

Transfer Learning For Predictive Maintenance: A Case Study

A Thesis Presented for the
Masters of Science
Degree

The University of Tennessee, Knoxville

Colter Augustyn Swanson

August 2024

© by Colter Augustyn Swanson, 2024
All Rights Reserved.

This thesis is dedicated to my parents, Greg Swanson and Carolyn Rezler, for their continued support and motivation in my academic journey.

Acknowledgments

I would like to start by thanking my academic advisor, Dr. Xueping Li. His efforts and support made this journey go from dreams to reality. Without his support, I would not be in the position I am today, and for that, I am eternally grateful. I would also like to give thanks to my two other committee members, Dr. Tom Berg and Dr. Bing Yao. Dr. Berg was the first professor at the University of Tennessee, Knoxville to give me a chance and provide me with supplementary work to feed my desire to learn. Dr. Yao was one of my first professors in the Industrial Engineering department when I made my switch from Computer Science. She made me feel as though I belonged there and never hesitated to provide me with a helping hand when I was struggling. The continued support of these three professors allowed me to focus on my studies and make strong contributions to the projects and research that I was a part of. I am eternally grateful to have encountered all three of them.

I would also like to thank my brother, Kimon Swanson. Kimon was the one who initially guided me toward Dr. Li's research team and Industrial Engineering. Working alongside him has been a blessing and has made me a better person. Without him, I would not be where I am today. Thank you, Kimon. I would also like to thank my sister, Reilly Swanson, who continued her education as well by going for her Masters. Her dedication and desire to learn have motivated me to finish strong and move forward confidently. Last, but not least, I want to thank my parents, Carolyn Rezler and Greg Swanson. Both are alumni of the Tickle College of Engineering and have guided and supported me diligently through my academic journey. Knowing that I have them in my corner has allowed me to tackle complex problems with confidence. I am very fortunate to be able to call these two my parents. I am forever grateful that I was blessed with them as my parents.

Abstract

In light of recent strides in high-performance computing, the concept of transfer learning has emerged as a prominent paradigm within the realm of Artificial Intelligence and Machine Learning methodologies. Analogous to the human brain's capacity to assimilate information across related domains for pattern recognition, transfer learning has swiftly asserted its dominance, particularly in deep learning applications such as image classification and natural language processing. Despite its ascendancy in these domains, there exists a lack of comprehensive investigations in alternative domains, notably those encompassing tabular data formats. This thesis seeks to redress this gap by conducting an empirical examination of transfer learning within the context of predictive maintenance. The study employs a case study methodology involving induced misalignments in various levels of horsepower motors. A comparative analysis is undertaken, juxtaposing transfer learning techniques against standard deep learning approaches, with the aim of elucidating the overarching efficacy and viability of transfer learning within the domain of predictive maintenance. This empirical exploration contributes to the broader discourse on the applicability and optimization of transfer learning methodologies across diverse domains, specifically addressing its efficacy in the nuanced context of predictive maintenance for machinery.

Table of Contents

1	Introduction	1
1.1	Description and Motivation	1
1.2	Applications	2
1.3	Basic Research	2
1.3.1	Economic Impact	2
1.3.2	Industrial Impact	3
1.4	Research Challenges	4
1.5	Outline of Thesis	5
2	Related Work and Background	6
2.1	Predictive Maintenance	6
2.1.1	Traditional Approaches	7
2.1.2	Deep Learning Approaches	9
2.2	Transfer Learning	12
3	Problem Definition and Approach	15
3.1	Problem Definition	15
3.1.1	Data Quality and Availability	15
3.1.2	Time and Compute Costs in Conventional Training	17
3.2	Approach	17
3.2.1	Transfer Learning	18
3.2.2	Deep Learning Models	20

4	Dataset, Implementation, Results	22
4.1	Data Analytics	27
4.1.1	Pre-Processing	27
4.1.2	Exploratory Data Analysis	34
4.2	Implementation Details	45
4.3	Network Structures	45
4.3.1	Model Input	47
4.3.2	Evaluation Criteria	47
4.4	Results	49
4.4.1	MLP	50
4.4.2	CNN-LSTM	58
5	Summary and Future Work	68
5.1	Summary	68
5.2	Limitations & Future Work	70
	Bibliography	72
	Vita	78

List of Tables

4.1	Levels of Misalignment introduced.	23
4.2	Features derived from the data.	28
4.3	Sample sizes of each class of Offset Classification.	36
4.4	Key statistics of Probe 6 : HighFrequency grms.	36
4.5	Key statistics of Probe 3 : DerivedPeak g.	38
4.6	MLP network structure.	46
4.7	CNN-LSTM network structure.	48
4.8	Accuracies for Zero-Shot technique using an MLP.	51
4.9	Accuracies for Zero-Shot technique using a CNN-LSTM.	59

List of Figures

4.1	Belt-driven motor test bed setup.	23
4.2	Sensor placement on motor housing.	24
4.3	Illustration of Offset Misalignment.	26
4.4	Illustration of Angular Misalignment.	26
4.5	Feature showing startup and shutdown outliers present in the data.	30
4.6	Feature after outlier removal technique was applied.	32
4.7	3 HP (Left) and 5 HP (Right) Probe 3 Derived Peak Box plots.	38
4.8	3 HP (Left) and 5 HP (Right) Probe 3 Derived Peak.	40
4.9	3 HP (Left) and 5 HP (Right) Probe 6 High-Frequency Box plots.	40
4.10	3 HP (Left) and 5 HP (Right) Probe 6 High Frequency.	40
4.11	Heatmap of 3HP and 5HP Vibrational Data correlations.	42
4.12	Heatmap of 3HP and 5HP MCSA Data correlations.	42
4.13	3 HP (Left) and 5 HP (Right) Probe 1 Raw Waveform.	44
4.14	MLP accuracy using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.	53
4.15	MLP training time using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.	53
4.16	MLP accuracy using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.	54
4.17	MLP training time using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.	54
4.18	MLP accuracy using incremental training data approach with a 3 HP base model and 5 HP transfer learning using MCSA data.	56

4.19	MLP training time using incremental training data approach with a 3 HP base model and 5 HP transfer learning using MCSA data.	56
4.20	MLP accuracy using incremental training data approach with a 5 HP base model and 3 HP transfer learning using MCSA data.	57
4.21	MLP training time using incremental training data approach with a 5 HP base model and 3 HP transfer learning using MCSA data.	57
4.22	CNN-LSTM accuracy using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.	61
4.23	CNN-LSTM training time using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.	61
4.24	CNN-LSTM accuracy using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.	62
4.25	CNN-LSTM training time using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.	62
4.26	CNN-LSTM accuracy using incremental training data approach with a 3 HP base model and 5 HP transfer learning on the raw waveform data.	64
4.27	CNN-LSTM training time using incremental training data approach with a 3 HP base model and 5 HP transfer learning on the raw waveform data.	64
4.28	CNN-LSTM accuracy using incremental training data approach with a 5 HP base model and 3 HP transfer learning on the raw waveform data.	65
4.29	CNN-LSTM training time using incremental training data approach with a 5 HP base model and 3 HP transfer learning on the raw waveform data.	65
4.30	CNN-LSTM accuracy and training time using incremental training data approach with a 3 HP base model and 5 HP transfer learning using MCSA data.	67
4.31	CNN-LSTM accuracy and training time using incremental training data approach with a 5 HP base model and 3 HP transfer learning using MCSA data.	67

Chapter 1

Introduction

1.1 Description and Motivation

Since the rise of the Industrial Revolution, the automation of many processes through the use of machines and other process-enhancing techniques has become a mainstay in the industrial domain. With this rapid development, issues in downtime and fault management have been increasing at an exponential rate. Techniques such as reactive and preventative maintenance were initially introduced, however, these techniques still caused unexpected and long down times that could hamper processes that industries relied on to keep their supply chains moving efficiently. [1] With the recent advancements in big data acquisition and machine learning/artificial intelligence, a new maintenance domain has been developed, predictive maintenance. Predictive maintenance utilizes big data and machine learning to detect irregularities in industrial equipment such that maintenance can be scheduled and performed before an issue has the chance to interrupt the process. Predictive maintenance techniques have shown great potential in their ability to reduce down times, reduce repair costs, and improve safety. As well as these benefits, predictive maintenance has the ability to improve the overall function of these systems through the optimization of resource allocation, better overall decision making, and inventory management. [46]

With the development of predictive maintenance, many industrial processes have been made to be more efficient, [46] however, issues still persist with this new technique. Limitations in data reduce the efficiency of predictive maintenance. This type of data is

also typically difficult and costly for companies to produce since it requires introducing faults into already working machines for the sake of collecting data. For this reason, many companies cannot/will not spend their time and resources on this type of data.

This thesis presents techniques to combat these issues through the use of transfer learning through a case study on motors with different levels of horsepower. It provides an overview of transfer learning techniques in which a similar machine may not have substantial fault data to train a new deep learning model. This is compared to using other models with a small data set to show the benefits of this technique.

1.2 Applications

Predictive maintenance is a growing field of research that has strong economic and industrial impacts. These applications have a wide range. Since most manufacturers, businesses, universities, households, etc. require the use of machines, this discipline of research can be applied in almost any system based area. The main application benefits can be broken down into two categories: economic impact and industrial impact. These are explored in the following sections.

1.3 Basic Research

1.3.1 Economic Impact

The economic impact of predictive maintenance is one of the most attractive aspects of this advancing technology. With the potential for large economic benefits after implementation, predictive maintenance's main drawback is its large initial implementation cost. This has been the focus of countless studies to understand whether or not predictive maintenance approaches are actually worth the initial cost for most companies. Adams et al. [1] directly investigated this topic by developing a system for manufacturing companies that analyzed the cost-benefit analysis of implementing different maintenance techniques. The research showed in a case study that over the course of a three-year period, predictive maintenance-based techniques outperform reactive maintenance-based techniques in their return on investment

(ROI). Subsequently, Meng et al. [24] also analyzed predictive maintenance-based techniques using ROI. Their research using Lithium Ion batteries showed that initial investment was returned in less than two years, and after ten years, the ROI increased to around three and a half of the investment, thus showing that predictive maintenance systems have the capacity to obtain considerable ROI in short time periods. Other studies showed that not only with time did ROI increase, but also with the number of failures. Chang et al. [7] evaluated the use of predictive maintenance on the new LED light bulbs, which were inclined to fail often. Their case study showed an exponential growth in ROI with respect to the number of failures that occurred. The economic impact that predictive maintenance has garnered from research is promising and shows why this field is growing in the direction it is.

1.3.2 Industrial Impact

The industrial impact of predictive maintenance is, in truth, the main component leading to the advancement of the field. Its increase in efficiency and reduction of downtime is what leads to the attractiveness of its economic impact. Thus, without the industrial impact of predictive maintenance, the economic impact would not be substantial. Several studies attempt to display these industrial impacts through multiple mediums. In 2005, Huang* et al. [19] developed a system that was used to be able to predict when an elevator door was to become defective. They used a discrete event simulation in *Anylogic* to test the system's ability to reduce downtime. Their study showed that the system was capable of reducing downtime significantly, in which money and labor hours were saved. Fischer et al. [14] noticed the demand for efficient production of semiconductors and used a case study to prove the usefulness of predictive maintenance in changing good production lines to world-class ones. Another development with industrial implications that has come from predictive maintenance is the Artificial Intelligence assistant. Shin et al. [33] displayed how this technology has assisted inspectors' ability to inspect renewable energy systems. They showed that it allowed general inspectors to perform at the same level as specialists and that it was able to improve the time efficiency and specificity of specialists. These industrial impacts of predictive maintenance give way to the rise of new methods regarding deep learning and predictive maintenance.

1.4 Research Challenges

Machine maintenance data is subject to many different challenges in deep learning, which are still under ongoing research:

- *Not enough fault data.* Predictive maintenance using DL/ML often needs large data sets to be able to accurately capture trends in the data. With most data being collected in research labs, most models are not suitable for real-time production implementation. Most data produced using real-life production machines is not fault data since faults usually occur over a long period of time. Fault data is hard to come by because it requires intentionally inducing fault so that there can be labeled data. Therefore, most data found from actual real-life situations and not research situations comes unlabeled and usually does not contain fault data or failure data.
- *Data collection is expensive and time-consuming.* It is difficult to be able to have enough data to be able to truly train deep learning models since most DL models need thousands of samples to be able to accurately train their models. This cost can be harsh if one is producing their own data.
- *Data varies based on machine type.* Since there is a plethora of machines that need to be analyzed to keep most systems running well, the overall amount of data needed is very large. In most cases it is hard to receive data that is generalizable for multiple different machine instances. Sometimes even machines that are the same but work on different tasks can have significantly different data trends.

This thesis attempts to address these challenges by developing a technique using deep learning and transfer learning for predictive maintenance use. By harnessing the momentum of transfer learning, a general predictive maintenance case study was performed that showed the benefits of including transfer learning in predictive maintenance ventures.

1.5 Outline of Thesis

The remainder of the thesis will follow the following structure: Chapter 2 will review and discuss related works that have contributed to the grounds for this research. Chapter 3 will define the problem and provide a description of the approaches used to combat these problems. Chapter 4 will include a description of the dataset used, the pre-processing techniques applied to the dataset, the implementation of the solutions, and the results of the case study. Chapter 5 will provide a summary of the findings and will discuss the future direction of this research.

Chapter 2

Related Work and Background

2.1 Predictive Maintenance

The emergence of Industry 4.0 has heralded a new era of profound industrial advancements, accompanied by the concurrent development of innovative frameworks and methodologies for equipment and process maintenance [10]. One particularly influential and transformative method that has gained prominence in this recent wave of innovation is Predictive Maintenance (PdM). In the pre-Industry 4.0 landscape, maintenance management techniques were primarily categorized into two conventional approaches: the run-to-failure model, where equipment is operated until it breaks down, and preventive maintenance, which involves scheduled maintenance at fixed intervals [34].

However, the advent of PdM has revolutionized the maintenance paradigm, leveraging the advancements in big data analytics and machine learning (ML). This sophisticated approach empowers industries to deploy ML models and other analytical tools to proactively forecast machine failures and identify their root causes [37, 27, 45]. Notably, PdM presents a compelling solution that not only conserves valuable resources but also simplifies complex maintenance tasks, ensuring a seamless and efficient maintenance workflow.

The backbone of PdM lies in its adept utilization of the abundant data available within modern industrial environments. By ingesting and analyzing data from various sources, such as sensor readings, historical maintenance records, and even external factors like environmental conditions, PdM algorithms can discern patterns, correlations, and

anomalous behavior. Armed with these data-driven insights, predictive models can be finely tuned, enabling highly accurate predictions regarding the timing and nature of potential machine failures. This foresight empowers maintenance teams to adopt a proactive stance, executing targeted interventions well in advance and preventing unexpected downtimes, costly breakdowns, and production halts [45, 41].

2.1.1 Traditional Approaches

A review of the literature associated with predictive maintenance shows most approaches use simple machine-learning models for classification or regression tasks. A recent systemic review of these scholarly papers by Carvalho et al. [6] showed that around 73% of papers used a variation of these three techniques: Random Forest (33%), Support Vector Machines (25%), and K-Means clustering (15%). Compared to deep learning techniques, these techniques are used more often because of their ease of implementation and the fact that they typically require less data. These techniques, however, are more susceptible to small changes in data when compared to deep learning models.

Random Forests are the most popular reviewed technique. [6] They work through a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. [5] There can be some confusion between Random Forest and standard decision trees. The main difference between the two comes in the fact that a Random Forest generates decision trees at random, thus Random Forests use decision trees. [6] This, in turn, reduces the amount of overfitting that is typically associated with decision trees by the randomness of the Random Forest. Hsu et al. [18] applied the Random Forest technique to wind turbine fault data in Taiwan. Using K-fold cross-validation, they were able to produce an accuracy of 91.98% using a Random Forest. From their research, they showed that faults can be predicted and maintenance can be performed in a reasonable time frame using Random Forest. Kulkarni et al. [20] used a similar method with Random Forest; however, they based their case study on predicting the presence of issues in cold storage systems using only temperature data. Their research resulted in an accuracy of 89% for their predictive maintenance venture, displaying

again that Random Forest is a top technique for the application of predictive maintenance to real-world industrial projects.

Another popular Machine Learning technique used for predictive maintenance is Support Vector Machines (SVM). SVM's are a common place for most classification problems as they are exceptionally known for producing high accuracies by finding the best place to separate the data into different classes. SVMs work by projecting the data to a high-dimensional feature space and then creating a hyperplane that accurately separates the space into different classes of data. [6] Although SVMs are very popular for predictive maintenance, they are not always the most suitable technique. SVMs struggle with noisy data and larger data sets which are common when working with predictive maintenance. This, however, has not detracted from the current research into this domain. Gohel et al. [15] proposed a technique for using SVMs to accurately predict when maintenance will need to be done on a nuclear reactor's infrastructure. Many different tests were run regarding this problem using the SVM. All tests reported an accuracy of 90% showing the SVM's robust capabilities. Praveenkumar et al. [29] developed an SVM model that was used to identify failures in automotive transmission boxes. The gearboxes were tested using different speed and load conditions. From these tests, vibrational features were extracted to train the SVM. The results showed that the SVM was able to classify failures with a precision of greater than 90%.

K-means clustering is another traditional Machine Learning technique often used in predictive maintenance research ventures. Its ease of implementation, ability to handle larger data sets, and adaptability to the introduction of new data make it a popular unsupervised machine learning technique for predictive maintenance. [6] K-means clustering works by using the Euclidean distance from each data point to the mean of all k clusters and then determining which cluster the point belongs to by which cluster's mean it is closest to. This technique has a strong approach to being able to pick up on similarities of unlabeled data, making it an ideal choice for unsupervised predictive maintenance projects. Durbhaka and Selvaraj [12] employed a K-means clustering model to analyze the behavior of wind turbines by using vibration signal analysis. The research was able to group similar behaviors such that experts could determine what signals showed faulty behavior. Eke et al. [13] produced similar research. They applied k-means clustering to extract groups from a dissolved gas data

set captured from an insulating oil of a transformer. The research was able to determine the key characteristics of each cluster such that the characteristics that induced a fault or alert were able to be discovered. This, in turn, leads to the ability to label the data as fault-inducing or not.

