A Comprehensive Experimental Evaluation of Anomaly Detection Techniques in Aviation Cyber-Physical Systems

Submitted by

Qurrat Ul Ain

I22-0727

Supervised by

Dr. Atif Aftab Ahmed Jilani

Master of Science (Software Engineering)

A thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science (Software Engineering)
at the National University of Computer & Emerging Sciences

Department of Software Engineering
National University of Computer and Emerging Sciences
Islamabad, Pakistan.

June 2024

Plagiarism Undertaking

I take full responsibility for the research work conducted during the Master's Thesis titled ***A Comprehensive Experimental Evaluation of Anomaly Detection Techniques in Aviation Cyber-Physical Systems***. I solemnly declare that the research work presented in the thesis is done solely by me with no significant help from any other person; however, small help, wherever taken, is duly acknowledged. I have also written the complete thesis by myself. Moreover, I have not previously presented this thesis (or substantially similar research work) or any part of the thesis to any other degree-awarding institution within Pakistan or abroad.

I understand that the National University of Computer and Emerging Sciences management has a zero-tolerance policy toward plagiarism. Therefore, I, as an author of the thesis mentioned above, solemnly declare that no portion of my thesis has been plagiarized and that any material used in the thesis from other sources is appropriately referenced. Moreover, the thesis contains only a literal citing of 70 words (total), even by giving a reference, unless I have the written permission of the publisher to do so. Furthermore, the work presented in the thesis is my original work, and I have positively cited the related work of the other researchers by clearly differentiating my work from their relevant work.

I further understand that if I am found guilty of any form of plagiarism in my thesis work even after my graduation, the University reserves the right to revoke my master's degree. Moreover, the University will also have the right to publish my name on its website and keep a record of the students who plagiarized in their thesis work.

_____

Qurrat Ul Ain

Date: _____

Author's Declaration

I, Qurrat Ul Ain, at this moment, state that my Master's thesis titled***: A Comprehensive Experimental Evaluation of Anomaly Detection Techniques in Aviation Cyber-Physical Systems*** is my work, and I have not previously submitted it for taking partial or full credit for the award of any degree at this University or anywhere else in the world. If my statement is incorrect, at any time, even after my graduation, the University has the right to revoke my master's degree.

_____

Qurrat Ul Ain

Date: _____

Certificate of Approval

*It is certified that the research work presented in this thesis, entitled "**A Comprehensive Experimental Evaluation of Anomaly Detection Techniques in Aviation Cyber-Physical Systems** was conducted by Qurrat Ul Ain under the supervision of Dr. Atif Aftab Ahmed Jilani.*

No part of this thesis has been submitted anywhere else for any other degree.

This thesis is submitted to the Department of Computer Science in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering

at the

National University of Computer and Emerging Sciences, Islamabad, Pakistan

June' 2024

Candidate Name: Qurrat Ul Ain Signature: _____

Examination Committee:

1. Name: Dr. Khubaib Amjad Alam Signature: _____
   Associate Professor, FAST-NU Islamabad.
2. Name: Mr. Bilal Khalid Dar Signature: _____
   Lecturer, FAST-NU Islamabad

Dr. Hammad Majeed _____
Graduate Program Coordinator, National University of Computer and Emerging Sciences, Islamabad, Pakistan.

Dr. Usman Habib _____
Head of the Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, Pakistan.

