operating condition to schedule maintenance plans on operating machines. As a benefit, it can reduce the cost of downtime and unexpected maintenance for machine failures (Li et al., 2016). However, machine failures in industrial systems were common scenarios that caused more damage to its regular ongoing operations. In the literature, different types of fault diagnosis or state prediction methods were available. These methods can be categorized into model-based, data-driven, signal-based, active fault-based, knowledge-based, and the combination of these methods. Combined methods were known as hybrid fault diagnosis methods, but data-driven methods were the most dominant methods. All these methods have been employed in various industrial PdM applications (Li et al., 2020; Li et al., 2016).

Data acquisition from several sensors involved maintenance problems known as multivariate time series data. Also, a higher number of independent variables consist in collected data referred to as high dimensional data. The procedures carried out before the analysis is called “pre-processing” collected data. Pre-processing the recorded signals involves several steps known as denoising and dimensionality reduction(Namuduri et al., 2020). Frequently used dimensionality reduction methods are principal component analysis (PCA), linear discriminant analysis (LDA), and quadratic discriminant analysis (QDA). Filtering the signal, statistical approaches, and transform domain (wavelet transform (WT), contourlet transform (CT)) are the available denoising methods (Mohan et al., 2014). Also, fast Fourier transform (FFT) and empirical mode decompositions (EMD) are other feature extraction methods available for data pre-processing (Flandrin et al., 2004). The next step was diagnosis and prognostics. Fault diagnosis supporting supervised ML was the famous technique that determines the target domain samples by learning the training examples. In this learning technique, the labeled samples are available. Famous supervised machine learning methods are SVM) naive Bayes (NB), logistic regression (LR) and ANN. Nevertheless, unsupervised approaches do not have labeled samples to learn, but it identifies the features hidden in the data structure. Self-organizing map (SOM), hierarchical clustering (HC), k-means, and dimensionality reduction are examples of unsupervised ML(Vakharia, Gupta, and Kankar, 2015a).

2.1.1. Deep learning (DL) and fault diagnosis

DL-based fault diagnoses are elaborated recently in the automotive industry, machine monitoring, environmental monitoring, and medical health applications. Different DL architectures have proven the ability of automatic feature extraction and deep feature learning in PdM. Autoencoders (AE), stacked autoencoders (SAE), convolutional neural networks (CNN), ANN, deep belief networks (DBN), and recurrent neural networks (RNN) are the recent advancements prevalent in literature (Gao et al., 2015; Vakharia, Gupta, and Kankar, 2015b).

Samanta and Al-Balushi (2003) presented ANN based fault diagnosis for rolling element bearings. Defective and normal time-dependent vibration signals of bearing were used in their analysis. Statistical features, such as root mean square (RMS), variance, kurtosis, and skewness, were used as input features to the ANN structure. The response variable was a binary variable that indicates the status of the bearing, i.e., normal or defective. Furthermore, multiple pre-processing steps, such as filtering the raw signals as high-pass and band-pass, WT, and envelope detection, were utilized in this analysis. Mao, He, Li, and Yan (2016) proposed another fault diagnosis method for the bearing faults by concerning a couple of drawbacks identified in machine learning techniques, SVM and ANNs, under extensive data analysis. Extreme learning machines (ELM) and auto encoder-based diagnosis approaches are introduced to overcome these weaknesses. The comparison study carried out with some state of art fault state prediction methods on rolling element bearing data shows that the proposed method is outperformed. Jia, Lei, Lin, Zhou, and Lu (2016) proposed a new intelligent diagnosis method for fault diagnosis in rotating machinery. Deep neural network-based fault feature extraction and intelligent diagnosis procedure have been investigated in their experimental study. Xia, Li, Xu, Liu, and Silva (2018) presented a method to reach more accurate diagnosis results by considering temporal and spatial information of training data collected from multiple sensors. CNN structure trained with these features employed in the application of fault diagnosis for rotating machinery. Liu et al., (2016) proposed another approach for fault diagnosis in rolling element bearing using sound signals. In their technique, they combine the short-time Fourier transform (STFT) and stacked sparse autoencoder (SSA). First, apply the STFT on sound signals and obtained spectrograms. Then utilize SSA to extract the fault features. Finally, softmax regression is introduced to classify the fault states.

Tran, Althobiani, and Ball (2014) introduced an advanced approach for PdM application, fault identification in valves located at reciprocating compressors adopting a DBN. In their analysis, three different measures in compressor valves, such as current, pressure, and vibration signals, were employed. Additionally, Teager–Kaiser energy operation (TKEO) distinguishes the fault patterns of three different signals. WT has been used for denoising the current and pressure signals. Another DL-based fault diagnosis method has been proposed, introducing a novel hierarchical diagnosis network (HDN) by Gan, Wang, and Zhu (2016). Hierarchical identification has been constructed by collecting the layers of DBNs for the mechanical systems. In their approach, two-layer HDN has been considered. In the first layer, fault type is identified, and then the ranking process is utilized to identify the severity of the faults. The performance of the proposed method was examined in a comparison study by constructing similar networks employing neural networks along with the backpropagation technique and SVM. Lu, Wang, Qin, and Ma (2017) examined the properties of stacked denoising autoencoder (SDA). He suggested that it was a reliable method for health state identification of raw signals, which contain ambient noise and fluctuations, generates in an operating condition. Xia, Li, Liu, Xu, and De Silva (2017) proposed an intelligent fault diagnosis method in another application using deep neural networks (DNN) and SDA. According to their invention, features of the signal learned by using a denoising autoencoder in an unsupervised mode. Then it used a DNN structure to train with few items of labeled data. The accurate results were reached by introducing fine-tuning in fault classification.

2.2. Transfer learning (TL) based fault diagnosis (FD) and predictive maintenance (PdM).

A fundamental assumption in traditional ML algorithms is that the training and testing data samples follow the same probability distribution or share the same feature space characteristics while learning and predicting processes. However, in practical implementations, many problems fail to hold this assumption. Once the distributions change in training and testing data, the performance of the predicting model degrades. In transfer learning methodology, learned knowledge is transferred from one task to another by developing the second task's learning processes' fulfillment. Also, the first and second tasks can be different, but they should be related. TL has become one of the powerful ML techniques involved in many research studies over the past decade (Perschl & Schmidt, 1993; Tsiakmaki, Kostopoulos, Kotsiantis and Ragos, 2020; Lu et al., 2017; Xia et al., 2017).

Figure 1 shows the significant difference between traditional ML and TL. The TL approach was first introduced in 1995. However, it was discovered in different names, lifelong learning, knowledge transfer, inductive or transductive transfer (Thrun & Pratt, 1998). In 1997, an exciting exploration introduced termed "multi-task learning" (Thrun, 1997). It learns multiple tasks, discovers the sources' latent characteristics, and employs them in another related task. Later, TL-based real-world applications show superior performances over the traditional approaches. Furthermore, traditional ML approaches such as neural networks, LR, SVM, and decision tree supported transferring scenarios in regression and classification problems become more realistic than improving independently (Zhu, 2008; Thrun et al.,1998; Thrun, 1997; Zhang, Hu, and Fang, 2010). Eventually, researchers studied the practice of TL in deep/convolutional neural networks for complex applications (De Felice et al., 2014; Paul, Rottensteiner, and Heipke, 2015; Shen, Chen, Yan, and Gao, 2016; Gideon, Khorram, Aldeneh, Dimitriadis, and Provost, 2017).

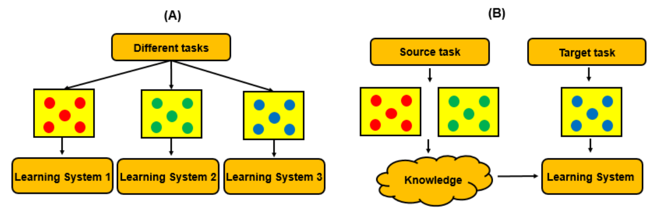

Figure 1: Different Learning Processes (A) Traditional machine learning (B) Transfer learning

Recently, transfer learning-based deep learning algorithms have been extensively used in fault diagnosis and PdM applications. TL-based DNN is one of the leading deep learning algorithms broadly adapted in FD and PdM. (Zhang et al., 2015) proposed a novel framework using TL and DNN to match source and target distributions differences and suggested deep transfer networks (DTN). In particular, two layers in the DNN structure work as feature extraction and discrimination layers to match the source and target distributions. Further, the complexity behind the DTN process declares a linear relationship with the number of training samples through empirical observations. (Xu et al., 2019) presented a new perspective, including TL to digital twin-assisted fault diagnosis. First, a diagnosis model was trained using DNNs in virtual space. Then, the diagnosis model, trained in virtual space, was migrated to physical space using TL. The performance of their proposed method was evaluated based on the car body-side production line application. (Shao, Member, Mcaleer, and Yan, 2018) introduced an FD method using DNNs and TL. It recommended that introducing a TL strategy to DNNs expedite the training process and model performances. According to their method, WT was used to convert the sensor data to images. The technique was called time-frequency imaging, and it tends to transform signal frequency time-series into the time-frequency distribution. Next, using pre-trained networks, low-level features were extracted. Then all model networks were fine-tuned through the labeled time-frequency images. The observed results from the three different experiments show that the test accuracies are approximately 100%. Also, researchers investigated the effectiveness of the popular pre-trained model's in FD applications (Xu et al., 2019; Sun, Ma, Zhao, Tian, and Yan, 2019; Shao et al., 2018).

2.3. Discussion of current research limitations

There exist weaknesses of currently available intelligent fault diagnosis methods incorporated with DL architectures. They analyze the problem at hand in the direction of modeling historical sensor data with deep architectures. Thus, training DL models is still very time-consuming and tends to be sensitive to historical data. As a consequence of the sensitivity, generalization of the trained model leads to complications(Macklin, 2019). In addition, fault diagnosis in complex systems may lead to significant difficulties while developing predictive models in real time (Tzafestas & Dalianis, 1994). Another aspect of limitations observed in currently available fault diagnosis methods and condition-based monitoring employing physical models for decision making (Chen, 2012; S. Lu et al., 2017). However, it may not be beneficial for complex systems because establishing explicit physical models for decision-making is extremely difficult and time-consuming (G. Xu et al., 2019). Therefore, developing data-driven methods to eliminate the complex training is adequate.

Transferring learned knowledge from the previously studied problem might accelerate training and improve the model's predictive performance. The DL-based TL approaches are widely acknowledged in fault diagnosis/prediction and conditioned-based monitoring, among other applications. The majority of them addressed the labeled target domains, while unlabeled targets, or few numbers of labeled samples, were rarely investigated. In addition, the research efforts are usually focused on solving single-source transfer learning with one target. However, an integrated study to reveal the hidden relationships of target and source data, combining multiple sources and targets, has not been examined broadly in these studies.

It is essential to consider the distribution characteristics between sources and the target concurrently to describe their hidden relationships to develop a predictive model for the unknown target. Therefore, this study introduces a novel time-series classification method for rarely labeled targets using similarity-based knowledge transfer for unknown state prediction in multi-source problems.

3. METHODOLOGY

In this section, theoretical details of the proposed SiMuS-TL method are discussed in detail.

3.1. Mathematical notation and intention

A set of multiple sources (multiple machine's conditions monitored previously) are denoted as $S = \{S_1, S_2, S_3, \dots, S_n\}$, and the target (An operating machine, with its state unknown) is denoted as T.