2.1.2 Deep Learning Approaches

Deep learning techniques for predictive maintenance have been on the rise; however, they are still a small subset of the overall predictive maintenance scene, consisting of only 27% of papers according to Carvalho et al. [6] These techniques have come to fruition from their robustness, their ability to learn complex behavior, and their ease of employment. They also have the ability to outperform standard machine learning and statistical learning methods when there are large quantities of historical data present. [32] Although deep learning techniques have become a prominent player in the field of predictive maintenance, there are still some drawbacks that researchers have to consider. Two of the notable drawbacks are the deep learning technique's training times and the notion that they are black boxes. Their training times are typically much longer than that of standard Machine learning models, making them less attractive when updating the model with new information. They are also considered black boxes in the sense that it is typically impossible to understand why they made a certain decision, i.e. there is no explainability of the model.

In the field of predictive maintenance, deep learning techniques are typically used in two different settings: Supervised Binary/Multi-class tasks or Semi-supervised One-class/Unsupervised tasks. [32] Supervised Binary/Multi class tasks refer to tasks in which the data is labeled into two or more classes. Semi-supervised One class/Unsupervised tasks refer to tasks in which either only correct data is labeled, or no data is labeled at all. [32] The most prominent deep learning techniques applied in Semi-supervised One-class/Unsupervised tasks are Recurrent Neural Networks (RNNs) and Auto encoders (AEs). RNNs are a type of Neural Network that are designed to handle sequential data and maintain a memory of the previous data. This helps to be able to capture underlying trends in time series data or other ordered data. RNNs often suffer from the vanishing gradient problem, thus the Long-Short Term Memory (LSTM) model was developed to address this issue. LSTMs use specialized

internal mathematical mechanisms to help learn long-term dependencies, thus eliminating the problem of losing memory too quickly. [43]

Aydin and Guldamlasioglu [3] used an LSTM to predict the state of an engine to catch a failure before it occurred. They fed the LSTM large quantities of real-time sequential data, showing the LSTM’s capability to be a high-scale network. Their results were promising, with 85% accuracy. Nguyen and Medjaher [26] used an LSTM in their complete predictive maintenance framework that began at the prognostic step and went all the way to making a decision on what maintenance needed to be carried out. They used the Turbo Engine Fan Data set from NASA as their case study. They benchmarked their technique versus classical periodic maintenance and ideal predicted maintenance approaches and displayed that the LSTM framework was superior to other non-AI maintenance methods. Yuan et al. [44] also used an engine data set to predict fault location and Remaining Useful Life. Their research showed that the LSTM outperformed the standard RNN and mirrored SVM fault detection accuracy while also providing RUL calculation.

With the mass amounts of raw data that are typically seen in predictive maintenance ventures, there comes a need to discover only which parts of the data are truly important to the task at hand. Instead of hiring experts to feature engineer raw data into useful features, researchers have been turning to Auto Encoders (AEs) more and more. AEs work by taking in data and developing an encoded version of it that removes noise and retains the important parts of the data. They are then able to decode this encoded data back into the input. This technique is a form of dimension reduction and saves time and effort for researchers. Roy et al. [30] found that predictive maintenance ventures had, up until then, used raw data but found that feature extraction was becoming more popular. They determined that using handcrafted features limited the feature subset to only those that the expert was knowledgeable about. Thus, they proposed an AE feature extraction layer to their Neural Network, which allowed them to achieve 100% detection accuracy using the NASA-bearing data set. Tao et al. [35] showed the capability of multiple AE layers in a Neural Network to improve training. They designed a Network consisting of two AE layers and a softmax regression layer. After the first AE layer, the training loss fluctuated heavily; however, with the addition of a second layer, the training loss converged smoothly to a global

minimum. This showed a promising result in the reduction of training loss leading to an eventual testing accuracy of around 93% for bearing fault diagnosis.

The typical structure of Supervised Binary/Multi class tasks, determined by Serradilla et al. [32], was that of Artificial Neural Networks (ANN). ANN is an umbrella term that covers all networks that are inspired by the biological structure of the brain in which neurons pass information between themselves and output decisions. A common type of ANN used for this type of task is a Multi-Layer Perceptron (MLP). MLPs are a standard form of ANN, which consists of an input layer, any number of hidden layers, and an output layer, all fully connected. MLPs are often used as a baseline ANN but also have the ability to outperform other models. Al-Raheem and Abdul-Karem [2] sought to classify the faultiness of bearings using time domain vibration signals of a rotating machine. They processed the data using Laplace wavelet analysis for feature extraction and then determined the number of hidden nodes using a Genetic algorithm. They compared the MLP with a Radial Basis Function (RBF) and a Probabilistic Neural Network (PNN) with the minimization of Mean Squared Error (MSE) as the evaluation criteria. They found that the MLP produced the best results with the second fastest training time, making it the best model for their task. As well as their high performance, MLPs are applicable to almost any deep learning task. Ullah et al. [38] determined that the increase in the internal temperature of electrical instruments would be detrimental to their operation. They used thermal images captured from infrared cameras, which were statistically processed into eleven features as data. Their MLP model performed at an accuracy of 84%.

Another ANN model that has been used for this type of predictive maintenance task is the Convolution Neural Network (CNN). CNNs are best known for their ability to analyze grid like data such as images, however, they can also be used in tabular data cases. They work by using a convolutional layer that applies filters to recognize different patterns in the data and pooling layers that downsample the data while retaining the relevant information for the task. CNNs are often used for their success in feature extraction. This is well shown by the research produced by Li et al. [21]. They used the C-MAPSS data set for aero-engine unit prognostics. The only pre-processing they did to the data was normalization. After that, they input the data into five different models. Their results showed that for

Remaining Useful Life (RUL) calculation, a deep CNN was able to outperform a standard Neural Network (NN), a Deep Neural Network (DNN), a Recurrent Neural Network (RNN), and a Long-Short Term Memory (LSTM) model. CNN’s ability to perform tasks on data that is not strictly grid-like is shown in Wen and Keyes [40]. They proposed a time series segmentation approach based on CNN’s for anomaly detection.

2.2 Transfer Learning

The idea of transfer learning in the computational sense has come along rather recently. From this, there is still a large amount of research to be done to solidify the overall definition of transfer learning. At the moment, there exist many different forms of transfer learning such that the overarching term transfer learning becomes almost ambiguous when referred to in most cases. From this, it is important to understand some key distinctions between the different forms of transfer learning.

To start, transfer learning can be broken down into three main label-setting categories: Transductive, Inductive, and Unsupervised [11, 9, 28, 4]. Transductive transfer learning is the most common form of transfer learning according to Azari et al. [4], consisting of 67% of reviewed transfer learning literature. Transductive transfer learning refers to the type of transfer learning where the task is the same for both the source and target domains [4, 28]; however, the domains may differ and often do. It is also stated that Transductive transfer learning needs unlabeled data in the target domain, but certain reviews such as Pan and Yang [28] argue that this is not a necessity for it to be considered Transductive transfer learning [4]. The label-setting for this thesis follows the Transductive approach based on Azari et al. [4] conclusion. Inductive transfer learning refers to the type of label-setting that contains differing tasks for the transfer learning approach regardless of whether the domains are different or not [4, 28]. Inductive Learning consists of 32% of papers reviewed by Azari et al. [4]. Unsupervised Learning in the transfer learning case follows the same label-setting trend as inductive learning but instead contains no labeled data in both the source and the target domains [4, 28]. Unsupervised learning in a transfer learning setting is a very

uncommon technique and was only represented by 1% of papers that were reviewed in Azari et al. [4].

Transfer learning can be broken down even further in the sense of which approach is taken. There are currently three main approaches to transfer learning as described by Pan and Yang [28] and Azari et al. [4]. These approaches consist of instance, feature, and parameter-based approaches. Feature-based approaches constitute 71% of the papers reviewed by Azari et al. [4]. This approach refers to when the transfer of knowledge is based on the feature space presented and learned by the models. In particular, the feature spaces are often the same between source and target in this approach. A typical feature-based approach allows the network to learn the features of the source and then fine-tune only the last couple layers of the network, keeping the source feature representation frozen in the first couple layers [4, 28]. A feature-based approach will be explored in this thesis. The second most common approach reviewed by Azari et al. [4] is the parameter-based approach. The parameter-based approach attempts to transfer knowledge from similar domains based on the parameters used in the models. Thus the update of the models is based on updating the parameters, not the features [4, 28]. The third most common approach for transfer learning is the instance-based approach [4]. An instance-based approach relies on specific instances, or data points, to drive the transfer of knowledge. This approach typically uses specific instances from the source domain in the target task, such that strong, specific examples from the source can influence the target's outcome [4, 28].

In predictive maintenance ventures using transfer learning, a large portion of them utilize bearing fault data, 69% based on the review conducted by Azari et al. [4]. Both Wu et al. [42] and Chen et al. [8] use the motor bearing dataset provided by Case Western Reserve University. Their approaches differ slightly. Wu et al. [42] used a type of transfer learning to create auxiliary datasets of the data based on an LSTM model. They then used a JDA to reduce the difference in probability distribution of the datasets. Chen et al. [8] simply used a temporal CNN to classify the bearing fault data. In both cases, however, they maintained one key similarity. This was that they converted the raw vibration data that they had into images of the waveforms. This step in the typical transfer learning step for predictive maintenance can be seen in many other studies. For example, Han et al. [17] developed a

simple novel approach to transfer learning which simply used a standard DL model as a base and then used limited target data to show the capabilities of transfer learning. However, again, the data that was used was in the form of images. Wen et al. [39] used the pre-trained ResNet image classifier as their base model, while they retrained it based on raw time-domain data that they had converted into RGB images. Liu et al. [22] also performed a similar task based on the LeNet-5 image classification model. Image classification can be seen to be a large chunk of transfer learning’s applications in predictive maintenance. This is consistent with transfer learning’s typical applications outside of predictive maintenance as well. The typical uses for transfer learning in literature reside in the domains of image classification and natural language processing [28]. For this thesis, a case study using tabular data will be performed as the use of tabular data in the literature is severely lacking.

Chapter 3

Problem Definition and Approach

3.1 Problem Definition

This thesis explores the classification of misalignments between a motor and its blower, with a focus on utilizing transfer learning across motors of varying horsepower. While predictive maintenance initiatives like this can be implemented independently of transfer learning, certain limitations come into play, primarily related to the nature of the data. Predictive maintenance hinges on the availability of high-quality data, which can be a resource-intensive and costly endeavor to procure. Moreover, the collected data should pertain specifically to the target machine to which the company intends to apply predictive maintenance practices. Another challenge lies in the time it takes to train models on large data sets, a process that can be quite lengthy, especially in dynamic machine environments. Addressing these limitations can also result in reduced computational resource requirements.

3.1.1 Data Quality and Availability

The main reason for the introduction of Transfer learning as a predictive maintenance technique comes from the fact that there are often too many limiting factors to both data and its collection. With the advance of Industry 4.0, there has been more and more availability of data, however, these efforts are restricted by multiple different shortcomings that often accompany data collection. Mahlamäki and Nieminen [23] define these different areas as

TOP or Technology, Organization, and People. They argue that these three aspects of data collection play a large role in whether or not quality data will be able to be captured.

Some companies can either not afford or are not willing to invest in high-end sensors that capture high-quality data. This often leaves them with standard run of the mill sensors that do not often produce data that is easily usable. Not only this, but also having the proper training in data collection and maintenance for machines takes time and effort in order to do properly. It was shown by Grudin [16], that as maintainers do not have access to the data they are providing, they will not be seeing the benefits of their work. Thus, they do not have as big of an incentive to provide the best data possible. As well as keeping the sensor maintainers motivated to produce high-quality data, Sandtorv et al. [31] also discussed the difficulties that come with finding people with adequate competencies to collect high-quality data. Oftentimes, the training required for these individuals needs to span both the data collection and data usage domains because if they do not understand the use of the data, they may often take shortcuts or steps that do not lead to more usable data.

Along with these human factor considerations in the data collection process, the type of data recorded can also often come into effect. Some companies use expensive software tools to clean and develop their raw data into easily usable features for predictive maintenance tasks. Others hire specialized feature/data engineers who are able to transform the data to provide the best view of the trends in the raw data. However, these employees and software are expensive, and not enough funds are allocated for them in the overall budget. For this reason, it can often be seen that even if you get over the hump of collecting usable data, the data will likely be noisy and in its raw format. This makes the development of new machine-learning models from scratch difficult.

The use of transfer learning can mitigate these concerns. Its ability to use small amounts of data to transfer the skills of a past deep learning model allows for less time, money, and effort to be spent on data collection. Therefore, these problems mentioned above are not as influential on a company’s actual ability to use predictive maintenance opportunities.

3.1.2 Time and Compute Costs in Conventional Training

In terms of predictive maintenance, not only the data collection is time consuming and costly, the actual training of the deep learning techniques is often a lengthy process. Deep learning techniques, in general, tend to take longer training times than traditional machine learning techniques. Especially as the amount of data, the complexity of the model, and the complexity of the data increase, the training times will significantly increase. These training times are often too slow to keep up with ever-changing production systems. This issue can be addressed in terms of using higher performance computers. Such as using graphics processing units (GPU) which have been shown to run consistently at least 2-3 times faster than the typical central processing unit (CPU) [25]. However, GPUs are expensive and would increase the budget of predictive maintenance jobs.

Instead, transfer learning approaches have been shown to speed up training times and cut down on computational resources. Techniques such as incremental transfer learning, in which the model is constantly being updated, show a reduced number of epochs to train while also modeling new data [36].

3.2 Approach

To address the challenges associated with predictive maintenance ventures in terms of data and resources, a transfer learning technique for belt-driven motor misalignment classification has been proposed. The approach is broken down into two parts. The first uses vibrational and Motor Current Signature Analysis (MCSA) feature data using Zero-shot transfer to transfer knowledge between 3 horsepower (HP) and 5 HP motors. The second uses MCSA and vibrational data to transfer between 3 HP and 5 HP motors using an incremental technique to evaluate one-shot and few-shot transfer learning. Both approaches will evaluate different deep learning architectures and determine the best fit for the given task. They will be compared against a baseline model in which the model is trained on the same split of data that is used in the transfer learning step.

3.2.1 Transfer Learning

Transfer learning in terms of deep learning model development attempts to emulate the behavior of the human mind in transferring related knowledge/skills to new domains. In general, transfer learning attempts to take a pre-trained, often generalized model and use it as the base point for a new task. In a more specific sense, transfer learning takes a pre-trained model in some domain A and then takes data from a similar or related domain B and re-trains parts of the network such that the network is now specific to the task associated with domain B while using its underlying knowledge from domain A as its base. As mentioned earlier, transfer learning, in this sense, generally does not need large amounts to train the new parts of the model unless the domains A and B are sufficiently different. This allows for the collection of less data depending on how related the two tasks are from the separate domains.

A technical definition of transfer learning can be presented as follows. Assume two domains: a source domain defined as

$$D_S = \{X_S, Y_S\} \quad (3.1)$$

and a target domain defined as

$$D_T = \{X_T, Y_T\} \quad (3.2)$$

where X_S and X_T represent the respective source and target domain's feature space. Y_S and Y_T represent the label space of the two domains. The respective tasks of the source and target domains can be defined by T_S and T_T where the source domain's task can be shown as

$$T_S = \{Y_S, f_S(X_S)\} \quad (3.3)$$

and the target domains task is represented as

$$T_T = \{Y_T, f_T(X_T)\} \quad (3.4)$$

In this case, $f_S()$ and $f_T()$ represent the prediction function of the two tasks. The two domains, D_S and D_T , should be sufficiently similar but should still satisfy the condition $D_S \neq D_T$. Meanwhile, the two tasks in this case of transfer learning, T_S and T_T , should satisfy the condition $T_S = T_T$. The goal of transfer learning here is to improve $f_T()$ by exploiting the relationship between D_S and D_T .

Transfer learning's popularity has increased in the past couple of years due to its significant progress in the field of computer vision. Typical uses of transfer learning in literature deal with image classification. A significant example is when a model has been trained on highly variable images with many different classes. Thus the early layers of the model have learned to be able to extract general features from each picture. With this in mind, a new classification layer is then trained on a specific subset of images. The model will be able to pick out the features based on the general model and then apply the specifics it has learned in its new layer to the output. This is a quite popular example of using transfer learning. However, transfer learning using tabular data sources is a topic that has not been widely researched. Instead, most predictive maintenance cases of transfer learning deal with image data.

For our case of transfer learning for predictive maintenance, a couple of different configurations will be explored. The first constitutes a form of transfer learning that does not typically fall under the technical definition of transfer learning. This technique is the zero-shot technique. This technique trains a model on a certain domain, in our case, a certain level of HP's fault data, and then uses a similar domain, in our case another level of HP's fault data, to test the model. Thus, there is no retraining of the model based on the new data. Instead it is inferred that the domains share significant similarities between them that the trends learned by the original model are the same as the trends that would be retrained into the model in classical transfer learning. This technique actively eliminates the need for extra training data and extra computational resources. For this reason, the technique is very attractive in its initial research.

The next technique employed will be to remove the output layer of the initial model and retrain it on the new domain's small data set. This is the most classical form of transfer learning and is often seen with generalized initial models like in the case of image

classification. By only training the output layer, it allows for the the output to pick up on key distinguishing features while basing the initial observations on the old domain's data. By only training the output layer, not much data is required and computational cost is not raised. Another option would be to add new layers on top of the preexisting model other than just an output layer. This could be done if the data is increasingly complex and thus would need more than an output layer to learn the key difference between the two domains. While this technique would require slightly more resources, it is a good option for when there is added complexity in the transfer domain's data set.