Abstract

Anomaly detection in Cyber-Physical Systems (CPS) is paramount for maintaining safety and operational efficiency across various sectors, particularly in aviation, where the stakes are high. This study focuses on identifying anomalies in aviation systems, comparing diverse machine learning (ML) techniques and deep learning techniques, such as), Naïve Bayes, Active Learning, and Neural Networks (NN), to ascertain their effectiveness in anomaly detection. Our comprehensive experiments revealed distinct performance characteristics across different algorithms. The results indicated that RESAD achieved the highest accuracy in the supervised category, while Deep SAD outperformed other models in the semi-supervised category. In the unsupervised category, LSTM-AE showed superior anomaly detection capabilities compared to other methods. The study underscores the importance of selecting appropriate ML and DL techniques tailored to specific datasets and scenarios within the aviation sector. This research provides critical insights into the strengths and limitations of various ML and DL approaches, enhancing the resilience and reliability of aviation systems. This work contributes significantly to the field by offering a detailed comparative analysis and establishing a foundation for future research in aviation anomaly detection. Future work will involve developing a benchmarking framework for more extensive evaluations, enabling continuous improvement and adaptation of ML and DL techniques to the evolving challenges in aviation safety.

Acknowledgments

First and foremost, I would like to extend my heartfelt gratitude to Allah Almighty for granting me strength, perseverance, and guidance throughout this thesis. His blessings and divine support have been my constant inspiration and motivation. I want to express my deepest gratitude to my supervisor, Dr. Atif Aftab Ahmed Jilani, for his valuable guidance, support, and expertise throughout the entire thesis journey.

I sincerely thank my thesis committee members, MSRC, for their valuable input, constructive criticism, and scholarly advice. Their collective wisdom and expertise have significantly enriched the quality of this work.

I am indebted to my family and friends for their unwavering support and encouragement throughout this demanding process. Their love, patience, and understanding have been a constant source of motivation.

Finally, I express my heartfelt appreciation to all the individuals of Quest Lab who have aided and provided advice and encouragement, both directly and indirectly, during the completion of this thesis. Their contributions, no matter how small, have significantly impacted this endeavor.

Dedication

This thesis is dedicated to my father (Muhammad Shabbir) and sister (Durre Shahwar) for always being there for me. Thank you for supporting and encouraging me to go beyond my expectations. Incredibly dedicated to my Late Mother(Shama Shabbir). She dreamed that I tried to fulfill.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Cyber-Physical Systems (CPS) are integrated systems consisting of physical and computer components that work together to operate processes efficiently in all environments. CPS is used across various sectors, including agriculture, defense, energy, transportation, healthcare, and aviation. The cost of the anomalies in cyber-physical systems can be significant, ranging from a few thousand dollars to millions of dollars per incident, depending on the severity of the anomaly and the affected system. Studies have shown that anomalies can lead to production downtime, equipment damage, and safety hazards. The state of deviation of system behavior from normal behavior is called an anomaly. Anomalies can lead to system failures that may result in life-threatening consequences. A minor sensor malfunction may have minimal impact, while a complete system shutdown can be costly. Also, the longer it takes to diagnose and resolve the anomaly, the more significant the financial impact. Therefore, anomaly detection is necessary to prevent such result failures. Anomaly detection in CPS involves identifying events that do not match the expected behavior, which is essential for ensuring the safety and security of the system. Cyber-physical systems have a profound impact, specifically on the aviation sector, focusing on a critical aspect of aviation: anomaly detection. There have been various tragic events in aviation where numerous lives were lost due to anomalies, such as the Cali Crash (1992), Ariane 5 failure (1996), and Heathrow Airport Disruption (2009), highlighting the severe consequences of undetected anomalies.

In the aviation sector, cyber-physical aviation involves physical processes and digital information integration in the air transportation system, leading to constant communication and interaction among manufacturers, operators, developers, and users throughout the entire lifecycle. Anomaly detection in aviation, within the context of CPS, plays a pivotal role in safeguarding the industry, given its unique safety and operational requirements. Unidentified anomalies in aviation can pose severe threats, ranging from safety hazards to operational disruptions, with far-reaching economic implications. Consequently, developing and implementing effective anomaly detection mechanisms are imperative in this domain.