Notation	Description
$\mathcal{V}$	Vibration signal/time series space
$V_{(.)}$	Vibration signal
$V_{n_{S_i}}^{S_i}$	Vibration signal from i^{th} source domain where n_{S_i} number of vibration signals are available
$V_{n_t}^{T}$	Vibration signals from the target domain where n_T number of vibration signals are available
$\mathcal{D}_{S_i}$	The i^{th} source domain, $i = 1,2,3, \dots, n$
$\mathcal{D}_T$	The target domain
$\mathcal{D}_{mix}$	A new domain called "mixed domain" which is built with weighted samples from multi-sources and the target.
$\mathcal{D}_{G_m}$	The m^{th} identity group domain, $m = 1,2,3, \dots, k$
D_{S_i}	The i^{th} source domain data
D_T	The target domain data
D_{mix}	A new domain data
D_{G_m}	The m^{th} identity group domain data, $m = 1,2,3, \dots, k$
$P_{S_i}(V)$	The marginal probability distribution of the i^{th} source
$P_T(V)$	The marginal probability distribution of the target
$P_{mix}(V)$	The marginal probability distribution of the D_{mix}
$P_{G_m}(X)$	The marginal probability distribution of the m^{th}identity group, $m = 1,2,3, \dots, k$
$P_{G_m^*}(X)$	The marginal probability distribution of the m^{th} identity group, satisfying the condition $P_T^{G_m^*} > 0, m = 1,2, \dots , k^*$
$\mathcal{Y}$	A label space of vibration signals multiple sources and the target

Table 1. List of notations

$y_{S_i}^c$	Multiple states of machines which belongs to the i^{th} source, $y_{S_i}^c = \{c \in \mathbb{R}\} \in \mathcal{Y}$, there exist c machine states
y_T^c	Multiple states of machines which belongs to the target $y_T^c = \{c \in \mathbb{R}\} \in \mathcal{Y}$, there exist c machine states
$y_{G_m^*}^c$	Multiple states of machines which belongs to the m^{th} identity group, $y_{G_m^*}^c = \{c \in \mathbb{R}\} \in \mathcal{Y}$, there exist c machine states, $m = 1,2,3, \dots, k^*$
$\mathcal{X}$	A feature space of vibration signals
X_{S_i}	Features extracted from i^{th} source, $i = 1,2,3, \dots, n$
X_T	Features extracted from target
G_m	The m^{th} group found in D_{mix}, $m = 1,2,3, \dots, k$
G_m^*	The m^{th} identity group found in D_{mix} satisfying the condition $P_T^{G_m^*} > 0, m = 1,2, \dots, k^*$
$\mathcal{T}_T$	Target domain task
$\mathcal{T}_{mix}$	Mixed domain task
$\mathcal{T}_{G_m^*}$	Identity group domain task
$F_s(.)$	A function F_s mapping data samples of D_{S_i} to D_{mix}
$F_c(.)$	A function F_c mapping data samples of D_{mix} to G_m
$F_T(.)$	A function F_T mapping data samples of G_m^* to D_T^*
P_{G_m}	Domain sharing percentage of the m^{th} group in D_{mix}, $m = 1,2,3, \dots, k$
$P_T^{G_m^*}$	Target sample distribution of the m^{th} identity group, $m = 1,2,3, \dots, k^*$
$X_{(.)}^{G_{(.)}}$	Group-based vibration signal features
$ANN_{G_m^*}$	Group-based m^{th} ANN model, $m = 1,2, \dots, k^*$
$W_m^{(l)}$	Group-based optimized/trained weights/parameters of m^{th} model related to l^{th} layer, $m = 1,2, \dots, k^*$
D_T^*	Group-based vibration features data selected for final training set based on target sample distribution

Table 1. List of notations (Continued 1)

w_{s_i}	A mixing weight defines for the specific source
w_T	A mixing weight defines for the target domain
p	Number of vibration features (i.e., principal components) extracted
n	The total number of multiple sources available
N	Total number of vibration signal data in D_{mix}
i	The index variable for multiple sources
m	The index variable for identity groups found in D_{mix}
j	The index variable for sample data in D_{mix}
n_{S_i}	The number of vibration signals belongs to i^{th} source
n_t	The number of vibration signals belongs to target
k	The number of identity groups found in D_{mix}
k^*	The number of identity groups satisfying the condition $P_T^{G_m^*} > 0$

Table 1. List of notations (Continued 2)

In this paper, a space of vibration waveform signal/time-series denoted as $\mathcal{V}$, calculated features from vibration signals space denoted as $\mathcal{X}$, and the shared label space for source and target is $\mathcal{Y}$. In fault diagnosis problems, distinct fault types have characteristic vibrational features, and extracting feature parameters from raw vibration data for fault/state classification is crucial. We considered that extracted features from time series data could classify different fault/state types in operating machinery. Therefore, we selected the time and frequency domain features in vibration signals to demonstrate the proposed method's effectiveness in rotating machinery. If we are supposed to extend this methodology with multiple faults/state problems, we might employ wavelet (WT) analysis (Donoho, 1995) methods for noise removal to isolate multiple fault features. The labels of the vibration signals ($y_{S_i}^c, y_T^c$), and multiple states of machines which belongs to the source or target $y_{(.)}^c = \{c \in \mathbb{R}\} \in \mathcal{Y}$, there exist c machine states. The marginal probability distribution of the i^{th} source is denoted as P_{S_i} (V), where time-stamp waveform signal $V = \{V_1, V_2, \dots, V_n\} \in \mathcal{V}$. There are n source domains (where the state and the machines relationship studied) available and denoted as $\mathcal{D}_{S_i}$, where $i = 1,2,..n$. In this study, we considered that one target domain $\mathcal{D}_T$ is available. The specific domain, sources and target can be defined as $\mathcal{D}_{S_i} = \{\mathcal{V}, P_{S_i}(V)\}$ and $\mathcal{D}_T = \{\mathcal{V}, P_T(V)\}$ respectively. The source domain data are denoted as $D_{S_i} =$

$\left\{\left(V_1^{S_1}, y_1^{c_{S_i}}\right), \dots, \left(V_{n_{S_i}}^{S_i}, y_{n_{S_i}}^{c_{S_i}}\right)\right\}$ and target domain data are denoted as $D_T = (V_1^T, V_2^T, \dots, V_{n_t}^T)$.

3.2. Assumptions

- Extracted features from time-dependent vibration signals belong to target and multi-sources can be grouped into groups $(G_1, G_2, \dots, G_k)$ based on similarities identified using a hierarchical agglomerative clustering technique.
- The label space of the multiple sources and the target domain are same, denoted as $\mathcal{Y}$.
- Features extracted from vibration signals of the target machine in identity groups follow the same probability distribution as the features from source machines vibration signals samples.
- A set of parameters ($W_m^{(l)}$: $l = 1,2,3 \dots, m = 1,2,3, \dots k^*$) obtained from generalized group-based models hold the potential of correctly predicting the partially distributed target's labels. Because, they follow the same group-dependent feature characteristics with source ($D_{G_m^*} = (X_1^{S_{(.)}}, X_2^{S_{(.)}}, \dots, X_{(.)}^T, \dots, X_{n_{gm}}^{S_{(.)}}$).

3.3. Outline of the algorithm of similarity-based multi-source transfer learning

The SiMuS-TL method includes a sequence of clustering, classification, and transfer learning processes. The workflow diagram of the proposed method is shown in Figure 2.

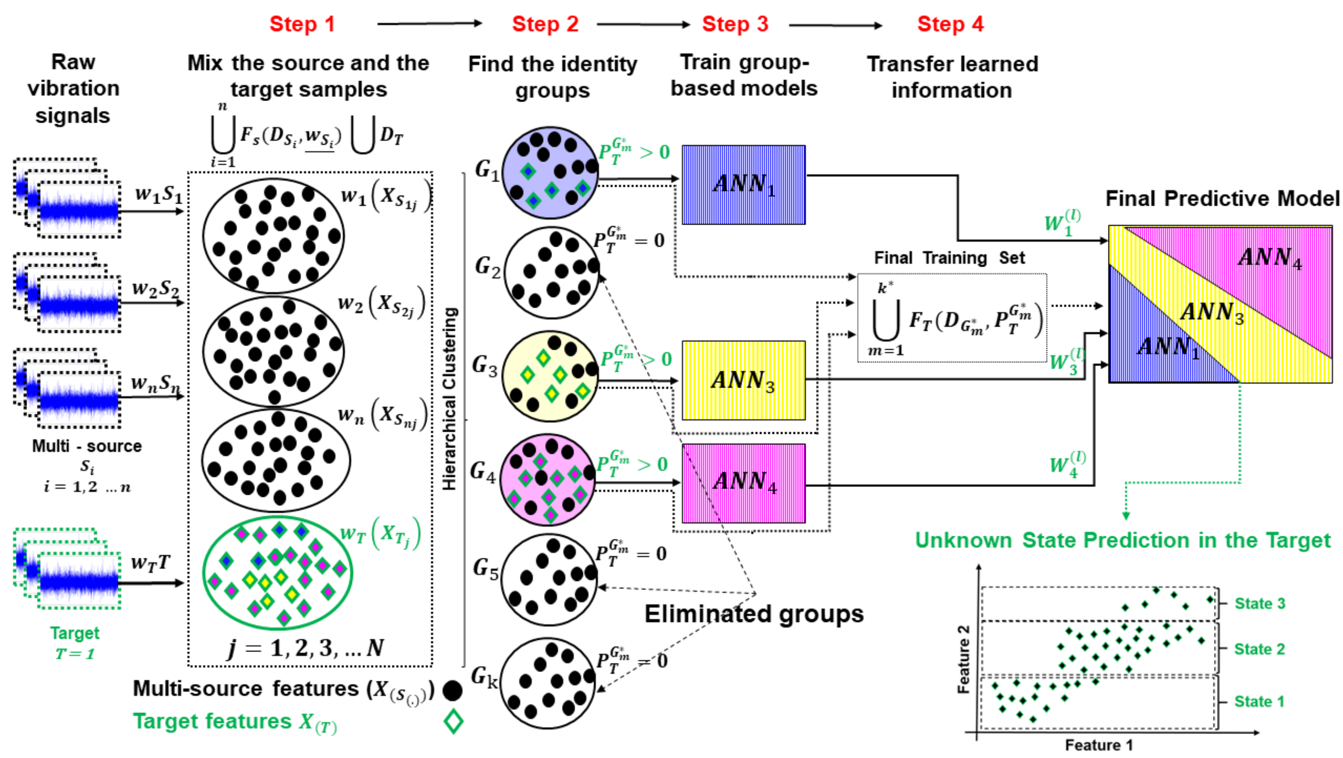

Figure 2. The workflow diagram of the proposed method. Sequence of process, raw vibration data, feature extraction, mixed domain clustering, pre-trained models and final model

Step 1. Mix the source and the target samples.

This step aims to form a new domain to define the similarities that the target shares with the multi-sources. The multi-sources and target vibration signals or time-dependent raw vibration signals are used to establish the "mixed domain." We compute multiple features to represent the vibration signals in this new domain to identify similarities between target and multi-sources. The waveform signals of sources and the target are considered the input of this step and the features extracted are the output of this step.

Step 2. Find the identity groups.

The objective of this step is to identify the identity groups that follow the condition $P_T^{G_m^*} > 0$ for further analysis. Based on the features extracted in Step 1, similarities are identified within multiple sources, and the target is distributed to independent groups using a hierarchical agglomerative clustering technique. We have chosen the hierarchical clustering with ward's linkage method. These identified groups are called "identity groups" and are denoted as

$G_1, G_2, \ldots, G_k$. These groups consist of unique group characteristics, and k number of groups found in D_{mix}. Let us identify groups that contain the target samples, denoted as G_m^* where $P_T^{G_m^*} > 0$. If there are no target samples ($P_T^{G_m^*} = 0$) in G_m, the m^{th} group is not considered in the current analysis.