3.2.2 Deep Learning Models

Deep learning offers many different approaches and models; however, certain architectures are better suited for certain types of problems. In terms of the data gathered in this predictive maintenance venture, its form is a tabular feature or waveform data. In the case of the waveform data, there is a strong temporal relationship between data points, as each second of data contains 10,240 samples. Since the structure of the data is tabular, models such as CNNs alone would not necessarily be the first choice since they typically deal with spatial and image data. Instead, MLPs would be better suited for this type of data since they have stronger capabilities with tabular data. However, the strong feature extraction capabilities of a CNN and the temporal ability of an LSTM make a CNN-LSTM hybrid model an easy second choice. Due to the specific structure of data and the task at hand, the main models chosen for the transfer learning analysis were an MLP and a CNN-LSTM.

Multi-Layer Perceptron

Due to its ability to fit almost any deep learning problem, the MLP is often considered the baseline deep learning model. Its flexibility allows for many different data structures and types to be utilized. In terms of predictive maintenance, data often comes in multiple different forms showing why an MLP is an important model for predictive maintenance. In this case, the tabular data can be used without any hitch, making a seamless and simple sensor to model pipeline. MLP's ability to learn features is another reason this architecture

was chosen to carry out this specific task. Its ability to learn intricate relationships between the different features in tabular data makes it a robust network architecture in terms of predictive maintenance. Along with an MLP’s flexibility and feature learning, an MLP’s ability to model nonlinear dependencies is top-notch. Often times in predictive maintenance scenarios, the relationship between input features and output variables is non-linear. Therefore, an MLP is a strong choice since it carries strong capabilities for modeling non-linear relationships between input and output variables.

CNN-LSTM

By using a combination of the CNN and LSTM frameworks, the model contains the best of both worlds. From the perspective of CNN, its feature extraction capabilities can be taken advantage of to define strong base layers of the model for the transfer learning venture. Along with its feature extraction, CNNs are extremely versatile when it comes to modeling the spatial patterns of the data. This allows for strong relationships between the features to become present in the model such that they can be stressed when the transfer of knowledge takes place. The LSTM layers will have the extracted features fed into them from the CNN feature extraction layers. The LSTM layers will work well to extract temporal relationships between the data. Its temporal capabilities will be able to capture the order and timing of events, which is crucial for predictive maintenance tasks. Thus the combination of the spatial and temporal relationships of the data will be strongly represented in the model such that the model will be robust and easily transferable. Along with this, the LSTM’s ability to reduce overfitting with respect to sequence data will be able to play a strong role in the classification of temporal data as well. Overall, the CNN-LSTM structure combines the best capabilities of both models that would be necessary for any predictive maintenance venture. Their strong relationship modeling skills will also strengthen the ability of the model to transfer knowledge between the two domains used.

Chapter 4

Dataset, Implementation, Results

The dataset was produced through a joint collaborative effort between the University of Tennessee, Knoxville, and a Sister Company. Specifically meant for predictive maintenance research, multiple different belt-driven motors varying in horsepower had two basic faults introduced into them. These were misalignments of the motor and the drive pulley regarding a linear offset and an angular offset.

The testbed is shown in Figure 4.1. It displays the motor fixed in place on a T-slotted table, which includes a cross-slide rotary fixture such that the misalignments could be measurably introduced. The 3 Horsepower motor, in this case, is connected to a blower through the use of a drive belt. The entire setup was contained in a container for safety precautions.

Figure 4.2 shows where the sensors were located on the motor. The types of sensors used were vibration-recording accelerometers, temperature-recording thermocouples, and current/voltage recording Motor Current Signature Analysis (MCSA) sensors. The vibrational sensors are placed on three sides: the side facing the blower, the side away from the blower, and the top of the motor. The vibration sensors are paired in groups, with one being on the front (pulley-side) and one on the back. There are four total thermocouples used. Two are shown in the figure on the motor housing while one of the others is placed on the front-facing bearing of the blower, and the other is floating, capturing the ambient temperature of the setup. The MCSA sensors are not shown in the figure either, they are located near the power source of the motor.

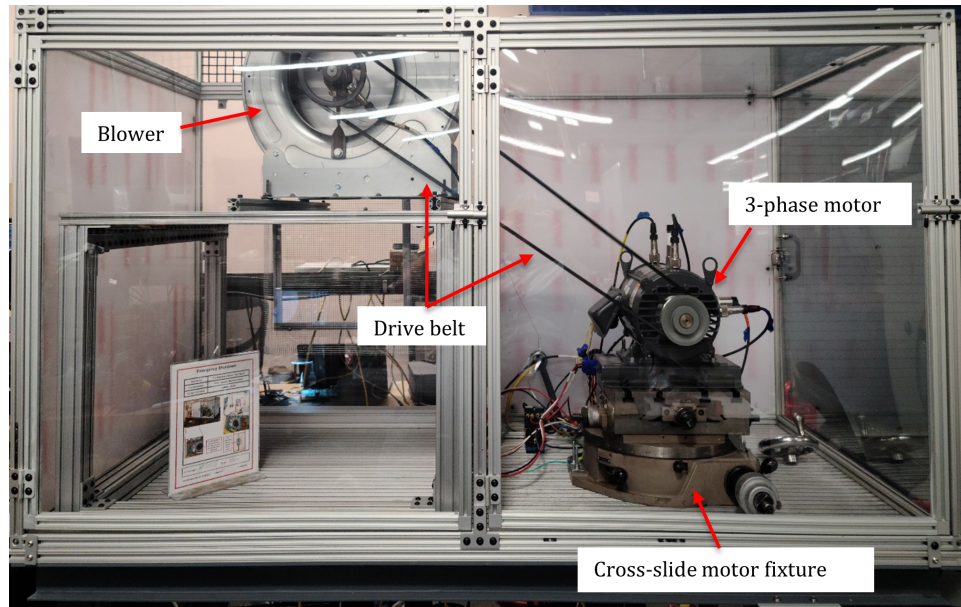


Figure 4.1: Belt-driven motor test bed setup.

Table 4.1: Levels of Misalignment introduced.

Misalignments	
Linear (Inches)	Angular (Degrees)
0	0
0.1	1
0.2	2
0.3	3
0.5	5

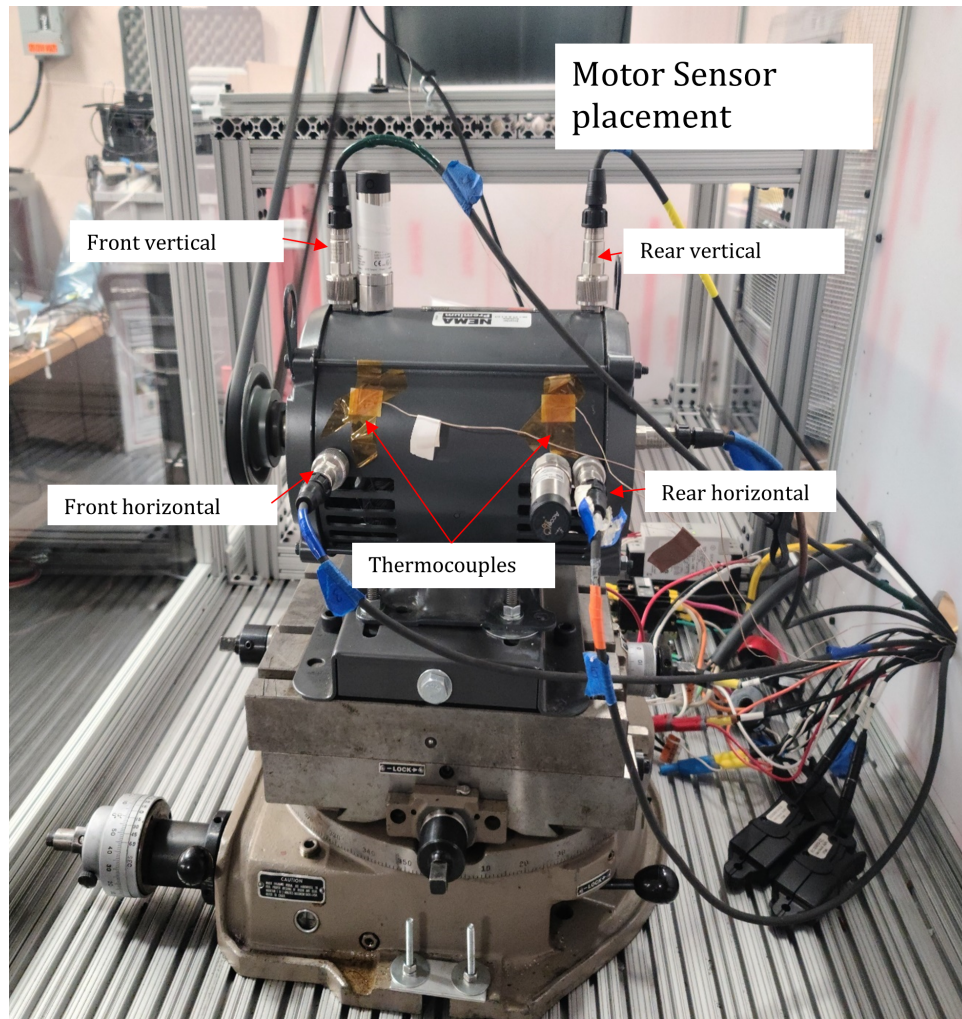


Figure 4.2: Sensor placement on motor housing.

The different levels of misalignments introduced into the differing horsepower motors are shown in Table 4.1. Both offset and angular misalignments share a similar trend in misalignment. First, a baseline is established with perfect running conditions. From there the misalignments are introduced such that data for each misalignment can be gathered over the course of a couple of days for robust data collection. The maximums found for the misalignments were 0.5 inches offset and 5 degrees angular. Under these maximum conditions, the motor was shown to cause dangerous behavior, in some cases producing smoke.

Figure 4.3 displays the type of misalignment that offset refers to. An offset misalignment, in this case, refers to a misalignment in the X direction toward the pulley side of the motor. This change is denoted by the Δx . This offset was created by using the cross-slide table attachment. Figure 4.4 displays angular misalignment. This type of misalignment is created by rotating the motor about the y-axis, thus creating a misalignment between the motor, belt, and blower. This is denoted by a $\Delta\theta$.

The sensors used were National Instruments (NI) 9232 for vibrational data, NI 9213 TC for temperature readings, and NI 9239 MCSA for the MCSA data. This data is sent to an Insight CM (ICM) server, which stores and processes it. From there, the data can be pulled from the server using PowerShell scripts in the form of TDMS files. From the TDMS files, CSVs are produced for ease of integration with ML and DL techniques.

For the different horsepower tests, there are three different categories of data: vibrational, temperature, and MCSA. The vibrational data is derived from six accelerometers placed strategically on the motors. These sensors provide raw waveform data in the form of 10,240 data points per second. From the raw waveform data, ICM produces seven predefined features per sensor. This, in total, produces 42 total features extracted per second from the raw vibrational data. These features are compiled into a CSV file for each day of data. The temperature data comes in the metric of Fahrenheit and has a total of four different readings per second. There are four thermocouples placed strategically around the motor, such that the heat produced from the motor can be analyzed at any given time. These four temperature readings are compiled into the same CSV file that contains the vibrational features.

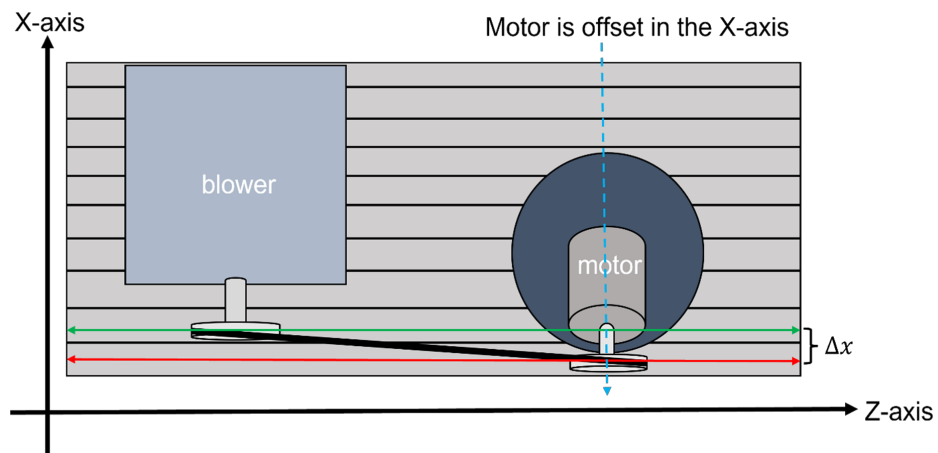


Figure 4.3: Illustration of Offset Misalignment.

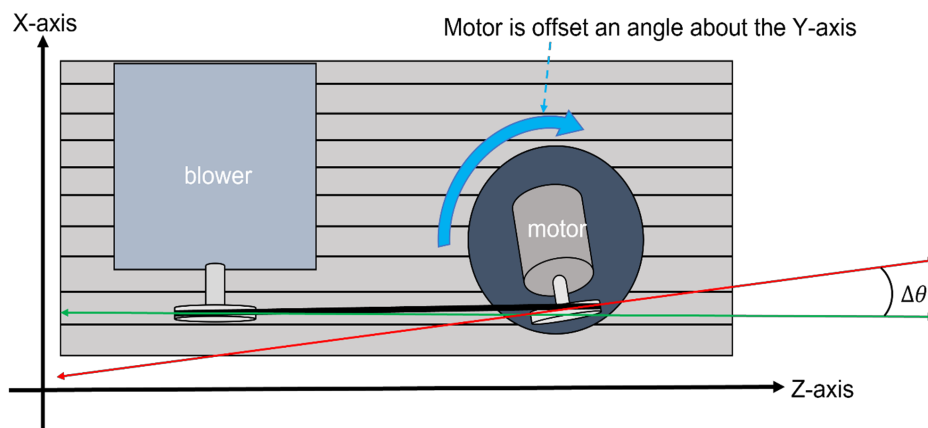


Figure 4.4: Illustration of Angular Misalignment.

MCSA data is data produced based on the unique electrical current signature produced by the motor at different levels of running. The electrical signature can be affected by load, anomalous conditions, etc., such that recording the signature allows for non-intrusive data collection that has the ability to show fault. The MCSA data comes in a similar waveform to that of the vibrational data, however, just like in the case of vibration, ICM produces its own set of predefined features. From the MCSA sensors, there are a total of 18 features that have been deemed usable for the task at hand. These features are compiled into a separate CSV file from the other features so that the comparison between how the different models perform with different data can be analyzed. The vibrational features and MCSA features are displayed below in Table 4.2.

4.1 Data Analytics

4.1.1 Pre-Processing

For the processing required prior to training the prediction models, four steps were followed and evaluated based on their effectiveness to provide stronger results for the predictive maintenance predictions. These steps were to look into different normalization techniques, the removal of outlier data, the transformation of the data to fit the specific deep learning models that were being used, and the sampling techniques that were to be applied.

Normalization

For normalizing data in the field of data science, there are two strong contenders that are typically deemed to be the strongest methods of normalization. These two methods are the standard scaler and the min-max scaler. Both of these have open source implementations in Python that allow for ease of access. These techniques can be found in the Sklearn package.

Standard scaling refers to a technique in which the data is scaled to fit a certain specified range, with the default range being between -1 and 1. This is a very common scaling technique that allows a stronger relationship between features that have largely differing ranges. To perform standard scaling, the mean, μ , and the standard deviation, σ , of

Table 4.2: Features derived from the data.

Vibrational Feature	Unit	MCSA Feature	Unit
Crest Factor	g g	MCSARMS	volt
Peak Peak	g	Phasor Magnitude	volt
High Frequency	grms	Phasor Phase	degree
Derived Peak	g	Rotor Bar Sideband	dB
True Peak	g	Effective Service Factor	
Kurtosis	g g	Torque	lb ft
RMS	g	Unbalance	
		Derating Factor	
		Efficiency	
		Load	HP
		MCSA Speed	RPM
		Percent Load	
		Active Power	kW
		Apparent Power	kVA
		Percent Full Load Amps	FLA
		Power Factor	
		Reactive Power	kVAR
		Torque Ripple	

each feature must be calculated. Then, the Normalized value of the data point would be represented by Eq. 4.1.

$$\text{Standard Normalized Value} = \frac{\text{Original Value} - \mu}{\sigma} \quad (4.1)$$

MinMax scaling is a technique that scales the features to be within a fixed range, typically 0-1. MinMax scaling provides strong interpretability, and it works better for neural networks such as MLP and CNN-LSTM. To perform MinMax scaling, the range of the data and the minimum value is needed. The normalized value based on MinMax scaling is represented by Eq. 4.2.

$$\text{MinMax Normalized Value} = \frac{\text{Original Value} - \text{Min}(A)}{\text{Max}(A) - \text{Min}(A)} \quad (4.2)$$

After some testing, it was found that MinMax scaling would be the most influential normalization method for this endeavor. Combined with the test results, the knowledge that MinMax scaling is often applied when neural networks are used and for non-normally distributed data, it was deemed the most effective form of scaling for this case study.

Outlier Removal

Most outliers present in the data are from startup and shutdown conditions during the testing phase of data acquisition. Since these conditions are not part of the motors standard operating behavior, these sections of data needed to be removed. These conditions can be easily seen in Figure 4.5.