Like other CPS domains, the aviation industry has witnessed the application of machine learning techniques to enhance the accuracy and efficiency of anomaly detection. Different algorithms, including decision trees(DT), support vector machines(SVM), and neural networks(NN), have been employed to detect anomalies in aviation operational processes. Recognizing the limitations of traditional anomaly detection methods.[1]The aviation industry is distinctive in its features, encompassing aircraft, airports, and air traffic control systems, necessitating a specialized approach to detecting anomalies. An effective anomaly detection system minimizes the likelihood of potential accidents, reduces maintenance costs, and boosts operational efficiency. The objective is to comprehensively assess the types of anomalies that current detection methods address and identify any remaining challenges in the aviation anomaly detection landscape.

A methodology assesses the effectiveness of machine learning and deep learning-based anomaly detection techniques in the aviation industry. The assessment aims to identify the best machine learning and deep learning algorithms for detecting aviation anomalies and highlight the associated challenges and limitations. The research also seeks to find the most suitable anomaly detection mechanisms for aviation systems. This study aims to enhance the security and reliability of aviation cyber-physical systems. This is particularly important because more anomaly detection strategies are needed in the aviation industry. The study aims to evaluate existing methodologies comprehensively and provide insights into their strengths and weaknesses. The experimental evaluation within this study is a crucial step toward strengthening the resilience of aviation systems. The findings of this study will significantly benefit the field of aviation anomaly detection.

Motivation

The motivation for doing experimental evaluation is to validate the efficacy and practical applicability of machine learning and Deep learning approaches, especially in the aviation industry. The experimental assessment can help determine the most effective anomaly detection mechanisms for specifically Flight data records(FDR). Additionally, having different datasets can help identify the scalability matrix and the effective and efficient ML approach to find anomalies.

Problem Statement

Anomaly detection in aviation is a critical research area that ensures operational efficiency and safety. However, the effectiveness of current detection methods and the selection of suitable machine learning and deep learning algorithms for anomaly detection in aviation systems still need to be improved. The goal is to evaluate and compare various machine learning and deep learning techniques for anomaly detection in aviation systems, identify the most effective approaches, and provide insights.

Major Contributions

The significant contributions of our work include:

- Identification of machine learning and deep learning approaches capable of working across different datasets within the aviation industry.
- Empirically experimental evaluation methodology to validate both approaches practical applicability and efficacy for anomaly detection techniques in aviation.

This thesis is structured into several chapters, each addressing specific aspects of the research. Chapter 2 provides background information and an understanding of the topics used in the thesis. Chapter 3 outlines the literature review on anomaly detection in CPS and aviation using machine learning and deep learning. It explores existing studies relevant to the research topic, providing a comprehensive understanding of this domain's current state of knowledge. Moving forward, Chapter 4 describes the research methodology and overall approach to the experiment, including the data collection, selection of methods, and any protocol followed. The Results are discussed in Chapter 5 and concluded in Chapter 6 with plans.

Chapter 2

Background

This chapter discusses cyber-physical systems (CPS) concerning the aviation sector, how anomalies can be hazardous for a system, different types of anomalies, and how they can be detected by anomaly detection. Our research aims at anomaly detection in aviation and anomaly detection using machine learning and deep learning.

Cyber-Physical System

Cyber-physical systems are an integral part of our lives. These systems combine sensing, networking, and computational control into our physical infrastructure, connecting everything to the internet and each other. These systems are used in various fields, such as robotics, healthcare, and aviation. One of the most impressive features of cyber-physical systems is their ability to automate complex processes, making tasks more accessible and efficient. They also provide real-time monitoring of physical systems, which is critical in scenarios like autonomous vehicles, which can be life or death. In aviation, cyber-physical systems using sensor data and automation systems improve flight control, reduce pilot workload, and enhance flight safety. Cyber-physical systems are crucial in addressing coexisting challenges and taking advantage of opportunities to improve air travel efficiency, safety, and performance.