Step 3. Train group-based models

This step aims to find the layer-wise $W_m^{(l)}$ for each identity group. Given the identity groups with $P_T^{G_m^*} > 0$, suppose that we have found k^* groups result in $P_T^{G_m^*} > 0$, and denoted them as $G_1^*, G_2^*, G_3^* \ldots G_{k^*}^*$, where $k^* \leq k$. We train k^* ANN models ($ANN_{G_1^*}, ANN_{G_2^*}, \ldots , ANN_{G_{k^*}^*}$) using source samples in the respective identity groups. These models are called pre-trained ANNs in this method. We expected to find the optimum parameters of each pre-trained ANN models. Let us define the optimal parameters/weights for the layers in the m^{th} pre-trained ANN as $W_m^{(l)} : l = 1,2,3 \ldots , m = 1,2,3, \ldots k^*$, where l denotes the layer number in the m^{th} pre-trained models.

Step 4. Transfer the learned information

The parameters obtained from pre-trained models ($W_m^{(l)}$) are transferred to the final model, and this process is called "transplanting layers." It copies the layer-wise weights from the pre-trained models to the final ANN model. In addition, randomly initialized connecting layers have been used to compile the copied layers in the final model. The final model training is subjected to weighted samples chosen from groups as per the $P_T^{G_m^*}$. We dropped the input and output layers of pre-trained models. Transplanted weights are frozen in the final model; otherwise, we lose the benefits of transferring pre-trained knowledge in the final model. The input of this step is the parameters/weights transferred to the final model and preparing the final training set. The output will be the generalized model to predict the complete set of target labels. The output of this ANN model prediction is the target machinery's states prediction in operation.

3.4. Detailed SiMuS-TL method

The problem of finding the state of the operating machine in the target domain is an unsupervised learning problem. The partially shared similarity measures of the target can be characterized by the formation of the D_{mix}, where the labels are irrelevant.

3.4.1. Step 1. Mix the source and target samples

The target and multi-sources are used to create the mixed domain D_{mix}. The mixing weights are calculated based on the available sample size of the target and sources. Suppose that you have two sources and one target. The number of time-dependent vibration signals found in the target is denoted as n_t , and sources 1 and 2 consist of n_1 and n_2 samples, respectively. The weighting coefficients for target, sources 1 and 2 are represented as w_T, w_{s_1}, w_{s_2}, respectively.

$$w_{s_1} = \frac{n_1}{n_t}, w_{s_2} = \frac{n_2}{n_t}, w_T = \frac{n_t}{n_t} = 1 \tag{1}$$

One can down/up sample the sources as per the target to include all samples from the target domain. We define a new domain called “mixed domain ($\mathcal{D}_{mix}$)” as in the Equation (2).

$$\mathcal{D}_{mix} = \{\mathcal{V}, P_{mix}(V)\} \tag{2}$$

$P_{mix}(V)$ denotes the marginal probability distribution, where $V = \left\{V_{(.)}^{S_{(.)}}, \ldots, V_{(.)}^{T}, \ldots V_{(N)}^{(.)}\right\} \in \mathcal{V}$. The data samples belong to the $\mathcal{D}_{mix}$ are denoted as D_{mix}.

$$D_{mix} = \{\cup_{i=1}^{n} F_s(D_{S_i}, w_{S_i}) \cup D_T\} \tag{3}$$

A sampling function F_s is defined for establishing D_{mix} using the weight w_{S_i} for each specific domain D_{S_i}. In this study, time-domain features and frequency domain features are extracted. Principal component analysis (PCA) is used for dimension reduction. It calculates the scores to obtain the variance by mapping original data onto spaces spanned by the eigenvectors, linking to the sample covariance matrix without losing major information (Jackson, 1991). The established mixed domain is used to find identity groups using the features and similarity grouping technique discovered by unsupervised clustering where the labels are irrelevant. How to set the similarity grouping will be further discussed in section 3.4.2.

3.4.2. Step 2. Find the identity groups

Identity groups are characterized using a hierarchical agglomerative approach (Szmrecsanyi, 2009). It determines which set of instances are similar to each other within sets of elements. A proper distance measure of connecting pairs of observations and a linkage process that explains the variation of sets based on the pairwise length in the sets is required. The selection of a relevant measure may affect the similarity groups. We investigated hierarchical clustering with multiple distance metrics and with different linkage methods. The planned distance measures and linkage method are shown below. Suppose that two data points are denoted as *a* and *b*. The distance between the $\boldsymbol{n}$-dimensional vector *a* and *b* is indicated as *d*, and different similarity distance metrics (Shirkhorshidi et al., 2015) are summarized in Table 2.

Distance metric	Formula
Euclidean distance *(Most frequently used distance metric in statistical studies)*	$d(a,b) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2}$
Squared Euclidean distance	$d(a,b) = \sum_{i=1}^{n}(a_i - b_i)^2$
Manhattan distance *(A grid path distance between two points)*	$d(a,b) = \sum_{i=1}^{n}\lvert a_i - b_i\rvert$
Chebyshev Distance *(Along any coordinate dimension the greatest difference)*	$d(a,b) = max_i\lvert a_i - b_i\rvert$
Canberra Distance *(Weighted version of Manhattan distance)*	$d(a,b) = \sum_{i=1}^{n}\frac{\lvert a_i - b_i\rvert}{\lvert a_i\rvert + \lvert b_i\rvert}$

Table 2. Distance metrics and formula

(1) Single linkage: The individual entities form the groups by merging the nearest neighbors, related to each other by the smallest distance or most significant similarity. Find the smallest distance between groups and merge the two group objects. Let G_1 and G_2 form a group, the distances update formula $d(G_1 \cup G_2, G_3)$, between *(G_1G_2)* and another group G_3 can be computed by Equation (4).

$$d(G_1 \cup G_2, G_3) \; = min\{d(G_1, G_3), d(G_2, G_3)\} \quad (4)$$

(2) Complete linkage – Complete linkage has similar criteria as a single linkage, but it finds the distance between groups based on two elements that are most separated. For example, after the general initialization, merging the corresponding two groups. Let G_1 and G_2 form a group, the distances update formula $d(G_1 \cup G_2, G_3)$, between *(G_1G_2)* and another group G_3 can be computed by Equation (5).

$$d(G_1 \cup G_2, G_3) \; = max\{d(G_1, G_3), d(G_2, G_3)\} \quad (5)$$

(3) Average linkage – Average linkage finds the distance between two groups as the average distance between all pairs of elements where one pair belongs to each group. For example, after the general initializing step of selecting criteria, let G_1 and G_2 form a group, the distances update formula $d(G_1 \cup G_2, G_3)$, between *(G_1G_2)* and another group G_3 can be computed by Equation (6), where n_{g_1}, n_{g_2}, and n_{g_3} are the total number of elements in different groups in the mixed domain.

$$\mathrm{d}(\mathrm{G}_1 \cup \mathrm{G}_2, \mathrm{G}_3) \; = \frac{\mathrm{n}_{\mathrm{g}_1}\mathrm{d}(\mathrm{G}_1,\mathrm{G}_3)+\mathrm{n}_{\mathrm{g}_2}\mathrm{d}(\mathrm{G}_2,\mathrm{G}_3)}{\mathrm{n}_{\mathrm{g}_1}+\mathrm{n}_{\mathrm{g}_2}} \quad (6)$$

(4) Ward's linkage – This is also known as the minimum variance method (Murtagh & Legendre, 2014; Szmrecsanyi, 2009). Initially, all elements in the space considered as a single object cluster. Then, the ward's method forms a cluster based on minimizing the total variance within pair of clusters. In other words, two groups are merged in finding the smallest increase calculated by the sum of squared error (SSE). Then, the distances update formula can be computed by Equation 7.

$$d(\mathrm{G}_1 \cup \mathrm{G}_2, \mathrm{G}_3) = \sqrt{\frac{(\mathrm{n}_{\mathrm{g}_1}+\mathrm{n}_{\mathrm{g}_3})\mathrm{d}(\mathrm{G}_1,\mathrm{G}_3)+(\mathrm{n}_{\mathrm{g}_2}+\mathrm{n}_{\mathrm{g}_3})\mathrm{d}(\mathrm{G}_2,\mathrm{G}_3)-\mathrm{n}_{\mathrm{g}_3}\mathrm{d}(\mathrm{G}_1,\mathrm{G}_2)}{\mathrm{n}_{\mathrm{g}_1}+\mathrm{n}_{\mathrm{g}_2}+\mathrm{n}_{\mathrm{g}_3}}} \quad (7)$$

Similarities are identified within sources, and the target is separated into independent groups using a hierarchical agglomerative technique. We have chosen the most frequently used distance metric in statistical studies, "Euclidean distance," with ward's linkage method in the proposed method. However, we tested with all these distance metrics measures and linkage methods in our experiments.

The group-based domains can be defined and denoted as $\mathcal{D}_{G_m}$ as in Equation (8).

$$\mathcal{D}_{G_m} = \{\mathcal{X}, P_{G_m}(\mathcal{X})\}, \tag{8}$$

where $m = 1,2,3 \dots, k$. Suppose that the m^{th} group data is denoted as D_{G_m} which consists of both source and target data.

$$D_{G_m} = (X_1^{S_{(.)}}, X_2^{S_{(.)}}, \dots, X_{(.)}^{T}, \dots, X_{n_{gm}}^{S_{(.)}}), \tag{9}$$

where, $X_{(.)}^{S_{(.)}}$ denotes the mixed domain source samples and $X_{(.)}^{T}$ denotes the target samples. We determine a correlation-based similarity ratio or a group-based measure called "target sample distribution ($P_T^{G_m^*}$)" across groups, calculate as in Equation (10).

$$P_T^{G_m^*} = \frac{Number\ of\ target\ samples\ in\ m^{th} group}{Total\ number\ of\ target\ samples\ in\ the\ target} \% \tag{10}$$

The method of obtaining the targets in each group is fundamentally based on general searching criteria. First, set an index variable j for sources and targets ($j = 1,2,3..N$). This process is called "indexing." Next, explore the particular index variable in a specific group (Looping in every group and obtaining the target indexes). This way allows us to get the number of targets and source samples in a particular group. After we obtained the number of targets in each group, we can calculate $P_T^{G_m^*}$. The ratio $P_T^{G_m^*}$, where $0 < P_T^{G_m^*} < 1$ for $m = 1,2, \dots, k^*$. Suppose that we found groups with $P_T^{G_m^*} > 0$, which are denoted as $G_1^*, G_2^*, G_3^* \dots G_{k^*}^*$ where $k^* \leq k$. If $k^* < k$ the groups ($k - k^*$) may end up with $P_T^{G_m^*} = 0$, and they are excluded in the analysis.