A simple quartile range outlier removal technique was chosen to perform this step for its strength and flexibility. To start the process the first and third quartiles, Q_1 and Q_3 , were calculated for the specific feature. Based on the difference between these quartiles, the interquartile range, IQR , was found.

$$IQR = Q_3 - Q_1 \quad (4.3)$$

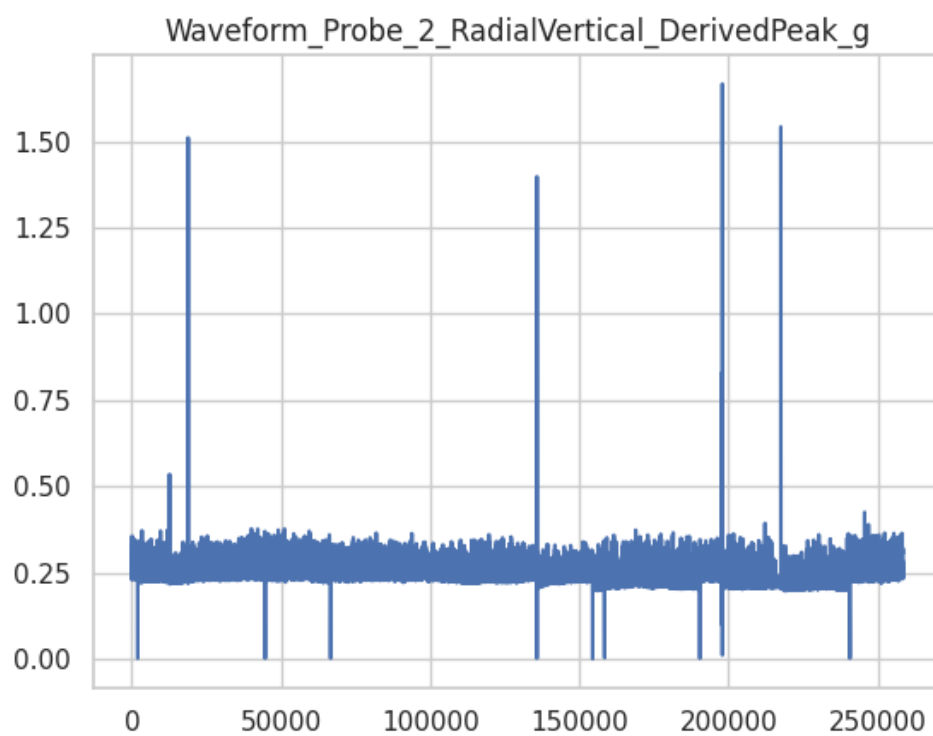


Figure 4.5: Feature showing startup and shutdown outliers present in the data.

A whisker range was then defined based on the feature present, which in turn was used to develop the upper and lower ranges in which all standard data was to be found. U is a user defined factor.

$$\text{Whisker Range} = U \times IQR \quad (4.4)$$

$$\text{Upper Range} = Q_3 + \text{Whisker Range} \quad (4.5)$$

$$\text{Lower Range} = Q_1 - \text{Whisker Range} \quad (4.6)$$

Then the data points outside the range were removed, leaving only the running condition data. This was repeated for each feature of the differing data sets until all outliers were removed as represented by Figure 4.6.

Less than 0.5% of the data was removed based on this technique, but after a visual analysis of the data, it could be seen that all start-up and shutdown conditions had been removed. For the most part, the 5 HP data sets contained mostly continuous data in the sense that there were easily identifiable outliers for each feature; however, that was not always the case for the 3 HP data sets. The 3 HP data sets tended to contain certain features that had chops in the data and did not always follow similar trends. For this reason, there were not as many outliers removed from the 3 HP data sets as for the 5 HP data sets. This particular problem with the data is discussed more in length in section 4.1.2

Data Transformation

Depending on the different network structures used, the data needed to be input in certain formats. For the waveform data, sampled at 10,240 samples per second, the structure of the neural network needed to be a bit more complex than for the feature data as there are more subtle underlying trends than the more direct feature data. A CNN-LSTM hybrid was used as the more complex model. With the CNN being a spatial model and the LSTM being a temporal model, three dimensions were required to properly feed the data into the model. These three dimensions were the size of the data, the number of features, and the size of each time step. The number of features stayed static, but the size of the data was dynamic in relation to the time steps. To evaluate which size time step to use, a simple iterative

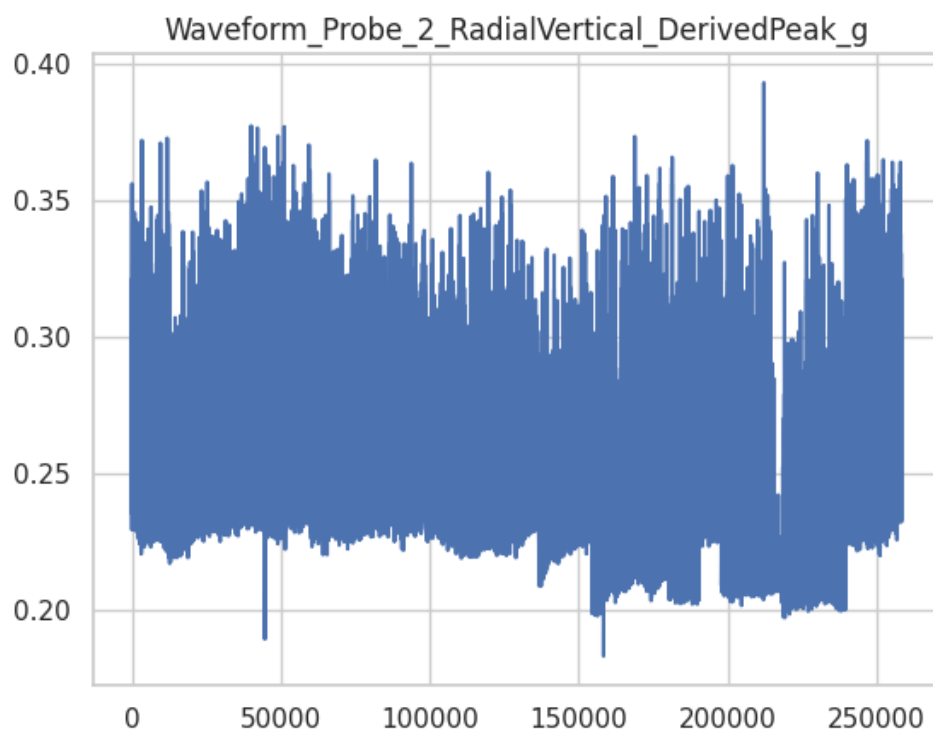


Figure 4.6: Feature after outlier removal technique was applied.

approach was applied to training and testing different models. Values ranging from 32 up to 10,240 were tested and evaluated based on the testing accuracy of the base models. After some experimentation, it was determined that the best sizes for the waveform time steps were 320 and 1,024 samples per time step.

For the feature data, sampled once per second, the same experimentation was performed to find the best size of the time steps per data entry. It was determined that only having one data point per timestep would provide the best results for the base models.

Sampling

Based on the sample sizes presented in Table 4.3 and the analysis in section 4.1.2, the last step of preprocessing the data is to decide on whether to sample and what sampling technique to choose. Based on the breakdown of class samples and the identification of DL as the approach, an over-sampling technique was deemed the most reliable. Based on the different oversampling techniques that are present, one that interpolates data was deemed the most suited characteristic for this task. Based on these observations, the main over-sampling technique that was applied to the data was the Synthetic Minority Over-sampling Technique (SMOTE). SMOTE was chosen due to its ability to create synthetic instances of data based on the interpolation of the minority classes. This enriches the data without duplicating existing samples. As well as interpolating the data, SMOTE’s abilities also apply to this problem through its generalization ability. By using SMOTE, some variability is added to the data such that the generalization of the model can be improved. SMOTE was only applied to the feature data, as it was not compatible with the structure of the waveform data.

SMOTE works by first selecting a random instance from the minority class, i . From there, it calculates the distance based on the Euclidean distance from the equation shown by Eq. 4.7.

$$d(A, B) = \sqrt{\sum_{i=1}^n (A_i - B_i)^2} \quad (4.7)$$

Then the k nearest neighbors are selected based on the calculated distances. From there, a new synthetic instance, i' , is made based on each of the neighbors selected. This is shown

by the equation shown by Eq. 4.8.

$$i' = i + rand(0, 1) * (k - i) \quad (4.8)$$

This process is then completed for each minority class until all classes contain the same number of samples as the majority class. SMOTE was applied to the entirety of both datasets, such that the oversampling of each class can be represented in the training and testing sets of the data. It was applied to the entirety of the datasets as a way to preserve the relationships between the data such that the characteristics of the minority class are appropriately represented. It was also done to promote the generalization of the imbalanced data. With imbalanced classes in training or testing, models tend to have a harder time generalizing to new information.

4.1.2 Exploratory Data Analysis

The Exploratory Data Analysis (EDA) will provide a deeper analysis of the data such that the limitations of the data can be presented. To best present these limitations, it is best to first understand what features are overtly important to our data science ventures. For this reason, a random forest model was run on all sets of data to best portray the feature's importance in predicting offset misalignment. After conducting this analysis, it was seen that for the vibrational data for the 5HP motor, the Thermocouples were the most important in predicting Offset. All four thermocouple features came within the first seven features in terms of importance. Along with the thermocouples, the next most important features based on the random forest analysis were Probe 6's High Frequency and Probe 3's Derived peak feature. Since the main focus of the EDA is to explore the vibrational data, the features just mentioned will be analyzed to show the similarities and differences between the 3 HP and the 5 HP data set.

Summary Statistics

Summary statistics are crucial to gaining a comprehensive understanding of the data presented. By understanding the data’s distribution, dispersion, and tendencies, the transfer learning efforts’ results will be more cohesive and paint a clearer, more vivid picture.

Table 4.3 refers to the sample sizes of each class of the Offset Misalignment data. These sizes are the same for both the Vibrational features and the MCSA features, as they were recorded and sampled at the same frequency. Each sample refers to one second of data and contains the number of features present depending on the type of data that is being observed. The 3 HP samples show that the testing started off consistent and then lost its consistency after moving from 0.1 to 0.2-inch offset. The large differences in the sample size of each class promote the idea of sampling and what technique would be most beneficial to DL models. The extreme lack of data in class 0.5 produces an interesting hurdle. With its sample size being more than four times smaller than the next smallest sample size and over twenty times smaller than the largest sample size, the issue of how to approach this imbalance in data is presented. This is an issue tackled in section 4.1.1.

The range of sample sizes for the 5 HP data shows more consistency than the 3 HP data. The two main outliers come from the Baseline and 0.5 offset data. These outliers contain enough data such that sampling techniques are not completely necessary but would still be suggested. The overall totals of both data sets are relatively substantial for a DL project, but it could be said that the large difference in sample size could have some effect on how strong of a base model can be produced for transfer learning, especially when considering the larger, more complex models that are to be applied.

Two of the key features shown by the Random Forest analysis were Probe 6’s High Frequency and Probe 3’s Derived Peak. The key statistics for Probe 6’s High Frequency are displayed in table 4.4. Overall, the differences between the different HPs in this case are relatively small, with only slight variations in all four key statistics that were measured. Calling back to the heatmap’s correlation exploration, the small difference in the statistics represents consistency with the findings there. With Probe 6’s features containing by far the

Table 4.3: Sample sizes of each class of Offset Classification.

Sample sizes of Offset Misalignments		
Offsets (in.)	3 HP	5 HP
0	38,987	18,680
0.1	35,517	66,601
0.2	84,170	87,831
0.3	18,272	59,492
0.5	4,089	25,836
Total	181,035	258,440

Table 4.4: Key statistics of Probe 6 : HighFrequency grms.

Probe 6 : HighFrequency grms		
Statistic	3 HP	5 HP
Mean	0.0638	0.0769
Standard Deviation	0.0107	0.0076
Minimum Value	0.0454	0.0556
Maximum Value	0.1035	0.1000

most correlation between the two datasets, it seems necessary that the statistics between the 3 HP and 5 HP Probe 6's High-Frequency feature be relatively consistent.

Contrary to the similarities between the 3 HP and 5 HP Probe 6's High-Frequency feature, Probe 3's Derived Peak feature shows very different statistics based on which HP motor is running. Based on the correlation seen earlier, the correlations between the two data set's Probe 3 features are small if any at all. This stays consistent with the findings of the summary statistics presented in table 4.5. So, although this feature is likely to be important in the initial base models, it seems unlikely that the feature will provide importance to the transfer learning endeavors. Without scaling, which was shown as detrimental to the transfer learning efforts, this feature will likely provide little insight on the transfer of knowledge.

Visualizing the Data

Visualizing the data is an important part of the data analysis step. By placing the data into a tangible form that can be easily digested by the human brain, certain aspects of the data can be more readily grasped. By not viewing the data, the data scientist may miss out on crucial trends that can inform them of the appropriate preprocessing steps and models to use. It also has the ability to inform them of the strength of the data and can easily present problems with the data, such as human error. In the following analysis the two featured features are viewed based on their box plots and standard waveform plots. Both provide insightful knowledge of how the data affects the models it will be used in and the problems they present to the task at hand, transfer learning.

When viewing the Box plots associated with both data sets' Probe 3 Derived Peak feature in Figure 4.7, it's easy to see the difference in range that was discussed in the summary statistics section. However, the Box plots also show the differences between the two HP motor datasets. The trends that the differing levels of offset follow are starkly different between the two datasets. The 5 HP data shows a strong continuation of means and distribution from the Baseline all the way to 0.5 inch offset. On the contrary, the 3 HP data contains large differences between the different levels of offset. When looking at 0.3-inch offset, every single data point is above the mean of both 0.1 and 0.2-inch offsets. This large difference in the distribution of the status labels provides easy patterns for DL models to pick up on.

Table 4.5: Key statistics of Probe 3 : DerivedPeak g.

Probe 3 : DerivedPeak g		
Statistic	3 HP	5 HP
Mean	0.1093	0.3206
Standard Deviation	0.0102	0.0414
Minimum Value	0.0922	0.1930
Maximum Value	0.1466	0.5656

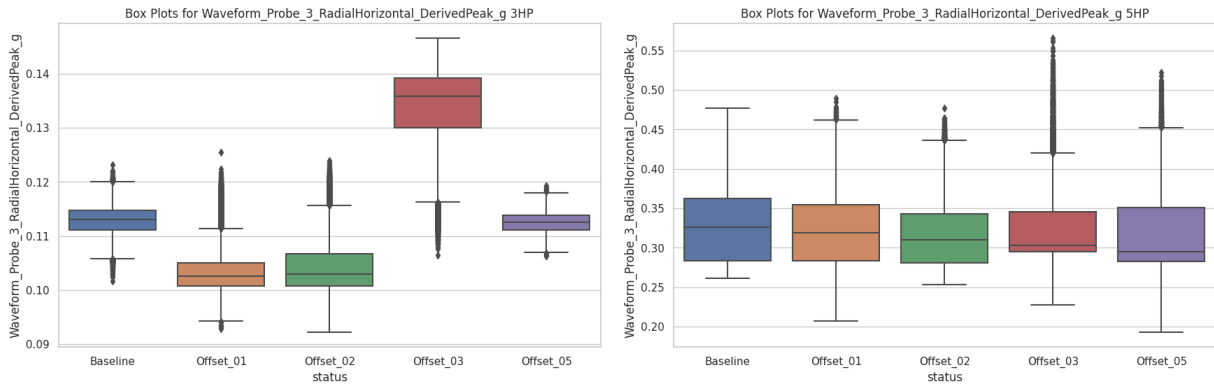


Figure 4.7: 3 HP (Left) and 5 HP (Right) Probe 3 Derived Peak Box plots.

On the transfer learning end, however, training a base model based on the 3 HP data would establish trends that are not seen in the 5 HP data, which in turn would make it difficult to transfer knowledge between the two models.

Figure 4.8 shows Probe 3's Derived Peak as its raw waveform after removing the outliers. As can be seen, the two follow extremely different trends. The 5 HP feature shows relatively consistent behavior during tests, but the 3 HP feature shows little consistency. Although the testing for the 3 HP is not linear in terms of which offset comes next, it would still be expected to follow a consistent trend. This difference between the two shows the effects that human error and time in between tests can truly have on a set of data. Also, since the data for the 3 HP feature is so spread out, there truly are little to no outliers present. In essence this can be a positive of a data set, however in this case, it seems to be more of a detriment to the overall validity of the results of the models.

Figure 4.9 shows the box plots for Probe 6's feature, High Frequency. The overall trends of both the 3 HP and 5 HP features share roughly the same pattern with a small initial drop in means to a large increase in offsets 0.3 and 0.5. The ranges of both HP's features stays relatively the same as well. Both of these insights show this feature will have a stronger correlation when used in a transfer learning setting than the previously seen Probe 3 feature. There are some key differences based on the box plots, but these are not substantial enough to be deemed problematic to transfer learning endeavors. The key insights into this feature come from the raw visualization of the waveform data.

Figure 4.10 shows the difference in the raw waveforms captured for Probe 6's High-Frequency feature. It can be easily seen that the two data sets contain very different trends for this specific feature. The 3 HP feature shows a parabolic trend after the initial baseline, while the 5 HP feature shows a more linear relationship with sections of data standing apart from the trend. The 5 HP trend seems to imply either an identifiable fault within the motor at the time of data capture or possibly human error in the adjustment of said probe. When viewing all other probe's High Frequency trends for the 5 HP motor, they all follow linear trends with little to no variation like what is seen in Probe 6's feature. This could be due to the specific placement of the probe on the motor or human error as is suggested by other trends seen in the data that will be discussed later. Although this feature contains two

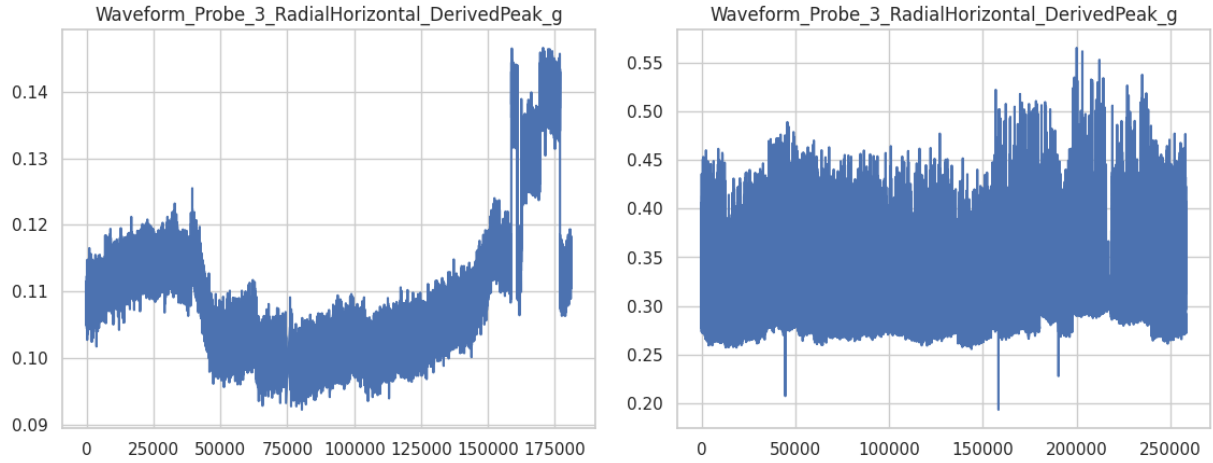


Figure 4.8: 3 HP (Left) and 5 HP (Right) Probe 3 Derived Peak.