3.4.3. Step 3. Training group-based models

Target samples in G_m^*, may follow similar distribution characteristics and features identical to the group G_m^*. The hidden differences between source and target domain samples are adjusted in this process as we group targets as they are gathered into identity groups with multi-source samples. Therefore, supervised ML models can be built on groups ($G_1^*, G_2^*, G_3^* \dots G_{k^*}^*$) to predict the target domain partially. However, our objective is not to build a model to make partial predictions on the target but to predict the complete target domain in one setting. We train k^* group-based pre-trained ANN models ($ANN_{G_1^*}, ANN_{G_2^*}, \dots, ANN_{G_{k^*}^*}$), using the group features from sources with labels across groups. ANN models consist of multiple layers, such as input, hidden layers, and fully connected dense layers. Let us define the layer-wise learned parameters/weights in $ANN_{G_m^*}$ as $W_m^{(l)} : l = 1,2,3 \dots, m = 1,2,3, \dots, k^*$, where l denotes the layer number of $ANN_{G_m^*}$. The identity group's domain task can be defined as $\mathcal{T}_{G_m^*}$.

$$\mathcal{T}_{G_m^*} = \{y_{G_m^*}^c \in \mathcal{Y}, ANN_{G_m^*}\} \tag{11}$$

It consists of two responses of interest, ($y_{G_m^*}^c, c \in \mathbb{R}$) and a group-based objective function ($ANN_{G_m^*}$). The cross-entropy loss will be combined with the softmax activation for binary/multi-class classification problems. The backpropagation and stochastic gradient descent (SGD) algorithms examine to find the optimum layer-wise weight matrices. Furthermore, if $c > 2$, this is the cost function becomes the categorical cross-entropy (CE), as shown in Equation (12).

$$CE = -\sum_{i=1}^{c} y_i \, log\big(f(x_i)\big) \tag{12}$$

where the CE loss with softmax activation $f(x_i)$, the binary indicator (y_i) results '1' if the class label is correct classification for the i^{th} observation. The loss is calculated for each class per observation and sum up to the result. We have obtained the learned parameters of each objective function of pre-trained ANN model.

3.4.4. Step 4. Transfer the learned information

The proposed SiMuS-TL method is established based on the assumption noted; the parameters from a specific pre-trained model could find the labels of partial targets that come into particular groups. The target domain task($\mathcal{T}_T$) mainly depends on source labels and transfer learning. It can be represented by two components, $y_T^c \in \mathcal{Y}$and ANN_T learned by training data pairs from $\mathcal{D}_{G_m^*}$ and based on $P_T^{G_m^*}$ as shown below.

$$\mathcal{T}_T = \{y_T^c \in \mathcal{Y}, ANN_T\} \tag{13}$$

Furthermore, the final ANN_T model, are essentially transferring pre-trained knowledge to initialize the model and learning group-based feature patterns from the weighted samples collected. The final training data set is generated for predicting the target domain, denoted as Equation (14).

$$D_T^* = \{\cup_{m=1}^{k^*} F_T(D_{G_m^*}, P_T^{G_m^*})\} \tag{14}$$

Figure 3. explains the final step of the proposed SiMuS-TL method.

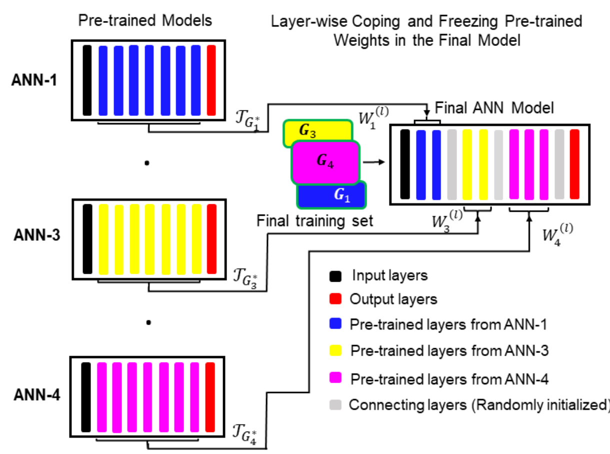

Figure 3. Transfer pre-trained weights from multiple pre-trained models as per the target sample distribution.

The $\mathcal{T}_{G^*_{(.)}}$ represents the pre-trained model task. Learned parameters $W_m^{(l)}$ from multiple $ANNs$ are transferred to the final model. Customized layer coping in the final ANN model implements transferring parameters. If we obtained a higher ratio $P_T^{G_m^*}$ for the specific group, we might use more weights that are layer-wise in the final network. Thus, we copied arbitrary layers with learned parameters in the final network. This study did not address which layer to pick on transferring or optimal layers from the specific model to transfer. Accompanying the adaption is succeeded; the purpose is to minimize the final model's loss. Importantly, parameters transferred from pre-trained ANNs, not required
to train over as they have trained already. Therefore, these parameters are frozen through the process of fine-tuning the final model. The output of this predictive ANN model is the complete set of target states of the machinery under operation.

3.5. An example of a practical scenario on state monitoring for rotating machinery operations.

The proposed SiMuS-TL method is described using a practical example of faulty bearings. First, assume three types of bearings available, bearing 1, 2 and 3, in three different locations. Suppose that bearing 1 and 2 are multiple sources, and bearing 3, will be the unknown target currently in operation. Bearing 1 and 2 share similarities with bearing 3, but they are not identical to bearing 3. A run to failure experiments have been arranged for two bearings (bearing 1 and 2) and studied the different states. They failed due to certain reasons, and three types of states are identified ($c = 3$) in degradation patterns and define as “Normal,” “Defective,” and “Failure.” The time-dependent waveform signals are captured from target bearing 3. Let us develop the proposed methodology to predict the states of heath of bearing 3. Figure 4. illustrates the work stream for the described example. In the 1st step of the proposed methodology, feature extraction is performed on time-dependent vibration signals.

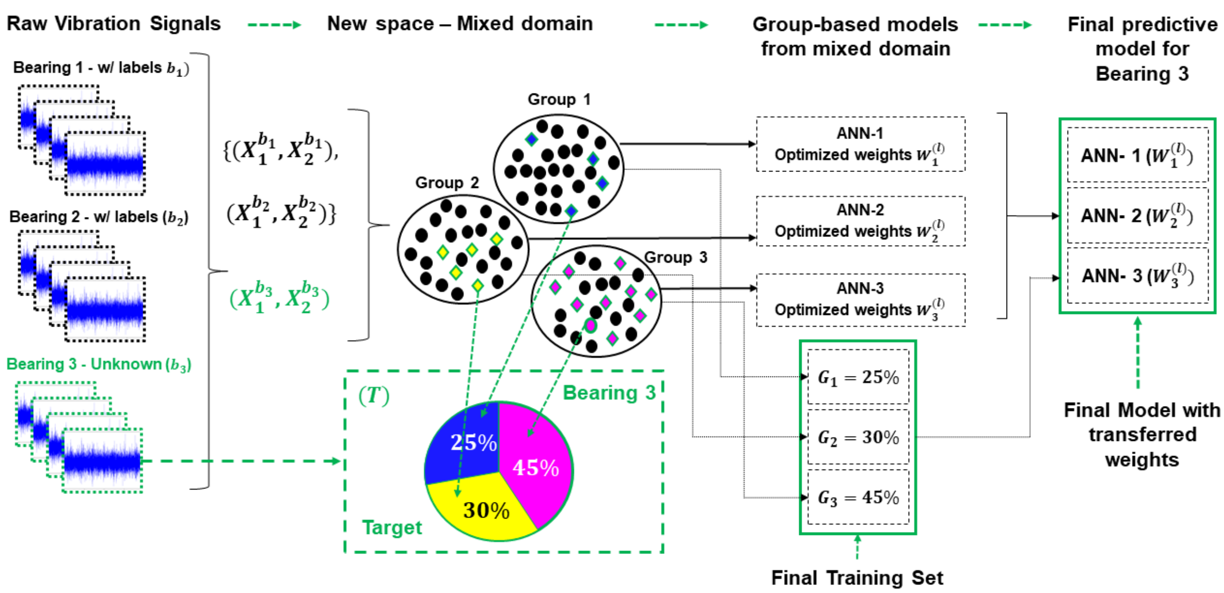

Figure 4. Detailed work stream illustration of the proposed methodology for two sources (bearing 1 and 2) and one target domain (bearing 3), clustered into three identity groups and three group-based pre-trained models.

The mixed domain is formed based on weighted samples from three different bearings. Based on assumptions, there are hidden similarities between source bearings, 1 and 2, and the target bearing 3, which can be clustered into identity groups. Let us assume that the mixed domain is grouped into three identity groups (Group 1, Group 2, and Group 3). Now, we have three identity groups with independent distribution characteristics. Then, using ordinary searching criteria, the target features are separated. Notice that bearing 1 and 2 features remain after separating the associated target bearing features in all identity groups. Then, group-based models ($ANN_{(.)}$) are built using labeled samples in each group.

Formally, as mentioned in assumptions, an optimized set of pre-trained models' parameters ($W_{(.)}^{(.)}$) can deliver the correct labels of the unlabeled targets clustered into identity groups. The optimized parameters list $W_{(1)}^{(l)}$ is correlated to the 25% partial labels of bearing 3. Likewise, parameters ($W_{(2)}^{(l)}$ & $W_{(3)}^{(l)}$) correlated to the other two portion of 30% and 45% partial labels respectively. One of the essential things to notice is that the individual groups share similar distribution characteristics. Hence, we can introduce any parametric classification method to predict the target domain partially. However, the proposed method employs transfer learning to build the final model that effectively predicts the complete target domain. The trained weights ($W_{(.)}^{(.)}$) are transplanted as customized layers in the final model to predict states of complete unlabeled bearing 3.

4. CASE STUDY: UNKNOWN STATE PREDICTION USING SIMUS-TL METHOD

The description of conducted experiments, setup, and the related results are described in this section. The performance of the proposed methodology is validated based on three case studies involving state prediction of rotating machinery. We evaluated four approaches, SVM, ANN without transfer, transfer learning method TLNN, and SiMuS-TL performance, and discussed the results for comparison and benchmarking. In a case study (A), a computer-controlled machine, "Skill boss manufacturing system (AMATROL Inc., n.d.)," was used for design experimentation. Afterward, publicly available bearing datasets, CWRU (Case Western Reserve University, n.d.) and PU (Kimotho et al., 2016), are used to validate the proposed method in case studies (B) and (C), respectively. Eventually, the relative performances of four approaches on target state prediction, given multi-sources with the same conditions and features extracted from the raw vibration signals, were demonstrated for a fair comparison. The ranking metrics for each model, area under the curve (AUC) of receiver operating characteristic (ROC) curves, are presented graphically to estimate classification accuracy for each class. The evaluation process is verified using Keras with TensorFlow libraries in the R programming software. Table 2 shows the confusion matrix and the performance measures formulated based on the confusion matrix.

	Positive Class or “1”	Negative Class or “0”
Positive Prediction or “1”	True Positive (TP)	False Negative (FP)
Negative Prediction or “0”	False Positive (FN)	True Negative (TN)

Table 3. Confusion matrix

4.1. Case study (A): Unknown state prediction on rotating machinery test bed

Our objective of this experiment is to verify the effectiveness of SiMuS-TL against SVM, ANN, and TLNN on the unknown state prediction of the target where the operating conditions are different.

4.1.1 Skill boss manufacturing and RDI technology

Skill boss manufacturing

The "Skill Boss Manufacturing" system evaluates practical knowledge required by manufacturing and production areas, particularly modeling & machine operation. It consists of an electric motor, frequency drive, interface design to incorporate operator and machine, pneumatic pick-and-place modules, quality testing and sorting metal and plastic blocks, and other features to develop skills for modern industry (AMATROL Inc., n.d.). However, in this experimentation, we consider the rotational phase of this system.

RDI technology

RDI Technologies introduces the Motion Amplification principles mechanism for detecting movement and vibration by analyzing a video (*RDI Vibration Monitoring Equipment | Motion Amplification® & Analysis*, n.d.). It motivates the industrial PdM application by detecting motion by a high-resolution camera and allows continuous monitoring of products, processes, and machinery.

4.1.2. Data description

The Skill Boss Manufacturing system operated under different motor speed conditions (425 - 1650 RPM). Randomly ordered 50 runs of the system assigned to collect three time-series data from three location bearings in the system simultaneously. The motion amplification camera was used to capture the vibration signatures of three bearings. The vibration data are recorded repeatedly with a 200Hz sampling rate. In each time-dependent vibration signal consist of 416 data points. The experimental setup of the skill boss manufacturing system is shown in Figure 5. The data sets were obtained as three independent data sets, bearing 1 - (B-1), bearing 2 - (B-2), and bearing 3 - (B-3).

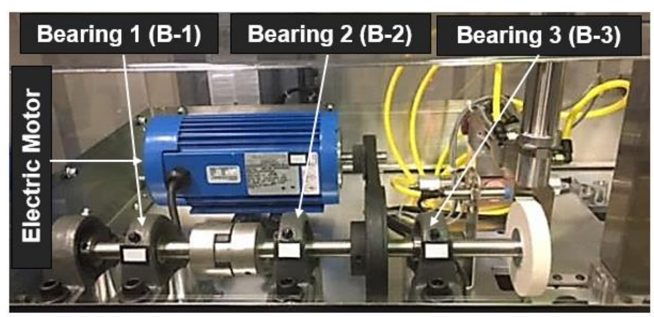

Figure 5. Experimental setup

We introduced two states for rolling element bearings in rotating machinery test beds instead of introducing bearing faults, which are installed in different locations/tasks in the system. We use bearing 1 and 2 data to predict/detect the change of states of bearing 3, which are not identical to the trained model. In detail, a model trained with bearing 1 data cannot precisely detect the changes in bearing three because bearing 1 and 3's data distribution does not match as they are installed to specific tasks. The time-dependent vibration signatures captured are shown in Figure 6.

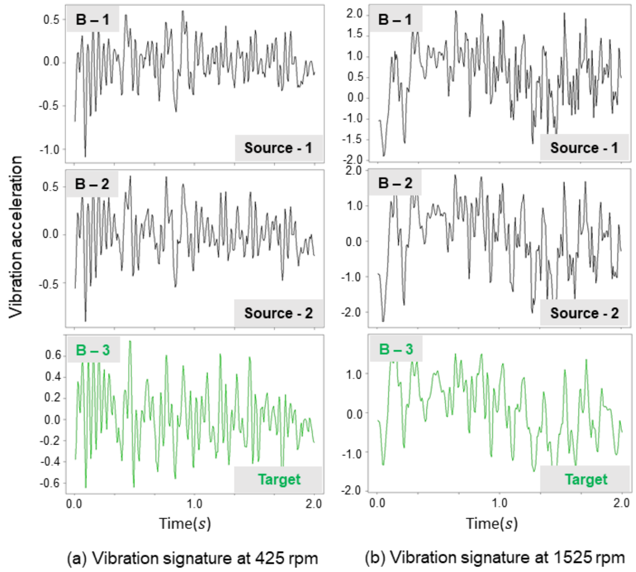

Figure 6. Vibration signals of 3 bearings at two different motor speeds (a) 425 and (b) 1525 rpm. Bearing 1 (B-1) and Bearing 2 (B-2) are multiple sources, and the bearing 3 (B-3) is the unknown target.

The next step is extracting features from the time-dependent vibration signals.

4.1.3. Feature extraction and labeling

Time-domain features and frequency-domain features can be calculated from raw vibration signals. Multiple features, such as the root means square (RMS) in the time domain and the frequency domain's spectral densities (SD), are widespread in condition-based monitoring (Caesarendra & Tjahjowidodo, 2017). Saruhan et al., (2014) determined that the RMS value can be applied in diagnosing the bearings as an indicator of the average amplitude level of vibration signals. Also, Azeem et al., (2019) effectively analyzed vibration-based power spectral density to detect the faults of rolling element bearings. Therefore, for the demonstration purpose, RMS in the time domain and SD in the frequency domain were considered in this analysis.

Labeling the feature pattern of each bearing is a challenging task because the recommended safe levels toward operating loads are different for various rolling element bearing. Individually, the design of the bearings depends on their corresponding function. In this experiment, the labeling process was done by user-defined conditions for the rolling bearings while changing the motor speeds of the system. We defined two conditions as a binary response, i.e., condition 1 – "State 1" and condition 2 - "State 2". In the experiment, vibration signals were collected from 25 different motor speeds for each condition. State 1 represented the system's rotational speed of an electric motor, 425 to 1025 RPM, and the State 2 for 1050 to 1650 RPM.

Recall that the target domain is unlabeled, and the multiple source domains are considered fully labeled. The target labels have been employed for the evaluation purpose.

4.1.4. $\boldsymbol{D_{mix}}$ grouping and finding the target sample distribution

First, standardized features from three bearings, located together with weighted samples to investigate the domain sharing based on similarities. Similar groups have been obtained by agglomerative clustering with Ward's method. It ordinarily produces compressed or even-sized groups. The grouping results of D_{mix}, such as grouped into 5,4,3, and 2 groups based on similarities, are shown in the Figure 7.

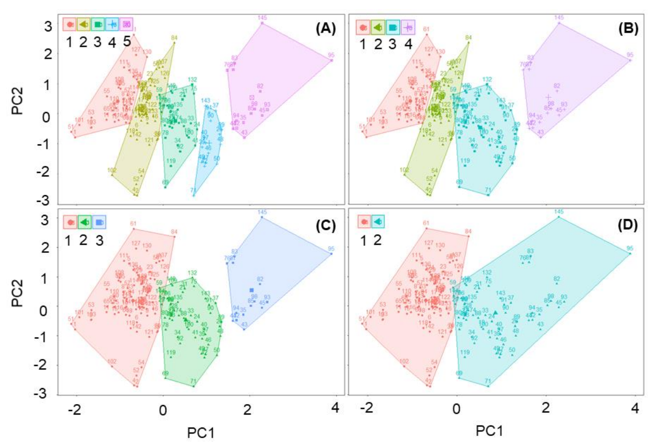

Figure 7. D_{mix} grouping results, (A) Five groups, (B) Four groups, (C) Three groups, (D) Two groups.

For example, in the trial (C), the first group shares 55.33 %, the second group shares 34.66%, and the third group shares 10% of total $\boldsymbol{D_{mix}}$. Using the indexing technique and searching through an individual group, we found the target sample distribution. The target sample distribution and domain sharing results in an individual group are shown in the Table 2. For example, in the trail (C), target sample distribution in three groups were 40%, 46%, and 14%, respectively.

Trial #	Groups in D_{mix}	$P_{G_m}(\%)$ $P_T^{G_m^*}(\%)$	G1	G2	G3	G4	G5
Trial (A)	5	$P_{G_{1,2,\ldots,5}}$	25.33	30	24	10.66	10
		$P_T^{G_5^*}$	24	16	32	10	18
Trial (B)	4	$P_{G_{1,2,\ldots,4}}$	25.33	30	34.66	10	N/A
		$P_T^{G_4^*}$	24	16	42	18	N/A
Trial (C)	3	$P_{G_{1,2,3}}$	55.33	34.66	10	N/A	N/A
		$P_T^{G_3^*}$	40	46	14	N/A	N/A
Trial (D)	2	$P_{G_{1,2}}$	55.33	44.66	N/A	N/A	N/A
		$P_T^{G_2^*}$	40	60	N/A	N/A	N/A

Table 4: Domain sharing results and target sample distribution in an individual group of mixed domain, with grouping technique Ward's method.

However, we investigated the similarity grouping with multiple dissimilarity matrices combined with different linkage methods to extend the experimentation to compare the Ward's linkage method.

Dissimilarity metric	P_{G_m} (%) $P_T^{G_m^*}(\%)$	Complete		Single		Average		Ward 1		Ward 2	
Euclidean	$P_{G_{1,2}}$ (%)	90	10	99	1	99	1	55	45	55	45
	$P_T^{G_2^*}(\%)$	82	18	98	2	98	2	40	60	40	60
Manhattan	$P_{G_{1,2}}$ (%)	90	10	99	1	90	10	55	45	52	48
	$P_T^{G_2^*}(\%)$	76	24	98	2	76	24	40	60	34	66
Chebyshev	$P_{G_{1,2}}$ (%)	90	10	99	1	90	10	73	27	55	45
	$P_T^{G_2^*}(\%)$	76	24	98	2	76	24	66	34	40	60
Canberra	$P_{G_{1,2}}(\%)$	92	8	24	76	54	46	56	44	56	44
	$P_T^{G_2^*}$ (%)	84	16	16	84	40	60	40	60	40	60

Table 5: Various dissimilarity metrics and linkage methods result for the domain sharing results and the target sample distribution concerning two similar groups in D_{mix}.

The results are shown in the Table 5. The different dissimilarities metrics and corresponding linkage methods are subjected to two groups($\boldsymbol{k=2}$) in $\boldsymbol{D_{mix}}$.

In this study, we have chosen Euclidean distance and Ward's linkage as the distance metric. The total number of elements was grouped into different groups. Purpose of demonstration and for simplicity, we step forward with two groups in the mixed domain. The two groups in the D_{mix} share similarities, approximately 55% and 45% in group 1 and group 2, respectively. Based on our findings, 40% of targets were grouped in the first group, and 60% were grouped into the second group.

4.1.5. Train two pre-trained models to find unique weights/parameters for transfer learning

The two pre-trained models are developed using the labeled samples. Recall that the labeled samples in a mixed domain adapted from multiple sources (bearing 1 & 2) are used to train the pre-trained models to predict the unknown target samples (the bearing 3).

Hence, two pre-trained models were fine-tuned until they exhibit comparable minimum loss and more precise training and testing accuracy (Table 6). Performance measures are based on 80% of training samples and 20% of the validation set.

Model	Sensitivity (SD)	Specificity (SD)	Precision (SD)	Bal. Accuracy (SD)	F1 Score (SD)	AUC (SD)
$\boldsymbol{Model_{G_1-Training}}$	0.9748 (0.0314)	0.8509 (0.0657)	0.8326 (0.0739)	0.9128 (0.0411)	0.8969 (0.0478)	0.9342 (0.0328)
$\boldsymbol{Model_{G_1-Testing}}$	0.9615 (0.0421)	0.8235 (0.0000)	0.8062 (0.0068)	0.8925 (0210)	0.8768 (0.0216)	0.9275 (0.0148)
$\boldsymbol{Model_{G_2-Training}}$	0.9091 (0.0331)	1.0000 (0.0000)	1.0000 (0.0000)	0.9545 (0.0165)	0.9521 (0.0182)	0.9826 (0.0142)
$\boldsymbol{Model_{G_2-Testing}}$	0.8333 (0.1183)	1.0000 (0.0000)	0.9481 (0.0803)	0.9166 (0.0591)	0.9054 (0.0684)	0.9992 (0.0018)

Table 6: The performances of the training and testing models developed in two groups.

In the first group, the accuracy recorded in testing samples is 89.25%, and an F1 score of 87.68%. In the second group, accuracy recorded in testing samples is 91.66%, and an F1 score of 90.54%. We obtained the trained weights/parameters from each network separately to use it in the next step, which we called transfer parameters in the final network. As stated in assumptions, we expect the above-mentioned weights have the potential of predicting the unknown labels, which were grouped into a distinct group.

4.1.6. Transfer pre-trained weights

Developing the final model to predict unknown labels of the target is primarily based on transferring weights from pre-trained models. The training samples were chosen as a weighted sample data set from two groups founded on target sample distribution. The trained weights from each pre-trained model were transplanted (randomly selected layers copied) in the final model (Oquab, Bottou, Laptev, and Sivic, 2014). However, randomly initialized layers were used to connect the incoming layers with trained weights in the final model. The transplanting took place in the reported literature (Oquab et al. 2014), and pre-trained weights do not train again. We freeze them in the final model, otherwise, we lose the benefit of using the transfer learning in this framework. We optimized the loss function and found the performance measures. The results are shown in the Table 7.