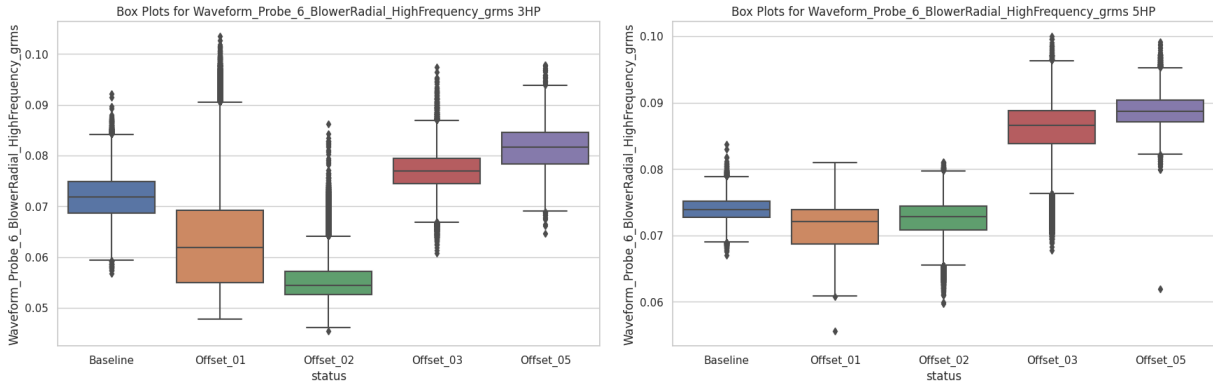


Figure 4.9: 3 HP (Left) and 5 HP (Right) Probe 6 High-Frequency Box plots.

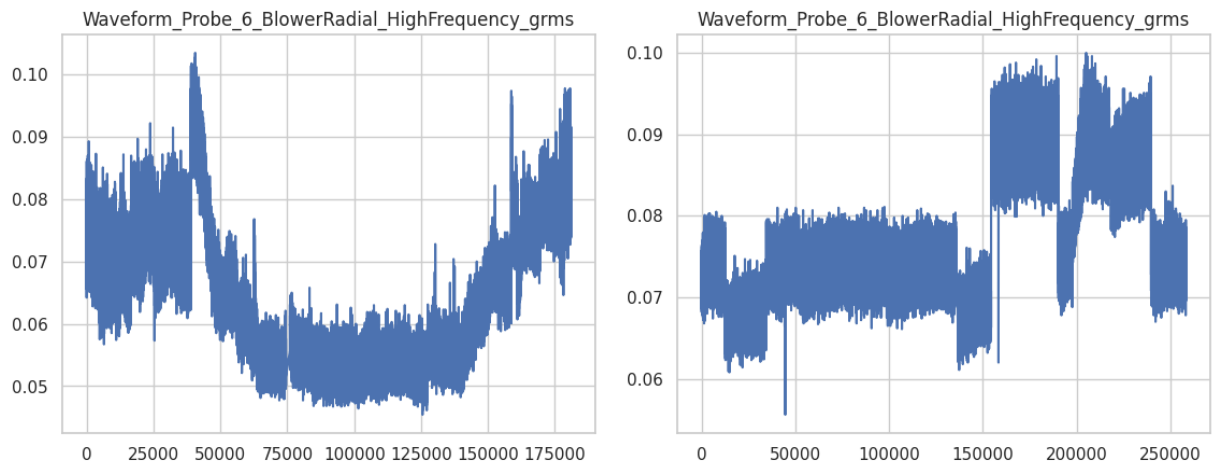


Figure 4.10: 3 HP (Left) and 5 HP (Right) Probe 6 High Frequency.

starkly different trends, its usefulness can be seen in the transfer learning domain based on the similarity in statistics present in Figure 4.9. These trends may be influenced by outside error and could be important for this reason, but it is undeniable that this feature will be important for transfer learning.

It is also important to visually evaluate the relationships between the two data sets. One potential way to view this relationship is through the correlation of the features between the two data sets. This correlation can be seen in the heatmap represented in Figure 4.11. From the correlation between the two data sets, it can be seen that there is a notable correlation between the features towards the end of the feature list, which refer to the features of Probe 6, the blower radial sensor. The other features show mainly a slight negative correlation. By understanding this correlation, it can be seen that Probe 6's features will be the most influential to the transfer learning efforts. In terms of the MCSA data sets, Figure 4.12 shows those correlations. Based on the correlations between the two datasets, it can be seen that most of the correlation is very slight, not passing the 0.25 correlation coefficient. This is not a good indicator for the transfer learning efforts. With little correlation, the data sets will likely not adjust well to each other.

Key Findings

The key findings from the EDA are presented here. These key findings provide insight that will be helpful in analyzing the results of the transfer learning process. The first main key finding is the differences that are present between the same features in the different HP data sets. It would be theorized that the same set of features, with the same testing conditions and the only difference being the HP of the motor, would produce similar trends across features. However, in this case, it can be seen that many of these features don't follow similar trends or even similar statistics. And in the rare case they follow similar statistical patterns, the trends shown seem to be influenced by some outside source. There is little correlation between the two data sets except for a select few of Probe 6's features, which have been theorized to contain human error. Other than Probe 6, there seems to be little to no correlation with the rest of the features in both vibration data and MCSA data.

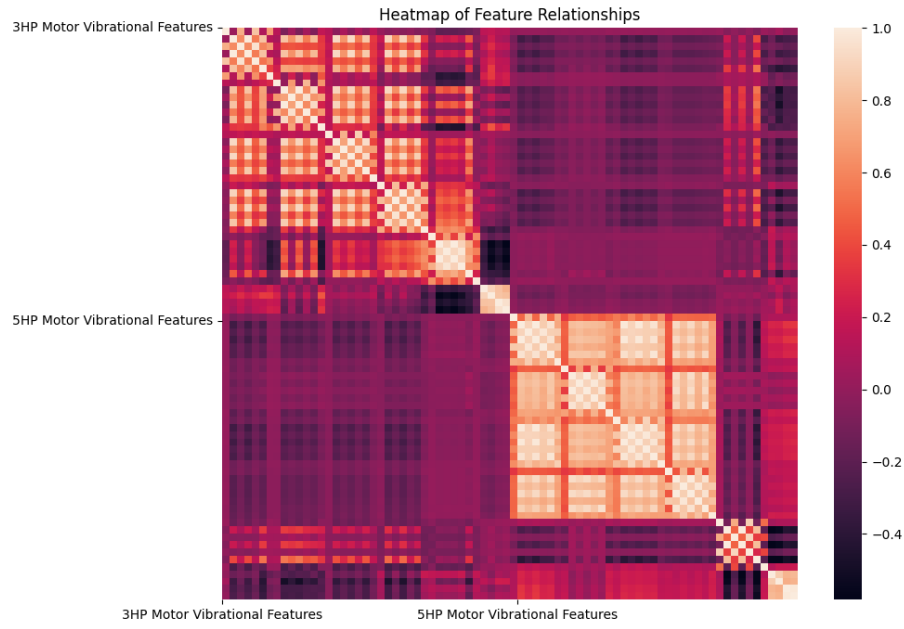


Figure 4.11: Heatmap of 3HP and 5HP Vibrational Data correlations.

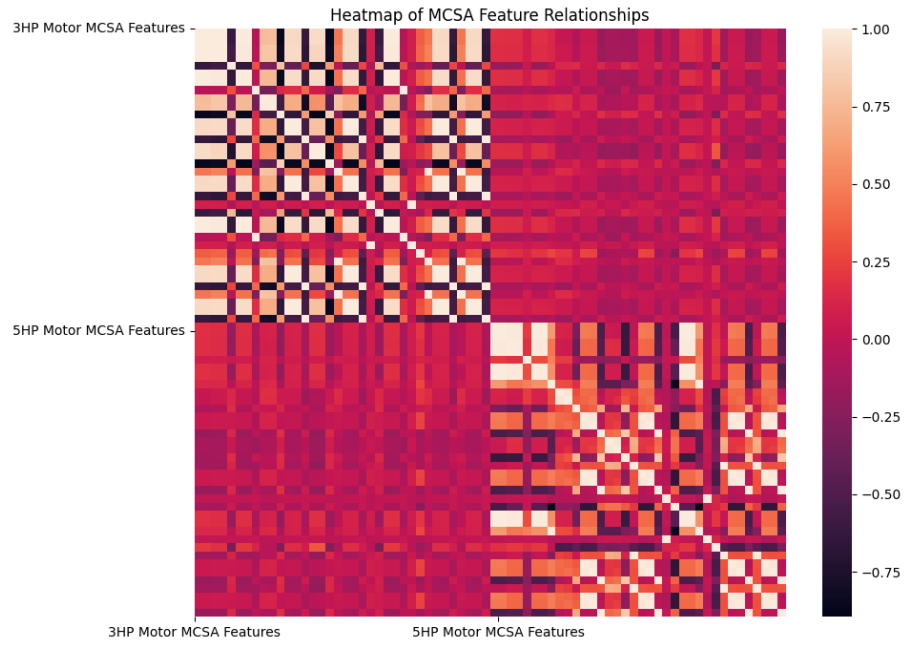


Figure 4.12: Heatmap of 3HP and 5HP MCSA Data correlations.

Another key finding in the EDA was the small presence of outliers in the 3 HP data set. Compared with the large amounts of startup and stopping conditions recorded in the 5 HP data, the 3 HP data was considerably lacking. The 5 HP outliers were easily identifiable by the human eye, and their cause was easily identifiable. However, there were little to no outliers present in the 3 HP data due to the structure of the trends. A large number of the features for the 3 HP data contain no linear trends in data, which in some cases contain large jumps in the data that would not be present in standard data without the inclusion of human error.

Another major find in the data that has not been discussed until this point is the human errors that were presented in the 3 HP motor's Probe 1 that caused its features to be removed completely from this endeavor. During the testing on the 3 HP motor, it was noticed that Probe 1's connection to the motor had become unreliable. For this reason, the Probe was removed and placed back with a new connection. This simple maintenance of the probe produced waveforms that had nearly double the typical range of the past recordings. This can be seen in figure 4.13. The probe was then reset again to where it recorded trends similar to that of the prior testing, but the tests that were run when the probe was recording incorrectly were never rerun. For this reason, it was deemed necessary to remove all data correlated to Probe 1.

Other considerations to take into account with the quality of the data were inconsistencies with how the data was collected. Often times, during the data capture process of the 3 HP and 5 HP data, there was data missed due to outages of the ICM server and wrongly configured capture alerts. These developments were typically caught before too much data was lost. For the 5 HP data, the tests were simply rerun in the order they were missed, thus the offsets continued to be captured in standard order. However, for the 3 HP data, when the data capture errors occurred, testing was continued, and the lost data was rerun after the completion of the standard testing timeline. This led to reusing lesser offsets after the greatest offset had already been applied, bringing in the notion that some degradation had already been suffered to the motor such that this new lesser offset likely does not represent how the motor would truly behave if it was only experiencing continued degradation. The sampling size was also a concern when viewing the data sets. Some classes had much larger

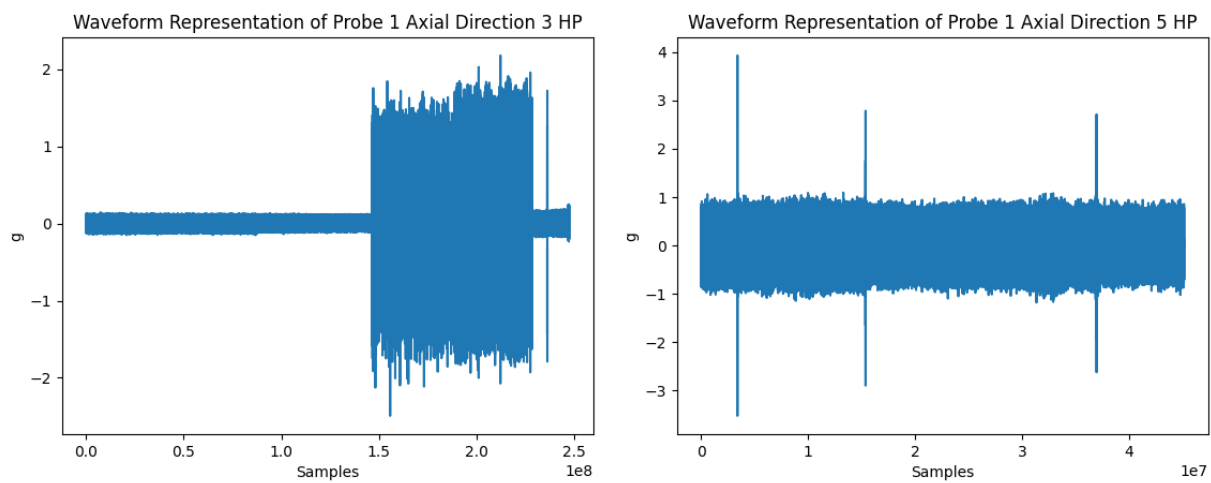


Figure 4.13: 3 HP (Left) and 5 HP (Right) Probe 1 Raw Waveform.

sample sizes than others. This becomes a problem when using deep learning models as they require large sample sizes and do not respond particularly well to off-balance data. These issues occurred due to the configurations within the server and the irregularities in how it captured data. Understanding these limitations of the data set helps ensure the results of the transfer learning experiments make sense.

4.2 Implementation Details

All experiments were run on a Windows server with an Intel Xeon Gold 5220R CPU and an NVIDIA RTX A5500. The server contains 128GB of RAM. The preferred code editor was Visual Studio Code. Python 3.9 was used for all data science related ventures. This specific version was used based on the dependency it had with respect to the use of the GPU and Tensorflow. Numpy, Pandas, and Sklearn were used to manage and format the data. Tensorflow/keras 2.10 was used to develop the deep learning models in conjunction with the Cudatoolkit 11.2 and cudnn 8.1 to enable GPU support. The matplotlib package was used to plot the output of the different models and provide visualization of the data.

4.3 Network Structures

A Multilayer Perceptron (MLP) model was chosen as the first model to test. The structure of the network was chosen by a standard parameter search process and is shown in table 4.6. This process tested different numbers of layers and neurons per layer. It also tested different activation functions. The search showed that the best activation function was the linear activation function. It also showed that two layers with 16 and 128 neurons produced the best results. These results were less than one percent improvement when compared with most other neuron sizes, but it still showed improvement so this structure was chosen. All model tests were run with the ADAM optimizer, cross-entropy loss function, and the primary metric being accuracy.

The CNN-LSTM structure found by the search is presented in table 4.7. This model had substantially more parameters than the MLP model with nearly six million parameters.

Table 4.6: MLP network structure.

Layer	Activation Function	Parameters
Fully Connected	Linear	16 neurons
Fully Connected	Linear	128 neurons
Classification	Softmax	

During the search process, it was shown that dropout was only influential with fully connected layers. Dropout in other parts of the model, particularly the CNN layers, showed reduced performance of the model. After a couple of iterations of the search, it was determined that the first layers should be convolutional due to their significant skill in extracting generalizable features. Max pooling layers were used to reduce the number of parameters, such that the model was not too cumbersome for the GPU. The search also showed that the number of LSTM layers did not significantly impact the performance of the model; however, it was important to use an LSTM layer since it was shown to produce better results than only using convolutional and fully connected layers.

4.3.1 Model Input

For the input of the models, multiple different data inputs will be tested. The main data that the models will use is the feature data. This feature data will come in the form of the vibrational data combined with the temperature data and the MCSA data. The feature data for the vibrational data has 42 total features but with the difference between Probe 1's data in the 3 HP tests shown in section 4.1.2, there will only be 35 features. These 35 features are combined with the four temperature readings to create an input of 39 total features for the vibrational tests. In addition to the vibrational feature data being used, the raw waveform data will also be used as an input to the CNN-LSTM model. It will only be tested with this model as this network structure allows for larger, more complex data input, whereas an MLP is not well-known for that capability. The raw waveform input will come with five features, one for each accelerometer reading except for Probe 1. For the MCSA data, only the feature data will be used as input to the models. The MCSA data contains 38 total features.

4.3.2 Evaluation Criteria

The primary evaluation metric used in this analysis is accuracy. Accuracy is commonly employed in classification problems to assess model performance by indicating the proportion of test samples that the model correctly classified, which is often the primary objective of deep learning classification models. In data science, accuracy is defined as the total number of

Table 4.7: CNN-LSTM network structure.

Layer	Activation Function	Parameters
Convolutional 1D	relu	64 filters, 2x1 filter size
Max Pooling 1D	N/A	1x1 window
Convolutional 1D	relu	64 filters, 2x1 filter size
Max Pooling 1D	N/A	1x1 window
LSTM	tanh	512 units
Flatten	N/A	N/A
Fully Connected	relu	256 neurons
Dropout	N/A	0.5 rate
Classification	Softmax	

correct predictions divided by the total number of predictions made. This is mathematically represented by Eq. 4.9.

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{All predictions}} \quad (4.9)$$

For the incremental transfer approach, the accuracy of the transfer learning model is compared to a model with the same network structure that is trained on the same train-test split used for the transfer model. This comparison simulates the challenge of data availability in the second domain. By using the same split, the effect of knowledge transfer can be evaluated by understanding how well a standard modeling approach performs with limited data availability.

In addition to accuracy, the incremental approach will also be evaluated based on the training times of the fine-tuning process. Literature indicates that one of the primary benefits of transfer learning is its potential to reduce model training time. Therefore, the training times for fine-tuning will be compared to those of a standard model trained on the same data split.

4.4 Results

The results of this study follow the following structure: First, they are separated based on their model structure. From there, both models are broken down into a vibrational data section and an MCSA data section. These sections report the findings from both the zero-shot transfer learning approach and the incremental addition of the training data approach.