Model	Sensitivity (SD)	Specificity (SD)	Precision (SD)	Bal. Accuracy (SD)	F1 Score (SD)	AUC (SD)
$\boldsymbol{Model}_{\text{Training}}$	0.9398 (0.0219)	0.8201 (0.0317)	0.8572 (0.0196)	0.8799 (0.0105)	0.8962 (0.0082)	0.9488 (0.0323)
$\boldsymbol{Model}_{\text{Testing}}$	0.9722 (0.0481)	0.7303 (0.1097)	0.7381 (0.1229)	0.8511 (0.0309)	0.8326 (0.0565)	0.9291 (0.0863)

Table 7: Final model training and testing accuracies

In view of the results, the training accuracy of the final model is 87.99% and 85.11% over the testing sample. The F1 scores of training and testing data are 89.62% and 83.26%.

4.1.7 Performance comparison

Recall that the goal of our experiment is to verify the effectiveness of SiMuS-TL against traditional SVM, ANN (without transfer learning), and TLNN on state prediction of unknown target bearings. In the conventional setup with multi-sources with one target problem, models trained in the source domain are used to predict the target domain, namely, bearings 1 & 2 are sources, and bearing 3 is the target domain for all empirical comparisons. It should be noted that the same extracted features from sources and the target, handcrafted time and frequency features, are used to compare all methodologies relatively. Fine-tuning was introduced for optimizing ANN and TLNN models with the source domain data. In addition, grid search is used to optimize the hyperparameters for SVM. The training and testing processes were subjected to 10-fold cross-validation. The accuracy comparison results are shown in the Table 8. The receiver operating characteristic (ROC) curve has been utilized to compare the performances graphically. The comparison curves can be found in the Figure 8.

Model	Sensitivity (SD)	Specificity (SD)	Precision (SD)	Bal. Accuracy (SD)	F1 Score (SD)	AUC (SD)
ANN	0.6488 (0.1015)	0.7600 (0.1095)	0.7474 (0.0997)	0.7044 (0.0260)	0.6841 (0.0394)	0.8662 (0.0194)
SVM	0.7438 (0.0998)	0.6756 (0.0247)	0.6842 (0.0410)	0.7097 (0.0616)	0.7118 (0.0691)	0.6622 (0.0594)
TLNN	0.7201 (0.0808)	0.8066 (0.0154)	0.7666 (0.0517)	0.7691 (0.0227)	0.7258 (0.0411)	0.9077 (0.1267)
SiMuS-TL	**0.8844 (0.0858)**	**0.8711 (0.0388)**	**0.8745 (0.0281)**	**0.8784 (0.0357)**	**0.8763 (0.0399)**	**0.9688 (0.0147)**

Table 8: Case study (A) - performance comparison for predicting the states of target bearing 3

It can be observed that the proposed SiMuS-TL method performs better than the SVM, ANN and TLNN methods, with significantly higher accuracy of 87.84% and F1 score of 87.63% over the target domain. The area under curve (AUC) values show that SiMuS-TL reached the minimum FPR and FNR, proving the proposed SiMuS-TL method well-suited to fulfill the precision demands including the transferred pre-trained weights. It is understood that SVM or ANN cannot perform well in this analysis as these methods follow the universal assumption that training and testing data comply with the same probability distribution. For this experiment, bearing 1 & 2 sample distributions are not identical to the same sample distribution of bearing 3, as they are installed in different tasks/locations. Thus, the TLNN method is comparably better than conventional methods but fails to yield the accuracies of our proposed SiMuS-TL method.

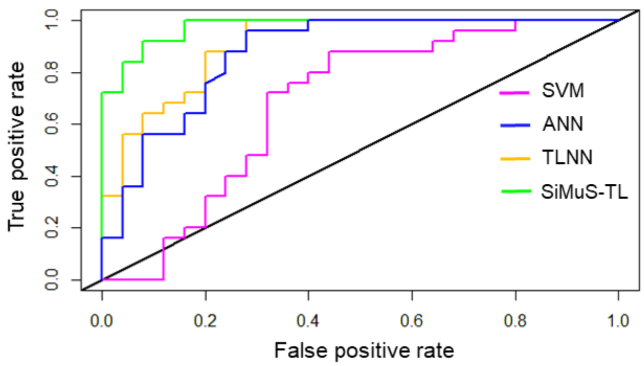

Figure 8: Case study (A) - receiver operating characteristic curve for SVM, ANN, TLNN, and SiMuS-TL methods

In this analysis, the class imbalance problem in each similar group was addressed by adjusting class weights while training the pre-trained models. Fine-tuning and transplanting custom layers explored the effectiveness of the model thoroughly. This makes it possible to have sensible results and enhance the performance of the final model classifier for state prediction on rolling element bearing in the system.

4.2. Case study (B): Proposed SiMuS-TL method with Case Western Reserve University Bearing Data

CWRU bearing data is collected from the experiment carried out in the test bed, including the electric motor, driving shaft, and dynamometer (Case Western Reserve University, n.d.).

4.2.1. Data description

In this experiment, multiple torques have been applied to generate damage in bearing components to measure the vibration signals in normal and faulty bearings. Electro-discharge machining (EDM) is used to damage these motor bearings. The EDM process introduced faults ranging from 0.007 inches in diameter to 0.040 inches in diameter at the rolling element, inner and outer race. Loads of 0 to 3 horsepower(HP) (motor speeds of 1797 to 1720 RPM) changed to collect vibration data with two sampling frequencies, such as 12k and 48k. Also, this platform has different locations of vibration data, such as the drive and fan sides of the motor.

4.2.2. Detail procedure of performance comparison

We employ these experimental combinations of motor loads and inner race faults in the different motor speeds to define states in two sources and the target. We determined the fault diameter of 0.007 inches with 0 HP motor load with 12K Hz drive end as "State 1" and combinations with different motor speeds as "State 2." Furthermore, sources 1 and 2 include fault diameter 0.007 inches with motor load 1 HP and 2HP with 12K Hz drive end bearing fault data, respectively. The target comprises an inner race fault diameter of 0.007 inches with a motor load of 3 HP, which is unknown to the trained models in this problem setup. Each state sample has a length of 1000 data points, and sources 1 and 2 data have been used to classify the unknown states in the target. Furthermore, we compare the exact setup of algorithms as detailed previously in the case study (A) to exhibit the performances of our proposed method in CWRU data. In Table 9, the performance comparison can be found. The receiver operating characteristic (ROC) curve is used to compare the performances graphically. The comparison curves can be found in the Figure 9.

Model	Sensitivity (SD)	Specificity (SD)	Precision (SD)	Bal. Accuracy (SD)	F1 Score (SD)	AUC (SD)
ANN	0.7351 (0.1140)	0.6616 (0.0832)	0.6678 (0.0641)	0.7001 (0.0501)	0.6972 (0.0738)	0.7791 (0.0844)
SVM	0.6722 (0.0404)	0.6854 (0.1271)	0.6809 (0.0883)	0.6801 (0.0622	0.6743 (0.0511)	0.7015 (0.0373)
TLNN	0.7233 (0.0521)	0.7228 (0.1260)	0.7083 (0.0597)	0.7251 (0.0467)	0.7153 (0.0255)	0.8271 (0.0536)
SiMuS-TL	**0.9900 (0.0223)**	**0.7500 (0.1172)**	**0.8038 (0.0680)**	**0.8701 (0.0570)**	**0.8856 (0.0429)**	**0.9366 (0.0129)**

Table 9. Case study (B) - performance comparison for predicting the states of target in CWRU data

It has been found that our proposed SiMuS-TL method performs better than the SVM, ANN without transfer, and TLNN transfer learning method. Unknown state prediction by SiMuS-TL records a higher accuracy of 87.01% and an F1 score of 88.56% over the target.

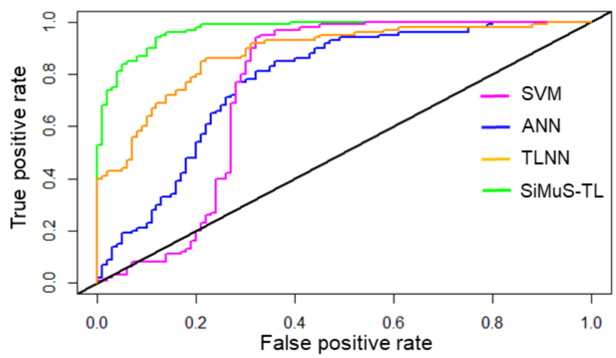

Figure 9: Case study (B) - receiver operating characteristic curve for SVM, ANN, TLNN, and SiMuS-TL with CWRU data

The AUC values show that SiMuS-TL attained the lowest FPR and FNR. The benchmarking transfer learning method TLNN offers higher accuracies than traditional methods without transferring parameters. Thus, TLNN fall against the SiMuS-TL proposed method showing comparably higher performances on the unknown target domain, revealing the proposed method well qualified to achieve a satisfactory level of accuracy.

As case study (A) mentioned, fine-tuning, transplanting custom layers, and adjusting class weights are explored while training ANNs to enhance performance accuracy.

4.3. Case study C: Proposed method SiMuS-TL with Paderborn University data

The Paderborn bearing dataset is collected based on ball bearings of type 6203. This test rig consists of a rolling bearing test module, torque-measurement shaft, motor, and a flywheel.

4.3.1. Data description

The data was generated on experiments with 32 different bearing damages, such as 6 bearings are undamaged, 12 bearings with damages artificially induced, and the rest of 14 bearings with actual damages caused by accelerated lifetime. Artificial cracks are created through electric discharge machining, drilling, and manual electric engraving. These damages are generated on the inner and outer raceway in bearings. Motor currents and vibration data have been collected for all bearing damages concurrently. Four operating conditions are experimented with in the testing platform, changing motor rpm, load torque, and radial forces. All these signals were collected as 20 measurements with a fixed length of data points. In the paper (Kimotho et al., 2016), one can find the detailed characterization of measures.

4.3.2. Detailed procedure of performance comparison

Using these experimental combinations, we selected an artificially pitted inner raceway bearing by an electric engraver in this study. The different rotational speeds of the drive system, load torque, and radial force parameters were considered while determining two sources and the target. The rotational speed of 1500 rpm, load torque of 0.7 Nm, and radial force of 1000 N are referred to as "State 1," and combinations of operating parameters with three other additional settings as "State 2." Moreover, sources 1 and 2 possess 900 rpm, 0.7 Nm, 1000 N, and 1500 rpm, 0.1 Nm, with 1000 N in the drive system, respectively. The target incorporates an inner race fault with 1500 N, 0.7 Nm, and 400 N, which is unknown to the trained models in this problem setup.

Each sample has a length of 1000 data points, and unknown target states are classified using the knowledge of learning of sources 1 and 2 data. Furthermore, we compare the exact setup of algorithms as detailed previously in case study (A) to exhibit the performances of our proposed method in PU data. In Table 10, one can find the performance comparison, and Figure10. shows the AUC ROC curves.