The zero-shot approach refers to a form of transfer learning that takes a base model in one domain and then tests it with data from a similar domain. In the study at hand, the base model will either be trained using 3 or 5 HP data and then tested on the opposite domain without any retraining.

The incremental addition of the training data approach begins with training a base model in the base domain. This base model is trained using an 80/5/15 split of data, with 5% allocated for validation during training. The validation step ensures that the model is progressing towards a more generalizable state. The base model is then retrained by updating

the output layer and potentially other outer layers of the network with a small amount of data from the new domain. Consequently, the base model is first trained on one dataset and then incrementally retrained with increasing amounts of data from the new domain for each test. The amount of data used for retraining is specified as a percentage of the total dataset. For example, when the incremental approach specifies using 5% training data, it refers to using 5% of the entire dataset. These percentages vary depending on the dataset being used. The retrained model is then tested on the 15% of data that was reserved during training. To ensure reliable results, the same test set is used for all tests. The performance of the transfer learning model is compared to a network with the same structure trained and tested only on the new domain, using the same amount of data permitted for training the transfer network.

4.4.1 MLP

Zero-Shot

For the zero-shot technique using the MLP, the following approach was implemented: First, the model is trained and validated on one dataset. For example, the base model may be trained and validated on the 3 HP vibrational data. Subsequently, this base model is used to predict outcomes based on the corresponding dataset from the other HP category. Following the example, the 3 HP vibrational base model would be used to predict results based on the 5 HP vibrational data. This approach assumes significant similarities between the domains. Given that the data come from similar motors (3 HP and 5 HP), substantial results were expected. However, as shown in Table 4.8, this expectation was not met.

Given that this classification problem involves five classes, Table 4.8 shows that the zero-shot technique effectively selects only one class, performing no better than random guessing. These results are disappointing, as some level of similarity between the datasets was anticipated, suggesting that retraining might not be necessary. The only combination that performed better than random guessing was the 3 HP Base/5 HP Testing for vibrational data, and even then, it only slightly outperformed the others.

Table 4.8: Accuracies for Zero-Shot technique using an MLP.

	Vibration Accuracy	MCSA Accuracy
3 HP Base/5 HP Testing	0.214	0.200
5 HP Base/3 HP Testing	0.200	0.200

Vibration Incremental Approach

For the incremental approach using the MLP structure, the only layer that was replaced and retrained for the transfer learning model was the output layer. The input and dense layers maintained the same weights from the base model, while the output layer was retrained on the new HP data.

The results for the 3 HP base model, incrementally retrained on the 5 HP data, are shown in Figures 4.14 and 4.15. The accuracy of the transfer learning model remains strongly consistent with the addition of new data, whereas the comparison model exhibits a more sporadic trend and a less consistent view of the data. Although the comparison model generally achieves higher accuracy, the erratic nature of its trend suggests the presence of overfitting. This overfitting contrasts sharply with the precision and stability of the transfer learning model.

In Figure 4.15, the training run times for the MLP using transfer learning and the standard model are compared. The run times for the transfer learning endeavors show a more consistent trend compared to the standard model. This can be attributed to the fact that transfer learning experiments begin their convergence from the same starting point based on the initially trained layers, while the standard model starts from random points on the gradient. From this, we can conclude that the transfer learning model benefits from a strong starting point on the gradient but tends to get stuck in finding a similar local minimum.

Figure 4.16 shows the accuracy of the incremental approach of a standard MLP on the 3 HP data and a 5 HP base model retrained to fit the 3 HP data. The standard MLP consistently performed better with the additional training data. The difference in accuracy between the two approaches remains consistent throughout the testing. The variable behavior observed with the standard model in the 3 HP base testing is not evident in this test. Instead, training from scratch with small amounts of data provides consistently strong results.

The training times shown in Figure 4.17 are consistent with the findings from the 3 HP base tests. The standard model exhibited longer and more variable training times,

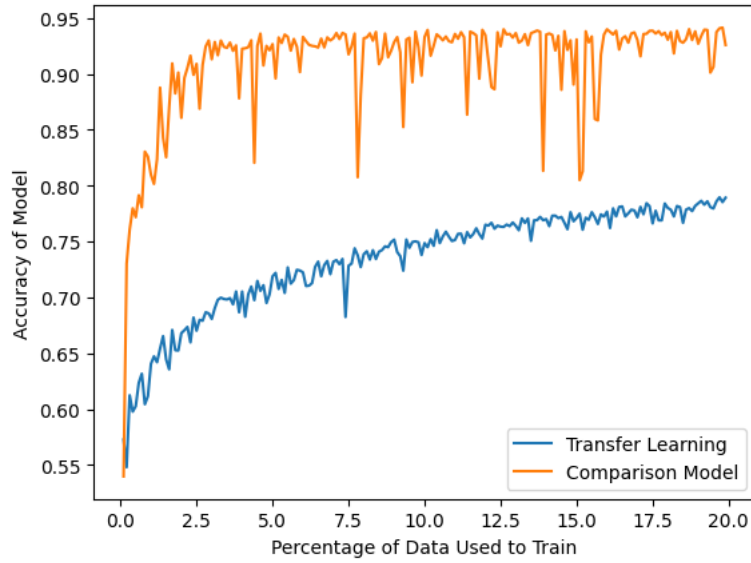


Figure 4.14: MLP accuracy using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.



Figure 4.15: MLP training time using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.

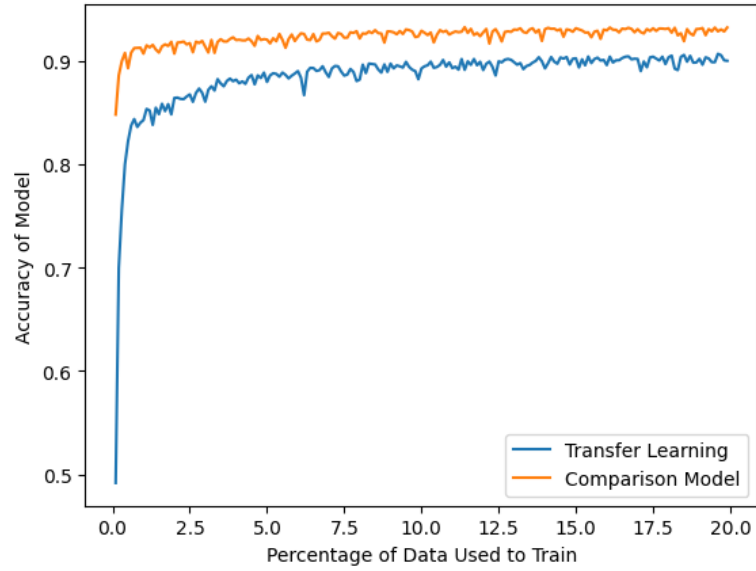


Figure 4.16: MLP accuracy using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.

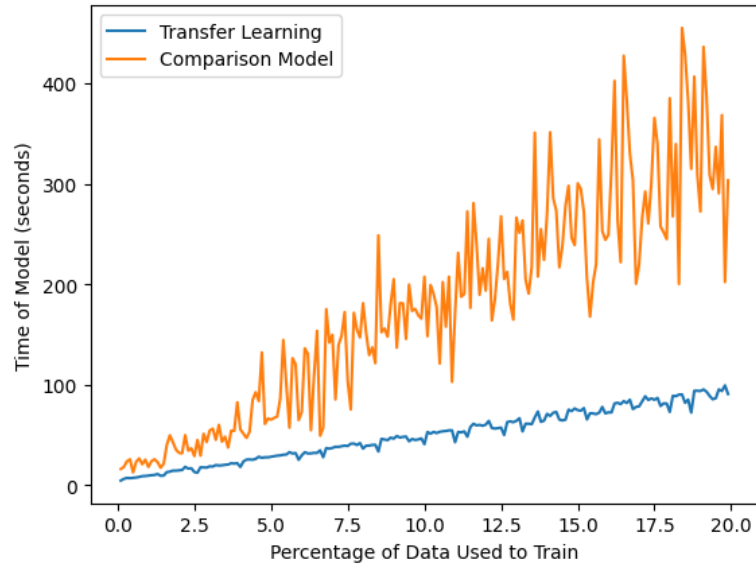


Figure 4.17: MLP training time using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.

whereas the transfer learning model maintained a more gradual slope with less variability in training times. Analyzing the two approaches with an MLP reveals that the transfer learning approach provides solid results in terms of accuracy and training times. However, when compared with the standard model, training a new network from scratch can yield higher accuracy at the expense of longer training times.

MCSA Incremental Approach

The MCSA incremental approach follows the same structure as the vibration incremental approach. The main difference is the type of data used for the tests. The following results are based on the MCSA feature data.

Figure 4.18 shows the accuracy of the MCSA 3 HP base model using the 5 HP transfer learning approach. The graph is cluttered with significant variability between different data splits. However, it can be observed that the standard model approach maintains a trend of higher accuracy and lower variability compared to the transfer learning approach. The overall trend of the standard model is approximately 20% better than that of the transfer learning approach. The 3 HP base model does not generalize well, resulting in difficulties in producing robust and strong results.

The results shown in Figure 4.19 depict the training times associated with the different data splits. The transfer learning times followed a more linear trend. It can again be seen that the main advantage of the transfer learning approach with an MLP is the reduction of variability and overall training time, which is more variable in the standard model. In this scenario, the primary benefit of the transfer learning approach is observed in the training times rather than in the accuracy of the models.

Figure 4.20 shows the accuracy of the MCSA 5 HP base model using the 3 HP transfer learning approach. Similar to the 3 HP base experiment with the MCSA data, the 5 HP base does not perform well in its transfer of knowledge. Both the accuracy and the variability of the transfer technique with the 5 HP base are worse than the standard model. These results are consistent with the 3 HP base experiment. This indicates that the MLP network structure does not generalize the MCSA data well for either dataset.

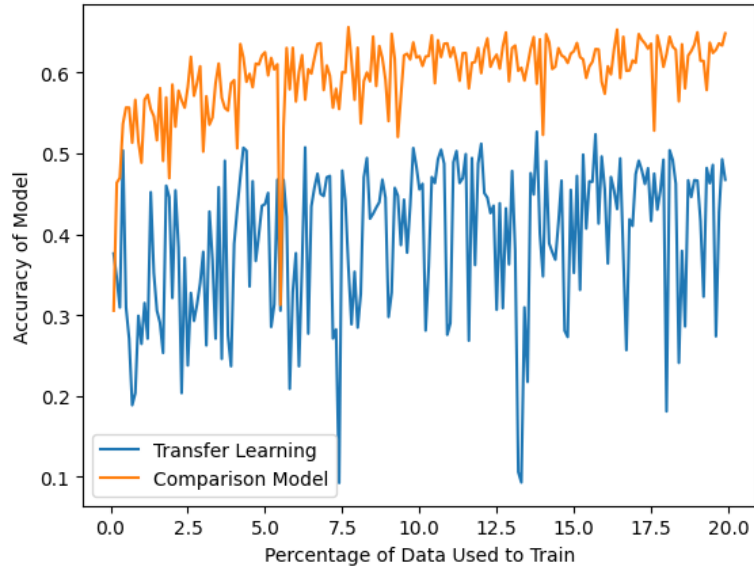


Figure 4.18: MLP accuracy using incremental training data approach with a 3 HP base model and 5 HP transfer learning using MCSA data.

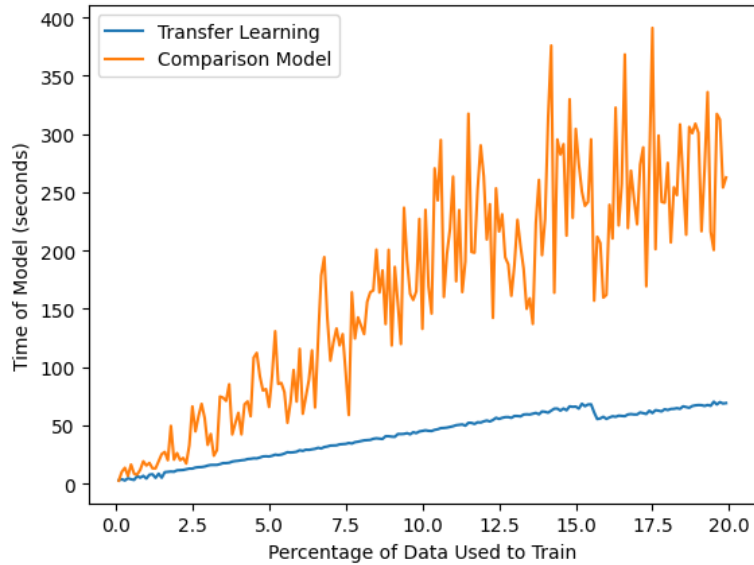


Figure 4.19: MLP training time using incremental training data approach with a 3 HP base model and 5 HP transfer learning using MCSA data.

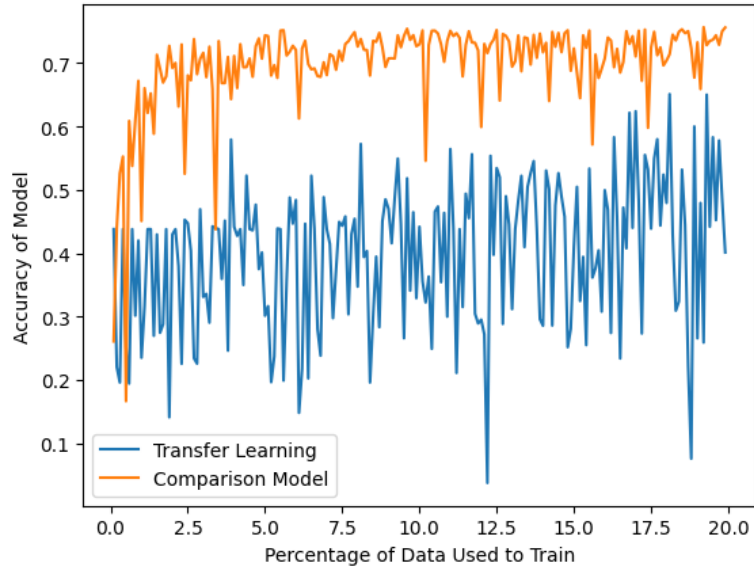


Figure 4.20: MLP accuracy using incremental training data approach with a 5 HP base model and 3 HP transfer learning using MCSA data.

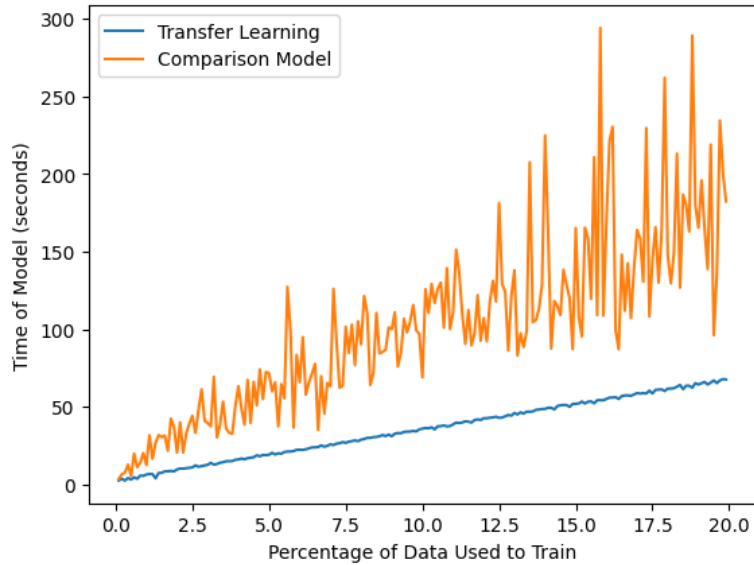


Figure 4.21: MLP training time using incremental training data approach with a 5 HP base model and 3 HP transfer learning using MCSA data.

Although the model’s testing results were not better than the standard model, Figure 4.21 tells a different story regarding training times for both techniques. The training times for the transfer learning technique significantly outperform those of the standard model. The rate at which the transfer learning technique’s training time increases with more data is much lower than that of the standard model, and it maintains less variability with each run. This trend in training times is promising, even though the accuracy results of the transfer learning technique were not.

Based on the results from the MLP transfer learning experiments, it can be observed that the MLP overfits the base data, resulting in a base model that does not generalize well and, therefore, shows weak transfer learning results. MLPs are not typically known for their generalizability; rather, their performance can be influenced by several factors that contribute to overfitting. It is likely that the depth of the data enhances the model’s tendency to overfit. In this case study, an MLP may not be the optimal architectural choice.

4.4.2 CNN-LSTM

Zero-Shot

Consistent with the findings presented in Section 4.4.1, the zero-shot technique combined with the CNN-LSTM performs close to, if not exactly, as guessing, as shown in Table 4.9. These results reinforce the idea that, despite the semantic similarity between the two datasets, they exhibit different trends that prevent solid results without retraining the model with some new data. This suggests that zero-shot transfer learning relies more on the similarities between the domains of the two datasets than on the choice of network structure. Although it might be expected that the initial CNN layers could capture generalizable features consistent across similar domains, this does not appear to be the case for these datasets. Based on the conclusions drawn in Section 4.1.2, it is not surprising that the zero-shot transfer technique yields these results.

Table 4.9: Accuracies for Zero-Shot technique using a CNN-LSTM.

	Vibration Accuracy	MCSA Accuracy
3 HP Base/5 HP Testing	0.1922	0.2000
5 HP Base/3 HP Testing	0.2169	0.2000

Vibration Incremental Approach

For the incremental approach using the CNN-LSTM, the only layers that were not replaced were CNN layers. The first two CNN layers were left with only the base data, while the LSTM and dense layers following the feature-extracting CNN layers were trained to fit the new data. The experiments were run first with the vibrational feature data and then with the vibrational raw waveform data.

Figure 4.22 shows the accuracy of the transfer learning CNN-LSTM and the standard CNN-LSTM using the vibrational feature data. The figure demonstrates that the transfer learning approach significantly outperforms the standard approach with low levels of data. The transfer learning model achieves maximum accuracy more quickly than the standard model and maintains a 10% advantage throughout the test. In addition to outperforming the standard model, the transfer learning approach also reduces the variability of test results, indicating a better fit for the data. This consistency suggests that the transfer learning model remains robust even with minimal training data.