Model	Sensitivity (SD)	Specificity (SD)	Precision (SD)	Bal. Accuracy (SD)	F1 Score (SD)	AUC (SD)
ANN	0.5043 (0.0247)	0.8417 (0.0151)	0.7615 (0.0239)	0.6751 (0.0197)	0.6091 (0.0275)	0.7487 (0.0252)
SVM	0.5091 (0.0801)	0.6809 (0.1511)	0.6654 (0.1161)	0.6146 (0.0429)	0.5967 (0.0368)	0.7064 (0.0248)
TLNN	0.6486 (0.0998)	0.7190 (0.0247)	0.7027 (0.0410)	0.6838 (0.0616)	0.6732 (0.0691)	0.7470 (0.0594)
SiMuS-TL	**0.7931 (0.0559)**	**0.8295 (0.0485)**	**0.8251 (0.0480)**	**0.8109 (0.0521)**	**0.8087 (0.0518)**	**0.8959 (0.0257)**

Table 10. Case study (C) - performance comparison for predicting the target in PU data

The proposed SiMuS-TL method performs better than the SVM, ANN without transfer, and TLNN transfer learning method, with increased accuracy of 81.09% and an F1 score of 80.87% over the target.

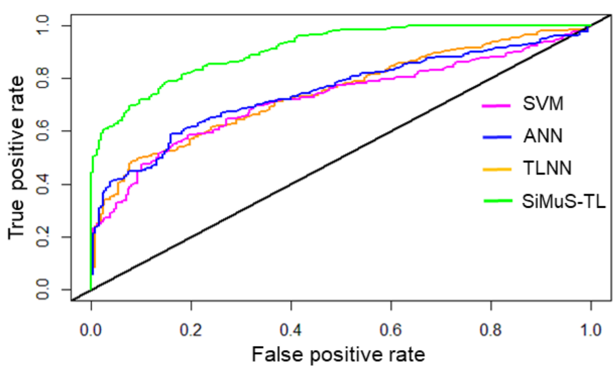

Figure 9: Case study (C) - receiver operating characteristic curve for SVM, ANN, TLNN, and SiMuS-TL methods

The AUC ROC curves show that SiMuS-TL acquired the lowest FPR and FNR scores over the target domain. The target conditions are new to the trained models, which incorporate a different distribution compared to source conditions trained to make predictions in original models. The traditional SVM, ANN, scarcely captures this difference and fails to perform well in this experiment. Thus, the transfer learning method TLNN is slightly better than conventional methods but cannot perform as effectively as SiMuS-TL in this analysis. To enhance the results, we introduced adjusting class weights to reduce the class imbalances while training and fine-tuning parameters.

4.4. Discussion

We experimented with the unknown state prediction problem, utilizing three different test datasets to investigate the multi-sources and one target problem. The target is not identical to the source, which describes that they have dissimilar distributions. In detail, the problem setup described in the case study (A), the sources, and the target bearing's locations/tasks are not identical. In case studies B & C, the operating conditions are different for sources and the target. The conventional ML algorithms, SVM, and ANN without transfer could not capture these differences, so the established classification boundaries with sources will drift when the condition changes and performance measures are degraded in the target domain. However, the transfer learning method TLNN benchmarking in this study shows comparably better results in unknown state prediction in the target as opposed to conventional methods. It reveals that transfer learning can be superior in predicting changes in conditions reasonable. However, the proposed method SiMuS-TL performs better than TLNN in all three datasets, proving the proposed similarity-based transfer learning qualified to meet the precision demands.

5. CONCLUSION AND FUTURE WORK

This paper develops an effective machinery state prediction method by classifying time series signals for PdM. Our method addresses numerous gaps in the state-of-the-art, including complex/multiple operating conditions, a lack of data samples, and conventional machine learning algorithms following a universal assumption of homogeneous distribution that is not practical in real-world scenarios yet are recurring issues. In this method, we introduced similarity-based transfer learning to combine the intelligence of multiple sources to classify time series signals with limited data availability. The significant contribution of this study can be listed as follows.

- We developed a novel state prediction method for machinery to identifying the unknown conditions of machinery in operation to encourage PdM applications.
- We addressed multi-source transfer learning instead of single-source transfer learning with one target problem.
- We introduced a new domain called "mixed domain" to identify the similarities and dissimilarities between multiple sources and the target.

- The state prediction results were compared with the conventional ML techniques SVM, ANN, and transfer learning method TLNN, to verify the effectiveness of the proposed method.
- The experimented results in various case studies, (A), (B), and (C) demonstrated that the proposed SiMuS-TL reached the maximum classification accuracies over the unknown target, which endorses the proposed method's effectiveness.

There exist two interesting issues to be addressed in future research. The proposed framework of SiMuS-TL assumes that similarities between the multiple sources and target domains are shared. If they do not share the similarities, the target samples are grouped separately from any source similarities in the mixed domain. Therefore, addressing this scenario is one of the interesting open research problems to extend the current framework in the future. In addition, we assumed that sources and the target share the same label space in this study. However, the proposed method needs some extensions if there are multiple label spaces or differences in the source and target tasks, i.e., multi-task domain problem.

We acknowledge the low interpretability of the proposed method as we introduced similarity-based transferring parameters from task-specific pre-trained models. This is one major limitation of any purely data-driven method. Therefore, further investigation is mandatory for the potential improvement concerning low interpretability. Moreover, the effect of training/validation ratio affection, compatibility of forming a specific training/validation ratio with transfer learning, and optimizing transferred layers and weights while building a final model under multi-task learning are potential improvements to be investigated in the future.

ACKNOWLEDGEMENT

This work by Mississippi State University was financially supported by the U.S. Department of Defense (DoD) High Performance Computing Modernization Program, through the US Army Engineer Research and Development Center (ERDC) Contract #W912HZ21C0014. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Army ERDC or the U.S. DoD.

REFERENCES

Afridi, M. J., Ross, A., & Shapiro, E. M. (2016). L-CNN: Exploiting labeling latency in a CNN learning framework. *Proceedings - International Conference on Pattern Recognition*, *0*, 2156–2161. https://doi.org/10.1109/ICPR.2016.7899955

AMATROL Inc. (n.d.). *Skill Boss Manufacturing: MSSC CPT Plus Certification & Assessment*. https://amatrol.com/product/skill-boss-performance-based-assessment/

Azeem, N., Yuan, X., Urooj, I., & Jabbar, J. (2019). Vibration-Based Power Spectral Density Analysis for the Detection of Multiple Faults in Rolling Element Bearings. *2019 5th International Conference on Control, Automation and Robotics, ICCAR 2019*, *April*, 719–726. https://doi.org/10.1109/ICCAR.2019.8813353

Caesarendra, W., & Tjahjowidodo, T. (2017). A review of feature extraction methods in vibration-based condition monitoring and its application for degradation trend estimation of low-speed slew bearing. *Machines*, *5*(4). https://doi.org/10.3390/machines5040021

Case Western Reserve University. (n.d.). *Bearing Data Center Seeded Fault Test Data*. https://engineering.case.edu/bearingdatacenter

Chen, W. H. (2012). Online fault diagnosis for power transmission networks using fuzzy digraph models. *IEEE Transactions on Power Delivery*, *27*(2), 688–698. https://doi.org/10.1109/TPWRD.2011.2178079

De Felice, F., Petrillo, A., & Autorino, C. (2014). Maintenance strategies and innovative approaches in the pharmaceutical industry: An integrated management system (ims). *International Journal of Engineering Business Management*, *6*(1), 1–9. https://doi.org/10.5772/59023

Donoho, D. L. (1995). *De-Noising by Soft-Thresholding*. *41*(3).

Flandrin, P., Rilling, G., & Gonçalvés, P. (2004). Empirical mode decomposition as a filter bank. *IEEE Signal Processing Letters*, *11*(2 PART I), 112–114. https://doi.org/10.1109/LSP.2003.821662

Gan, M., Wang, C., & Zhu, C. (2016). Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings. *Mechanical Systems and Signal Processing*, *72–73*, 92–104. https://doi.org/10.1016/j.ymssp.2015.11.014

Gao, Z., Cecati, C., & Ding, S. X. (2015). A survey of fault diagnosis and fault-tolerant techniques-part I: Fault diagnosis with model-based and signal-based approaches. *IEEE Transactions on Industrial Electronics*, *62*(6), 3757–3767. https://doi.org/10.1109/TIE.2015.2417501

Gideon, J., Khorram, S., Aldeneh, Z., Dimitriadis, D., & Provost, E. M. (2017). Progressive neural networks for transfer learning in emotion recognition. *Proceedings of the Annual Conference of the International Speech*

Communication Association, INTERSPEECH, 2017-Augus, 1098–1102. https://doi.org/10.21437/Interspeech.2017-1637

Guo, L., Lei, Y., Xing, S., Yan, T., & Li, N. (2019). Deep Convolutional Transfer Learning Network: A New Method for Intelligent Fault Diagnosis of Machines with Unlabeled Data. *IEEE Transactions on Industrial Electronics*, *66*(9), 7316–7325. https://doi.org/10.1109/TIE.2018.2877090

JACKSON, J. E. (1991). A User ' s Guide to Principal Components . by J . Edward Jackson Review by : Brian D. In *A Wiley-Interscience* (Issue March).

Jia, F., Lei, Y., Lin, J., Zhou, X., & Lu, N. (2016). Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical Systems and Signal Processing*, *72–73*, 303–315. https://doi.org/10.1016/j.ymssp.2015.10.025

Jin, X., Weiss, B. A., Siegel, D., & Lee, J. (2016). Present status and future growth of advanced maintenance technology and strategy in us manufacturing. *International Journal of Prognostics and Health Management*, *7*(3). https://doi.org/10.36001/ijphm.2016.v7i3.2409

Kimotho, J. K., Lessmeier, C., Sextro, W., & Zimmer, D. (2016). 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. *Third European Conference of the Prognostics and Health Management Society 2016*, *Cm*, 152–156. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1088.9087&rep=rep1&type=pdf

Li, X., Li, X., & Ma, H. (2020). Deep representation clustering-based fault diagnosis method with unsupervised data applied to rotating machinery. *Mechanical Systems and Signal Processing*, *143*. https://doi.org/10.1016/j.ymssp.2020.106825

Li, X., & Zhang, W. (2020). Deep Learning-Based Partial Domain Adaptation Method on Intelligent Machinery Fault Diagnostics. *IEEE Transactions on Industrial Electronics*, *0046*(c), 1–1. https://doi.org/10.1109/tie.2020.2984968

Li, X., Zhang, W., Ma, H., Luo, Z., & Li, X. (2020). Partial transfer learning in machinery cross-domain fault diagnostics using class-weighted adversarial networks. *Neural Networks*, *129*, 313–322. https://doi.org/10.1016/j.neunet.2020.06.014

Li, Z., Wang, K., & He, Y. (2016). *Industry 4.0 - Potentials for Predictive Maintenance*. *Iwama*, 42–46. https://doi.org/10.2991/iwama-16.2016.8

Liu, Han, Zhou, J., Xu, Y., Zheng, Y., Peng, X., & Jiang, W. (2018). Unsupervised fault diagnosis of rolling bearings using a deep neural network based on generative adversarial networks. *Neurocomputing*, *315*, 412–424. https://doi.org/10.1016/j.neucom.2018.07.034

Liu, Hongmei, Li, L., & Ma, J. (2016). Rolling Bearing Fault Diagnosis Based on STFT-Deep Learning and Sound Signals. *Shock and Vibration*, *2016*. https://doi.org/10.1155/2016/6127479

Lu, C., Wang, Z. Y., Qin, W. L., & Ma, J. (2017). Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification. *Signal Processing*, *130*, 377–388. https://doi.org/10.1016/j.sigpro.2016.07.028

Lu, S., He, Q., Yuan, T., & Kong, F. (2017). Online Fault Diagnosis of Motor Bearing via Stochastic-Resonance-Based Adaptive Filter in an Embedded System. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *47*(7), 1111–1122. https://doi.org/10.1109/TSMC.2016.2531692

Macklin, C. (2019). Measuring Generalization and Overfitting in Machine Learning. *UC Berkeley Electronic Theses and Dissertations*, 170. https://escholarship.org/uc/item/98384265

Mao, W., He, J., Li, Y., & Yan, Y. (2016). *Bearing fault diagnosis with auto-encoder extreme learning machine : A comparative study*. *0*(0), 1–19. https://doi.org/10.1177/0954406216675896

Mohan, J., Krishnaveni, V., & Guo, Y. (2014). A survey on the magnetic resonance image denoising methods. *Biomedical Signal Processing and Control*, *9*(1), 56–69. https://doi.org/10.1016/j.bspc.2013.10.007

Murtagh, F., & Legendre, P. (2014). Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion? *Journal of Classification*, *31*(3), 274–295. https://doi.org/10.1007/s00357-014-9161-z

Namuduri, S., Narayanan, B. N., Davuluru, V. S. P., Burton, L., & Bhansali, S. (2020). Review—Deep Learning Methods for Sensor Based Predictive Maintenance and Future Perspectives for Electrochemical Sensors. *Journal of The Electrochemical Society*, *167*(3), 037552. https://doi.org/10.1149/1945-7111/ab67a8

Niu, S., Liu, Y., Wang, J., & Song, H. (2020). A Decade Survey of Transfer Learning (2010–2020). *IEEE Transactions on Artificial Intelligence*, *1*(2), 151–166. https://doi.org/10.1109/TAI.2021.3054609

Niu, S., Liu, Y., Wang, J., & Song, H. (2021). A Decade Survey of Transfer Learning (2010–2020). *IEEE Transactions on Artificial Intelligence*, *1*(2), 151–166.

https://doi.org/10.1109/tai.2021.3054609

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1717–1724. https://doi.org/10.1109/CVPR.2014.222

Pan, S. J., & Yang, Q. (2009). *A Survey on Transfer Learning*. https://doi.org/10.1109/TKDE.2009.191

Paul, A., Rottensteiner, F., & Heipke, C. (2015). Transfer learning based on logistic regression. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *40*(3W3), 145–152. https://doi.org/10.5194/isprsarchives-XL-3-W3-145-2015

Perschl, F., & Schmidt, G. (1993). Model- and knowledge-based fault detection and diagnosis of gas transmission networks. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, *1*, 749–756. https://doi.org/10.1109/icsmc.1993.384834

RDI Vibration Monitoring Equipment | Motion Amplification® & Analysis. (n.d.). https://rditechnologies.com/

Samanta, B., & Al-Balushi, K. R. (2003). Artificial neural network based fault diagnostics of rolling element bearings using time-domain features. *Mechanical Systems and Signal Processing*, *17*(2), 317–328. https://doi.org/10.1006/mssp.2001.1462

Saruhan, H., Saridemir, S., Çiçek, A., & Uygur, I. (2014). Vibration analysis of rolling element bearings defects. *Journal of Applied Research and Technology*, *12*(3), 384–395. https://doi.org/10.1016/S1665-6423(14)71620-7

Selcuk, S. (2017). Predictive maintenance, its implementation and latest trends. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *231*(9), 1670–1679. https://doi.org/10.1177/0954405415601640

Shao, S., Member, S., Mcaleer, S., Yan, R., & Member, S. (2018). Highly-Accurate Machine Fault Diagnosis Using. *IEEE Transactions on Industrial Informatics*, *PP*(c), 1. https://doi.org/10.1109/TII.2018.2864759

Shen, F., Chen, C., Yan, R., & Gao, R. X. (2016). Bearing fault diagnosis based on SVD feature extraction and transfer learning classification. *Proceedings of 2015 Prognostics and System Health Management Conference, PHM 2015*, *1*. https://doi.org/10.1109/PHM.2015.7380088

Shirkhorshidi, A. S., Aghabozorgi, S., & Ying Wah, T. (2015). A Comparison study on similarity and dissimilarity measures in clustering continuous data. *PLoS ONE*, *10*(12), 1–20. https://doi.org/10.1371/journal.pone.0144059

Sun, C., Ma, M., Zhao, Z., Tian, S., Yan, R., & Chen, X. (2019). Deep Transfer Learning Based on Sparse Autoencoder for Remaining Useful Life Prediction of Tool in Manufacturing. *IEEE Transactions on Industrial Informatics*, *15*(4), 2416–2425. https://doi.org/10.1109/TII.2018.2881543

Szmrecsanyi, B. (2009). Grammatical variation in British English dialects: A study in corpus-based dialectometry. In *Grammatical Variation in British English Dialects: A Study in Corpus-Based Dialectometry*. https://doi.org/10.1017/CBO9780511763380

Thrun, S. (1997). *Multitask Learning *. 75*, 41–75.

Thrun, S., & Pratt, L. (1998). Learning to Learn: Introduction and Overview. In *Learning to Learn* (pp. 3–17). Springer US. https://doi.org/10.1007/978-1-4615-5529-2_1

Tran, V. T., Althobiani, F., & Ball, A. (2014). An approach to fault diagnosis of reciprocating compressor valves using Teager-Kaiser energy operator and deep belief networks. *Expert Systems with Applications*, *41*(9), 4113–4122. https://doi.org/10.1016/j.eswa.2013.12.026

Tsiakmaki, M., Kostopoulos, G., Kotsiantis, S., & Ragos, O. (2020). Transfer learning from deep neural networks for predicting student performance. *Applied Sciences (Switzerland)*, *10*(6). https://doi.org/10.3390/app10062145

Tzafestas, S. G., & Dalianis, P. J. (1994). Fault diagnosis in complex systems using artificial neural networks. *Proceedings of the IEEE Conference on Control Applications*, *2*, 877–882. https://doi.org/10.1109/cca.1994.381206

Vakharia, V., Gupta, V. K., & Kankar, P. K. (2015a). Ball Bearing Fault Diagnosis using Supervised and Unsupervised Machine Learning Methods. *The International Journal of Acoustics and Vibration*, *20*(4). https://doi.org/10.20855/ijav.2015.20.4387

Vakharia, V., Gupta, V. K., & Kankar, P. K. (2015b). Ball Bearing Fault Diagnosis using Supervised and Unsupervised Machine Learning Methods. *The International Journal of Acoustics and Vibration*, *20*(4). https://doi.org/10.20855/ijav.2015.20.4387

Xia, M., Li, T., Liu, L., Xu, L., & de Silva, C. W. (2017). Intelligent fault diagnosis approach with unsupervised feature learning by stacked denoising autoencoder. *IET Science, Measurement and Technology*, *11*(6), 687–

695. https://doi.org/10.1049/iet-smt.2016.0423

Xia, M., Li, T., Xu, L., Liu, L., & De Silva, C. W. (2018). Fault Diagnosis for Rotating Machinery Using Multiple Sensors and Convolutional Neural Networks. *IEEE/ASME Transactions on Mechatronics*, *23*(1), 101–110. https://doi.org/10.1109/TMECH.2017.2728371

Xu, G., Liu, M., Jiang, Z., Shen, W., & Huang, C. (2019). Online Fault Diagnosis Method Based on Transfer Convolutional Neural Networks. *IEEE Transactions on Instrumentation and Measurement*, *PP*, 1–12. https://doi.org/10.1109/tim.2019.2902003

Xu, Y., Sun, Y., Liu, X., & Zheng, Y. (2019). A Digital-Twin-Assisted Fault Diagnosis Using Deep Transfer Learning. *IEEE Access*, *7*, 19990–19999. https://doi.org/10.1109/ACCESS.2018.2890566

Zhang, R. A. N., Tao, H., & Wu, L. (2017). *Transfer Learning With Neural Networks for Bearing Fault Diagnosis in Changing Working Conditions*. *5*, 14347–14357.

Zhang, X., Yu, F. X., Chang, S.-F., & Wang, S. (2015). *Deep Transfer Network: Unsupervised Domain Adaptation*. http://arxiv.org/abs/1503.00591

Zhang, Y., Hu, X., & Fang, Y. (2010). Logistic regression for transductive transfer learning from multiple sources. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *6441 LNAI*(PART 2), 175–182. https://doi.org/10.1007/978-3-642-17313-4_18

Zhu, X. (2008). *Semi-Supervised Learning Literature Survey Contents*.

BIOGRAPHIES

Ayantha Senanayaka is a Ph.D. student in the Industrial and Systems Engineering Department at Mississippi State University. He received his M.Sc. degree in Statistics at Mississippi State University and his bachelor's degree in Mechatronic Engineering at the University of Wolverhampton UK. His research interests focus on predictive maintenance, condition monitoring, additive manufacturing, and data mining using machine learning techniques. He is a member of the Institute for Operations Research and the Management Sciences (INFORMS).

Abdullah Al Mamun is a current Ph.D. student in the Department of Industrial & Systems Engineering (ISE) at Mississippi State University. Prior to his current Ph.D. studies, he has received a M.S. degree in Manufacturing Engineering Technology at Minnesota State University Mankato (MNSU) and a B.S. degree in Textile Technology at the University of Dhaka, Bangladesh. Mr. Abdullah's research interest focuses on Data-driven Artificial Intelligence in Additive Manufacturing, Sensor fusion for real-time Condition Monitoring of manufacturing machinery and Energy consumption optimization in advanced manufacturing systems. His Ph.D. and M.S. research works are being published in renowned peer-reviewed journals and proceedings, including Journal of Manufacturing Systems, Journal of Intelligent Manufacturing, International Journal of Computer Integrated Manufacturing, the American Society of Mechanical Engineers, and Procedia Manufacturing.

Dr. Glenn Bond is a computer scientist with the US Army Engineer Research and Development Center's Information Technology Laboratory. His research interests include high performance data analytics, prognostic and predictive maintenance (PPMx), decision-making support, and dynamic data mining.

Dr. Wenmeng Tian is currently an assistant professor with the Department of Industrial and Systems Engineering at Mississippi State University. Her research interests include process monitoring and machine learning for advanced manufacturing applications.

Dr. Haifeng Wang is currently an Assistant Professor with the Department of Industrial and Systems Engineering, Mississippi State University, Starkville, MS, USA. His research interests include ensemble decision-making, deep neural network optimization, and machine learning interpretability with applications in computer-aided diagnosis, precision agriculture, and manufacturing.

Ms. Sara Fuller is currently the Strategic Programs Manger at the Center for Advanced Vehicular Systems at Mississippi State University. Her research interests include prognostic and health management (PHM) and advanced maintenance strategies and applications within small and medium manufacturers.

Mr. T. C. Falls is the Associate Director, Software Design and Development for the Institute for Systems Engineering Research (ISER). He provides technical leadership to the ISER for software design, development, architecture, implementation and integration. His research interest include data fusion, management, analysis and visualization primarily in support of advanced maintenance.

Dr. Shahram Rahimi is a Professor and Head at the Department of Computer Science and Engineering at Mississippi State University. He is a recognized leader in artificial and computational intelligence with over 200 peer-reviewed publications and a few patents or pending patents in this area.

Dr. Linkan Bian is a Thomas B. & Terri L. Nusz Endowed Associate Professor in Industrial and Systems Engineering Department at Mississippi State University. He received his Ph.D. in Industrial and Systems Engineering from Georgia

Institute of Technology, and B.S. in Applied Mathematics from Beijing University. Dr. Bian's research interests focus on the analytics of Big Data generated from complex engineering systems. His work has been widely recognized in the Industrial and System Engineering professional communities. He received the Outstanding Young Investigator Award from Institute of Industrial and Systems Engineering (IISE), as well as multiple Best Paper Awards.