Figure 4.23 illustrates the difference in training times between the transfer learning method and the standard modeling approach. Contrary to the findings with MLP and common expectations about transfer learning in this context, the transfer learning approach takes considerably longer than the standard approach. The trend indicates that transfer learning training times increase at a faster rate compared to the standard model training times.

In comparison with the 5 HP base transfer learning model, Figure 4.24 displays the transfer learning attempt with a 3 HP base model. The trends are similar between the two bases. The transfer learning approach significantly outperforms the standard approach, demonstrating the CNN-LSTM's effectiveness in a predictive maintenance setting. When compared to the 5 HP base model tests, the 3 HP base model shows greater improvements over the standard model. The consistency of the transfer learning approach compared to the standard model remains evident in this experiment's results.

Similar to the findings with the 5 HP base model, the training times for the 3 HP base transfer learning model, as shown in Figure 4.25, are worse than those for the standard

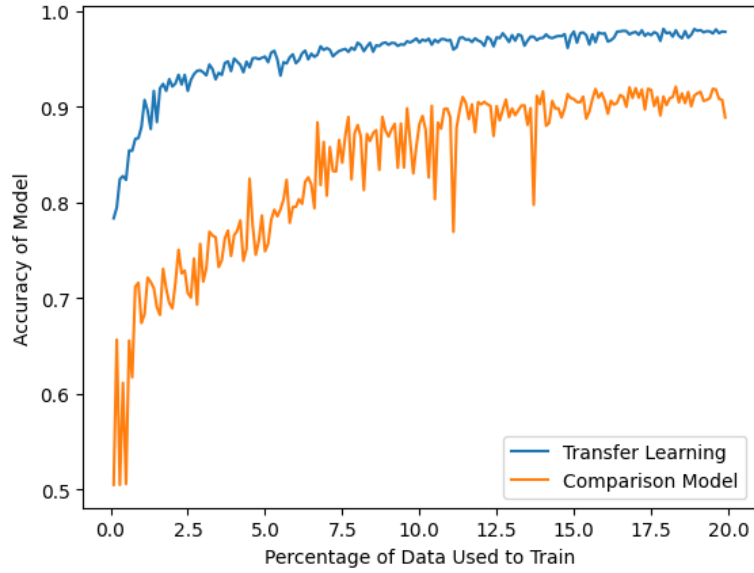


Figure 4.22: CNN-LSTM accuracy using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.

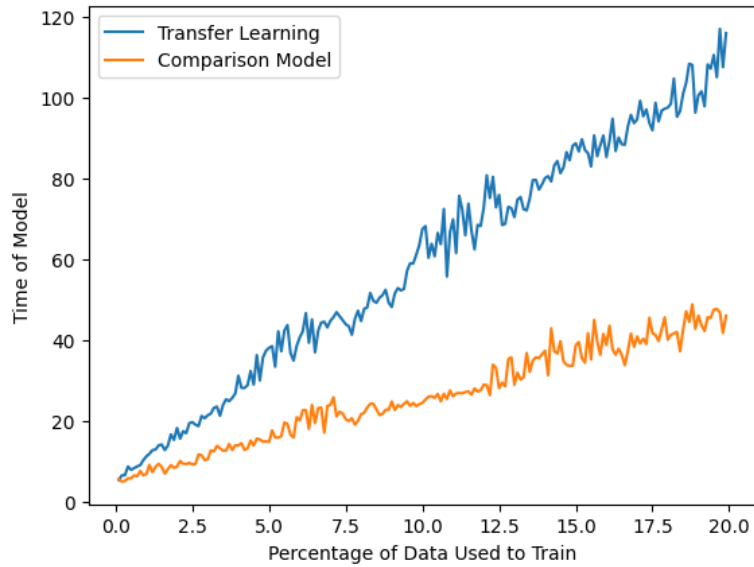


Figure 4.23: CNN-LSTM training time using incremental training data approach with a 5 HP vibrational base model and 3 HP transfer learning.

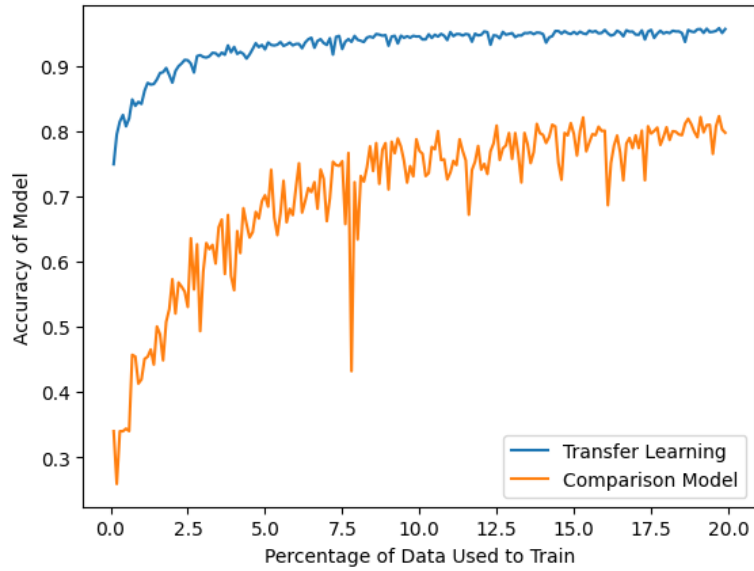


Figure 4.24: CNN-LSTM accuracy using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.

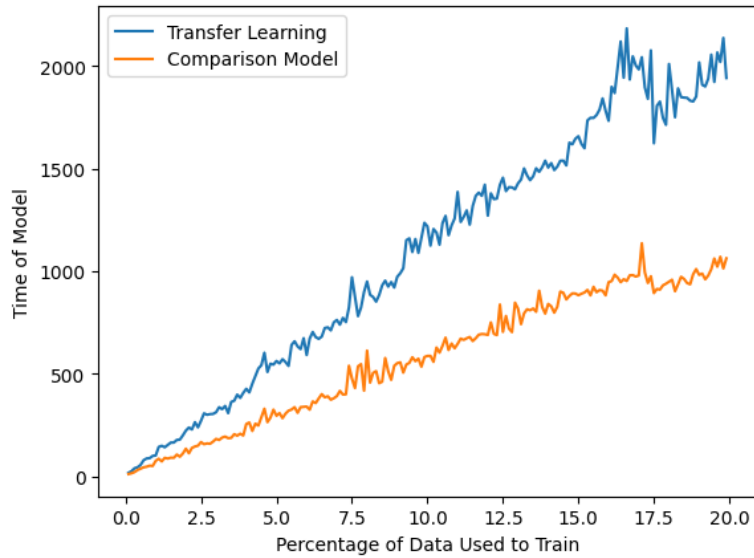


Figure 4.25: CNN-LSTM training time using incremental training data approach with a 3 HP vibrational base model and 5 HP transfer learning.

approach. These results present another instance of unexpected outcomes for transfer learning. However, the improvements in accuracy still justify the use of the CNN-LSTM transfer learning approach, despite the significantly longer training times compared to the standard approach.

To evaluate the performance of the transfer learning technique on different datasets, the same experiments were conducted with raw waveform vibrational data. This data represents tests from the same motors but is much larger and lacks well-defined features. Figure 4.26 shows the results of the 3 HP base model experiment using this new dataset. Although the overall accuracy of the models is not strong, the transfer learning technique demonstrates better performance with less of the new domain's data compared to the standard model. In addition to better initial accuracy, the transfer learning technique exhibits less variability than the standard model. However, the transfer learning technique's learning curve seems to plateau more quickly than the standard model's. Therefore, as more data from the new domain is added, the standard model is expected to eventually outperform the transfer learning technique.

Figure 4.27 reveals a different trend in training times compared to previous tests, based on the different data splits. This experiment shows that the transfer learning technique has significantly larger training times than the standard model. While the transfer learning approach starts with high training times, it quickly stabilizes to a non-increasing trend. In contrast, the standard model exhibits a slightly increasing trend from the beginning.

The 5 HP base model exhibits results similar to those of the 3 HP base model tests but with a quicker transition to the standard model outperforming the transfer learning model. Figure 4.28 also shows a more pronounced increase in the accuracy of the standard model, whereas the transfer learning technique appears to get stuck in a local minimum, leveling out quickly and showing little to no improvement with additional data.

The timing trends for both the transfer learning with a 5 HP base and the standard model are consistent with those observed in the 3 HP base tests. Figure 4.29 illustrates that the transfer learning technique has higher training times compared to the standard model. This time, there is increased variability in the transfer learning training times, with larger

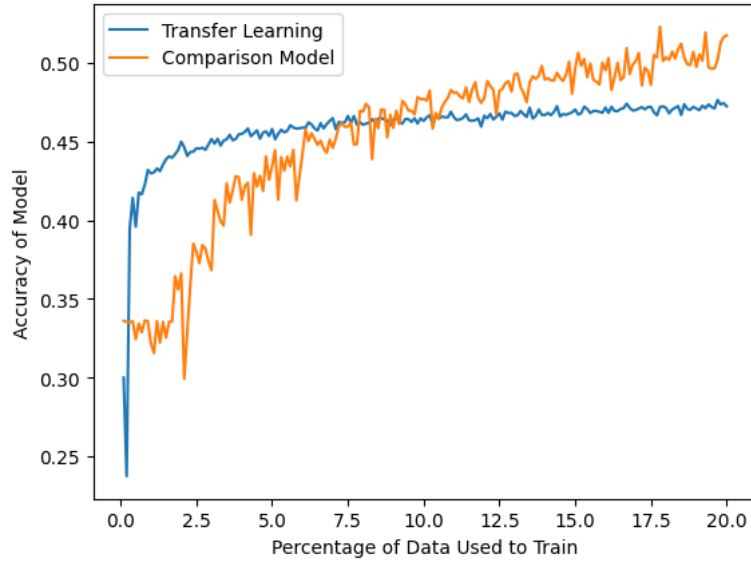


Figure 4.26: CNN-LSTM accuracy using incremental training data approach with a 3 HP base model and 5 HP transfer learning on the raw waveform data.

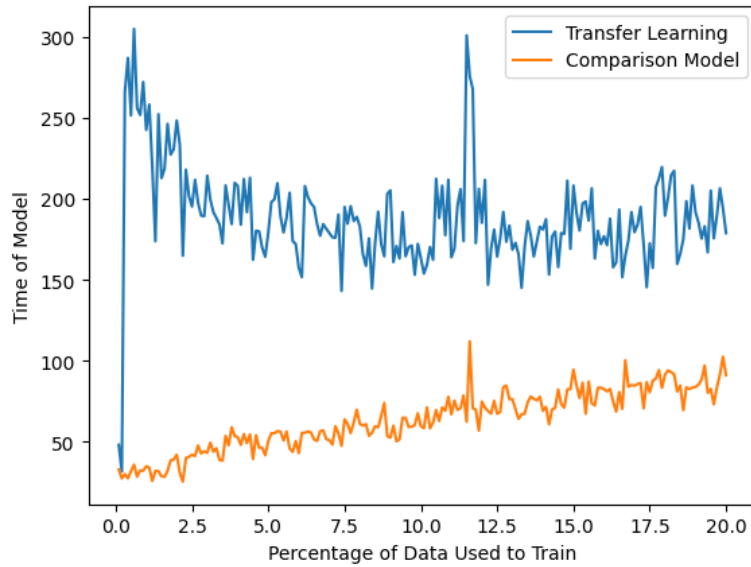


Figure 4.27: CNN-LSTM training time using incremental training data approach with a 3 HP base model and 5 HP transfer learning on the raw waveform data.

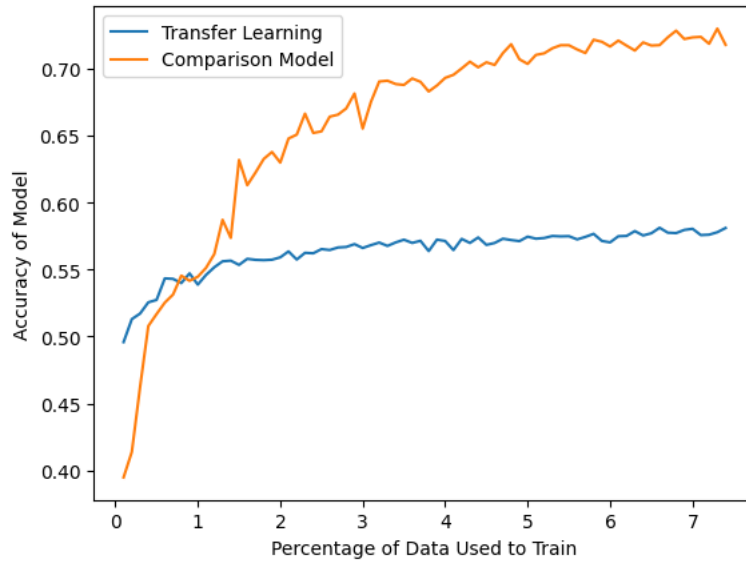


Figure 4.28: CNN-LSTM accuracy using incremental training data approach with a 5 HP base model and 3 HP transfer learning on the raw waveform data.

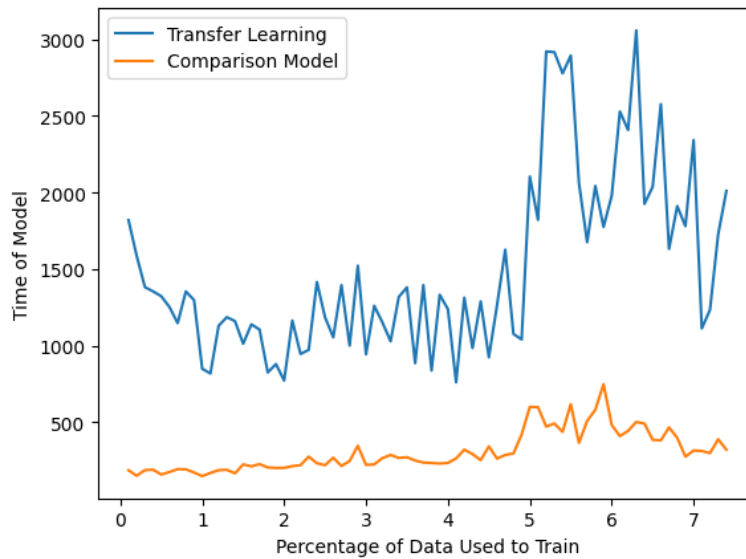


Figure 4.29: CNN-LSTM training time using incremental training data approach with a 5 HP base model and 3 HP transfer learning on the raw waveform data.

spikes toward the end of the tests. While the standard model again shows better training times, it also exhibits less of an increasing trend compared to the previous experiment.

MCSA Incremental Approach

The MCSA experiments for the CNN-LSTM follow the same retraining paradigm as the vibrational feature tests. The LSTM and Dense layers are retrained, while the CNN layers are left from the base model.

Figure 4.30 illustrates both the accuracies and training times for the transfer learning method and the standard CNN-LSTM approach using the 3 HP data as a base. When predicting based on the 5 HP MCSA data, it is evident that the standard CNN-LSTM approach is ineffective until at least 10% of the data is used. In contrast, the transfer learning approach quickly achieves maximum accuracy and reduces variability in the results. The training times for both approaches remain relatively similar with increasing training data. However, at 14% training data, both approaches experience a significant drop in training times, which affects only the accuracy of the standard model. Overall, this experiment demonstrates that transfer learning can be particularly effective in cases where target domain training data is limited.

Figure 4.31 initially shows promising results for the transfer learning approach. Compared to the standard approach, transfer learning provides better accuracy, suggesting its suitability for this particular situation. Additionally, the transfer learning approach demonstrates greater consistency, aligning with findings from other experiments, while the standard approach remains highly volatile throughout the tests. Even with larger datasets, the standard approach struggles to match the accuracy achieved by transfer learning. However, the transfer learning approach again exhibits longer training times compared to the standard model. Overall, this exemplifies the potential benefits of transfer learning in practice.

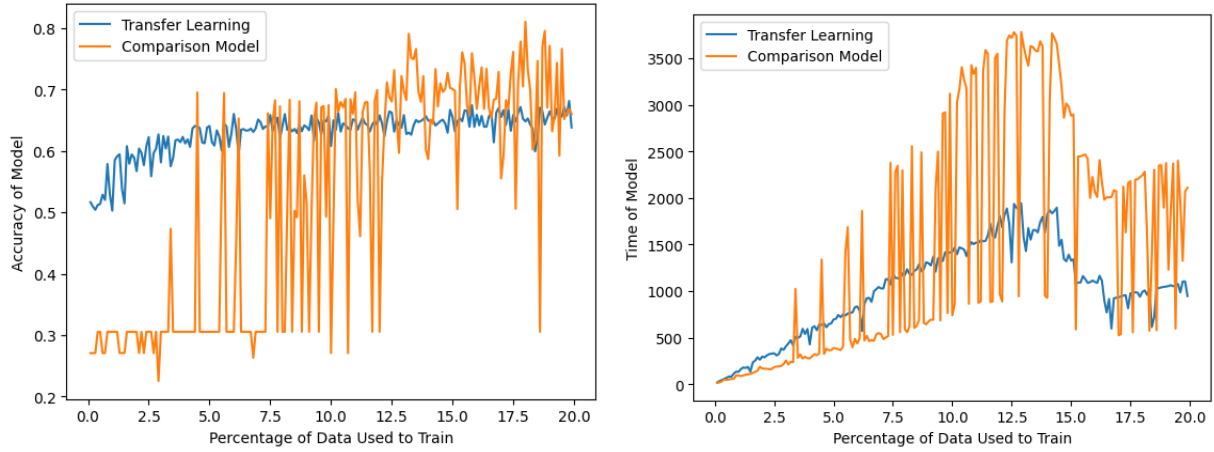


Figure 4.30: CNN-LSTM accuracy and training time using incremental training data approach with a 3 HP base model and 5 HP transfer learning using MCSA data.

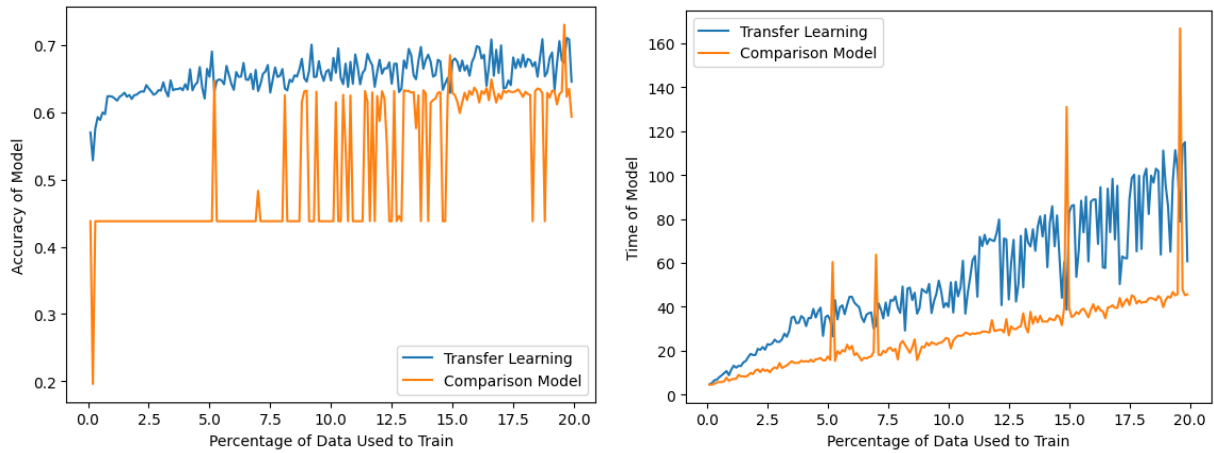


Figure 4.31: CNN-LSTM accuracy and training time using incremental training data approach with a 5 HP base model and 3 HP transfer learning using MCSA data.

Chapter 5

Summary and Future Work

5.1 Summary

Based on the Zero-shot MLP experiment results, there was no notable significance shown. All experiments showed behavior consistent with guessing only one class of the classification problem. From this, it can be concluded that even though the domains were closely related, with the only main difference being the HP of the motors, that Zero-shot with an MLP/CNN-LSTM will not provide substantial results. While it can be concluded that the technique does not work for this specific problem, it can not be concluded that this technique should be completely ignored in future research. Having seen and analyzed the differences in the two domains of data that were presented by mechanical and human error, it would be non-scholarly to completely dismiss this technique. However, it can be concluded that in this specific case of predictive maintenance, significant transfer learning results can only be obtained by retraining parts of the network with data from the new domain. It is also very likely that in real-world applications of transfer learning, the two domains would likely contain significant amounts of noise such that the differences in data from different machines would likely result in similar results to what is presented in this thesis.

Although there was little to no promise shown in the results of the Zero-shot transfer learning approach, there are significant findings using the incremental approach that will be able to guide further research based on the idea of using transfer learning on tabular predictive maintenance data ventures. Based on the results presented in Chapter 4, the

incremental approach showed promising results in a couple of the different scenarios tested. These results showed that transfer learning has the capabilities to reduce variability in the training of models, reduce training times, and converge to the maximum accuracy with less data than a standard modeling approach would. On the other hand, it showed that the transfer learning approach could also increase training times and converge to local minimums instead of global minimums. By understanding transfer learning's ability to perform both better and worse than a standard modeling approach, it should be concluded that the incremental transfer learning approach is case-specific. Its ability to perform well depends greatly on the architecture of the model as well as the quality of the data and the inferences made by the base model. This can easily be seen between the use of the MLP and the CNN-LSTM. The MLP architecture showed strong capabilities in reducing training times and their variability while performing worse in terms of accuracy. The CNN-LSTM showed opposing results to the MLP. It performed much better in terms of accuracy but lost out to the standard approach in terms of training times. So to apply transfer learning in this domain successfully, great attention must be given to the quality of the two data sources and the structure of the network that will be chosen.

The application of transfer learning for these ventures seems to rely heavily on the amount of resources the user has to dedicate to the venture. In most cases, it seems worthwhile to perform transfer learning, especially in situations with small amounts of data in the target domain. With transfer learning's low variability and reliance on less data to find the global minimum, it becomes an extremely useful tool when data and computational resources are low. However, when large amounts of data are available in the target domain, transfer learning loses its edge and will likely not be worth the extra effort when compared with using a standard deep learning approach. Overall, the use of tabular data can be shown to be a viable input for transfer learning; however, transfer learning's impact seems to be less with tabular data than in other research areas that mainly focus on natural language processing and image recognition. It can be concluded that deep learning techniques have a slightly harder time generalizing to numerical input with little spatial relation. Instead, deep learning techniques used with spatial data, such as images, have more capability to identify spatial relationships.

5.2 Limitations & Future Work

A noticeable limitation of this thesis is the fact that when using the incremental approach for transfer learning, the target domain's data, although small, contained all classes of fault that needed to be classified. In a real-world situation, it is unlikely that all classes would be present with such small amounts of data. Initial experimentation, which was explored in testing but not presented in this thesis, experimented with the notion of only having baseline data in the new domain. Simple transfer learning approaches were tested but showed no significant results, as the new model only selected the base class in the new domain. This topic in itself could be seen as an avenue to create a whole new thesis, however, the basic introduction to transfer learning for tabular predictive maintenance data was deemed to be worthy of further research and thus this thesis maintained that focus.

Another notable limitation that was explored in depth in Section 4.1.2 is the quality of the data captured by the experiments. While real-world data is commonly found with noise and other issues that impede the efforts of predictive maintenance techniques, data produced in a controlled environment should be free of most errors or noise unless it is specifically added for a purpose. In this case, the errors present in the data actually detract from the data's ability to simulate real machine processes. The human errors were present due to the fact that the tests required constant readjustment of the motor to simulate the faults. During these constant adjustments, small differences lead to big differences and errors in the testing. Instead, in a real-world setting, most machines that are in constant use or are automated would not produce data that contained large amounts of human error, such as these data sets consist of. Future experiments should strive to maintain data quality with constant checks to see whether data is being collected properly and makes sense based on the previous data acquired.

To further this research it is important to view what steps could be taken next. An important next step would be to redo this thesis with data captured in a real-world setting with similar machines. This would not only validate the results of this thesis but would also play a pivotal role in understanding the capabilities and limitations of transfer learning for predictive maintenance in the real world. While this thesis shows the promise that is

available through this research, validating it with real-world data would provide a stronger understanding of its practical uses in industrial applications. Another important next step would be to take a deeper dive into the different network architectures available for this problem. Due to constraints in computational resources and time, the network architectures were not investigated holistically. Based on the conclusion that network structure plays a large role in transfer learning, it would be extremely beneficial to conduct an entire thesis on how different architectures affect transfer learning in predictive maintenance ventures.

Another transfer learning technique that was considered during the development of this thesis but was not experimented with was the idea of using transfer learning of one type of fault to predict a different but similar fault in the same machine. This idea utilizes the notion of having extensive fault data in one fault type with minimal data in another. The target domains, in this case, share even more similarities than in the experiments conducted in this thesis since the machine itself does not change. Further research into this transfer learning technique would be beneficial to create a more robust understanding of transfer learning in predictive maintenance ventures.

Bibliography

- [1] Adams, S., Malinowski, M., Heddy, G., Choo, B., and Beling, P. A. (2017). The wear methodology for prognostics and health management implementation in manufacturing. *Journal of Manufacturing Systems*, 45:82–96. [1](#), [2](#)
- [2] Al-Raheem, K. F. and Abdul-Karem, W. (2010). Rolling bearing fault diagnostics using artificial neural networks based on laplace wavelet analysis. *International Journal of Engineering, Science and Technology*, 2(6). [11](#)
- [3] Aydin, O. and Guldamlasioglu, S. (2017). Using lstm networks to predict engine condition on large scale data processing framework. In *2017 4th International Conference on Electrical and Electronic Engineering (ICEEE)*, pages 281–285. [10](#)
- [4] Azari, M. S., Flammini, F., Santini, S., and Caporuscio, M. (2023). A systematic literature review on transfer learning for predictive maintenance in industry 4.0. *IEEE Access*, 11:12887–12910. [12](#), [13](#)
- [5] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32. [7](#)
- [6] Carvalho, T. P., Soares, F. A. A. M. N., Vita, R., da P. Francisco, R., Basto, J. P., and Alcalá, S. G. S. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, 137:106024. [7](#), [8](#), [9](#)
- [7] Chang, M.-H., Sandborn, P., Pecht, M., Yung, W. K., and Wang, W. (2015). A return on investment analysis of applying health monitoring to led lighting systems. *Microelectronics Reliability*, 55(3):527–537. [3](#)

- [8] Chen, Z., Gryllias, K., and Li, W. (2019). Intelligent fault diagnosis for rotary machinery using transferable convolutional neural network. *IEEE Transactions on Industrial Informatics*, 16(1):339–349. [13](#)
- [9] Dai, W., Yang, Q., Xue, G.-R., and Yu, Y. (2008). Self-taught clustering. ICML '08, page 200–207, New York, NY, USA. Association for Computing Machinery. [12](#)
- [10] Dalenogare, L. S., Benitez, G. B., Ayala, N. F., and Frank, A. G. (2018). The expected contribution of industry 4.0 technologies for industrial performance. *International Journal of Production Economics*, 204:383–394. [6](#)
- [11] Daume III, H. and Marcu, D. (2006). Domain adaptation for statistical classifiers. *Journal of artificial Intelligence research*, 26:101–126. [12](#)
- [12] Durbhaka, G. K. and Selvaraj, B. (2016). Predictive maintenance for wind turbine diagnostics using vibration signal analysis based on collaborative recommendation approach. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1839–1842. [8](#)
- [13] Eke, S., Aka-Ngnui, T., Clerc, G., and Fofana, I. (2017). Characterization of the operating periods of a power transformer by clustering the dissolved gas data. volume 2017-January, page 298 – 303. Cited by: 8. [8](#)
- [14] Fischer, D., Moder, P., and Ehm, H. (2021). Investigation of predictive maintenance for semiconductor manufacturing and its impacts on the supply chain. In *2021 22nd IEEE International Conference on Industrial Technology (ICIT)*, volume 1, pages 1409–1416. IEEE. [3](#)
- [15] Gohel, H. A., Upadhyay, H., Lagos, L., Cooper, K., and Sanzetenea, A. (2020). Predictive maintenance architecture development for nuclear infrastructure using machine learning. *Nuclear Engineering and Technology*, 52(7):1436–1442. [8](#)
- [16] Grudin, J. (1988). Why csw applications fail: problems in the design and evaluation of organizational interfaces. In *Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, pages 85–93. [16](#)

- [17] Han, T., Liu, C., Yang, W., and Jiang, D. (2019). Learning transferable features in deep convolutional neural networks for diagnosing unseen machine conditions. *ISA transactions*, 93:341–353. [13](#)
- [18] Hsu, J.-Y., Wang, Y.-F., Lin, K.-C., Chen, M.-Y., and Hsu, J. H.-Y. (2020). Wind turbine fault diagnosis and predictive maintenance through statistical process control and machine learning. *IEEE Access*, 8:23427–23439. [7](#)
- [19] Huang*, R., Xi, L., Lee, J., and Liu, C. (2005). The framework, impact and commercial prospects of a new predictive maintenance system: intelligent maintenance system. *Production Planning & Control*, 16(7):652–664. [3](#)
- [20] Kulkarni, K., Devi, U., Sirighee, A., Hazra, J., and Rao, P. (2018). Predictive maintenance for supermarket refrigeration systems using only case temperature data. In *2018 Annual American Control Conference (ACC)*, pages 4640–4645. [7](#)
- [21] Li, X., Ding, Q., and Sun, J.-Q. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety*, 172:1–11. [11](#)
- [22] Liu, Q. and Huang, C. (2019). A fault diagnosis method based on transfer convolutional neural networks. *IEEE Access*, 7:171423–171430. [14](#)
- [23] Mahlamäki, K. and Nieminen, M. (2020). Analysis of manual data collection in maintenance context. *Journal of Quality in Maintenance Engineering*, 26(1):104–119. [15](#)
- [24] Meng, H., Liu, X., Xing, J., and Zio, E. (2022). A method for economic evaluation of predictive maintenance technologies by integrating system dynamics and evolutionary game modelling. *Reliability Engineering & System Safety*, 222:108424. [3](#)
- [25] Mo, Y. J., Kim, J., Kim, J.-K., Mohaisen, A., and Lee, W. (2017). Performance of deep learning computation with tensorflow software library in gpu-capable multi-core computing platforms. In *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 240–242. IEEE. [17](#)

- [26] Nguyen, K. T. and Medjaher, K. (2019). A new dynamic predictive maintenance framework using deep learning for failure prognostics. *Reliability Engineering & System Safety*, 188:251–262. [10](#)
- [27] Nowakowski, T., Tubis, A., and Werbińska-Wojciechowska, S. (2019). Evolution of technical systems maintenance approaches—review and a case study. In *Intelligent Systems in Production Engineering and Maintenance*, pages 161–174. Springer. [6](#)
- [28] Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359. [12](#), [13](#), [14](#)
- [29] Praveenkumar, T., Saimurugan, M., Krishnakumar, P., and Ramachandran, K. (2014). Fault diagnosis of automobile gearbox based on machine learning techniques. *Procedia Engineering*, 97:2092–2098. "12th Global Congress on Manufacturing and Management" GCMM - 2014. [8](#)
- [30] Roy, M., Bose, S. K., Kar, B., Gopalakrishnan, P. K., and Basu, A. (2018). A stacked autoencoder neural network based automated feature extraction method for anomaly detection in on-line condition monitoring. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1501–1507. [10](#)
- [31] Sandtorv, H. A., Hokstad, P., and Thompson, D. W. (1996). Practical experience with a data collection project: the oreda project. *Reliability Engineering & System Safety*, 51(2):159–167. [16](#)
- [32] Serradilla, O., Zugasti, E., Rodriguez, J., and Zurutuza, U. (2022). Deep learning models for predictive maintenance: a survey, comparison, challenges and prospects. *Applied Intelligence*, 52(10):10934–10964. [9](#), [11](#)
- [33] Shin, W., Han, J., and Rhee, W. (2021). Ai-assistance for predictive maintenance of renewable energy systems. *Energy*, 221:119775. [3](#)
- [34] Susto, G. A., Beghi, A., and De Luca, C. (2012). A predictive maintenance system for epitaxy processes based on filtering and prediction techniques. *IEEE Transactions on Semiconductor Manufacturing*, 25(4):638–649. [6](#)

- [35] Tao, S., Zhang, T., Yang, J., Wang, X., and Lu, W. (2015). Bearing fault diagnosis method based on stacked autoencoder and softmax regression. In *2015 34th Chinese Control Conference (CCC)*, pages 6331–6335. [10](#)
- [36] Tercan, H., Guajardo, A., and Meisen, T. (2019). Industrial transfer learning: Boosting machine learning in production. In *2019 IEEE 17th international conference on industrial informatics (INDIN)*, volume 1, pages 274–279. IEEE. [17](#)
- [37] Theissler, A., Pérez-Velázquez, J., Kettelgerdes, M., and Elger, G. (2021). Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry. *Reliability Engineering & System Safety*, 215:107864. [6](#)
- [38] Ullah, I., Yang, F., Khan, R., Liu, L., Yang, H., Gao, B., and Sun, K. (2017). Predictive maintenance of power substation equipment by infrared thermography using a machine-learning approach. *Energies*, 10(12):1987. [11](#)
- [39] Wen, L., Li, X., and Gao, L. (2020). A transfer convolutional neural network for fault diagnosis based on resnet-50. *Neural Computing and Applications*, 32:6111–6124. [14](#)
- [40] Wen, T. and Keyes, R. (2019). Time series anomaly detection using convolutional neural networks and transfer learning. *arXiv preprint arXiv:1905.13628*. [12](#)
- [41] Wu, S.-j., Gebraeel, N., Lawley, M. A., and Yih, Y. (2007). A neural network integrated decision support system for condition-based optimal predictive maintenance policy. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(2):226–236. [7](#)
- [42] Wu, Z., Jiang, H., Zhao, K., and Li, X. (2020). An adaptive deep transfer learning method for bearing fault diagnosis. *Measurement*, 151:107227. [13](#)
- [43] Yu, Y., Si, X., Hu, C., and Zhang, J. (2019). A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270. [10](#)
- [44] Yuan, M., Wu, Y., and Lin, L. (2016). Fault diagnosis and remaining useful life estimation of aero engine using lstm neural network. In *2016 IEEE International Conference on Aircraft Utility Systems (AUS)*, pages 135–140. [10](#)

- [45] Zhang, W., Yang, D., and Wang, H. (2019). Data-driven methods for predictive maintenance of industrial equipment: A survey. *IEEE Systems Journal*, 13(3):2213–2227. [6](#), [7](#)
- [46] Zonta, T., Da Costa, C. A., da Rosa Righi, R., de Lima, M. J., da Trindade, E. S., and Li, G. P. (2020). Predictive maintenance in the industry 4.0: A systematic literature review. *Computers & Industrial Engineering*, 150:106889. [1](#)

Vita

Colter Swanson was born and raised in Knoxville, TN where he attended Knoxville West High School and graduated as a part of the International Baccalaureate Certificate program. From there, he stuck to his roots and continued his education with the University of Tennessee, Knoxville. As a part of the Cook Grand Challenge Honors program, Colter graduated with a 3.96 GPA in Honors Computer Science. Knowing he wanted to continue his education and broaden his horizons, Colter joined Dr. Xueping Li's research team in the Industrial and Systems Engineering Department as a Masters Candidate. During his time as a Masters Candidate under Dr. Li, Colter's research focused heavily on Predictive Maintenance and the application of Transfer Learning to Predictive Maintenance. Working in joint collaboration with Y-12 and Consolidated Nuclear Security (CNS), Colter was able to apply his research to real world engineering problems. Colter graduated with his Masters in Industrial Engineering in August 2024 and is now planning on joining the workforce before considering a PhD.