Anomalies Detection:

Anomaly refers to any deviation from the expected behavior of software that may lead to system failures [3]. Such anomalies can range from minor to considerable damage to the system and its users [4], resulting in loss of productivity, consumer goodwill, and revenue. There have been various tragic events in aviation where numerous lives were lost due to anomalies, such as the Cali Crash (1992), the Ariane 5 failure (1996), and the Heathrow Airport Disruption (2009) [5], [6]. These examples demonstrate the potential

consequences of system anomalies, which can negatively impact public safety. Anomaly is defined abstractly as an unusual pattern from the anticipated behavior. The most common definition of anomalies is the occurrence of unexpected patterns or behaviors in software.

"Anomalies are patterns in data that do not conform to a well-defined notion of normal behavior." [2]

1.1.1 Types of Anomalies:

The Classification of anomalies is divided into three major categories that commonly exist. [2][3], [4]

1. **Point anomalies:** A single data point significantly different from the rest of the data points in the dataset is considered a point anomaly and the simplest form of anomaly. For instance, in a time series of Antarctica temperatures in months, a temperature of 40℃ or 50℃ can be considered an anomaly. In Figure 1, the point in red represents a point anomaly compared to the average points presented in *blue.*
2. **Collective anomalies:** A group of anomaly points, which might not be an anomaly individually, but their appearance together is an anomaly. It is associated with the anomaly of the entire dataset. These anomalies can only be detected where data is related in datasets, i.e., spatial, graph, or sequential.
3. **Contextual anomalies:** If the data is anomalous in a particular context but not in another context, it is known as contextual anomalies [5]. For instance, the usual summer temperature is exceptionally average for summer but would not be considered normal in winter.

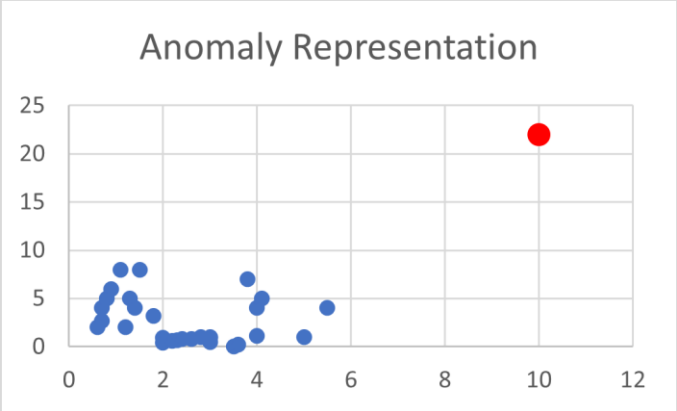


Figure 1 Point Anomaly

After defining anomaly, we will define anomaly detection and the methods for detecting abnormalities.

In literature, different terms are used to refer to anomaly detection, such as *novelty detection, deviant discovery, (rare) event detection, change point detection, misuse detection, fault detection, or intrusion detection* [9], [10]. All these terms have the same objective - to identify rare data points that significantly differ from the distribution of the dataset generally. It should be noted that anomalies differ from novel data, as they are typically considered average data before being detected [11]–[13].

Choosing the appropriate method for effective anomaly detection depends on the data's properties. The following properties must be considered when selecting an appropriate approach:

1. Non-Temporal vs Temporal data: Temporal data includes data with unequal timestamps and time series intervals. Non-temporal data can be protein sequences, medical images, etc.

2. Univariate vs. Multivariate data: Instances of multivariate data are observed time-series data by several sensors and contain more than one time series, whereas univariate data has only one dimension.

3. Labeled and unlabeled data: If a label exists for each element, then the data is labeled and determines whether the data is anomalous or normal. Labeled data is the subject

of supervised anomaly detection approaches. Unsupervised anomaly detection approaches will be used for the unlabeled data, as it does not have any label for the element.

4. Types of anomalies in the dataset: different anomaly types are defined in section 2.1. The method selection is affected by this information as rare classification is used for point anomalies and unusual shapes to detect collective anomalies. In contrast, searching for deviation-finding aid is used for contextual anomalies.

This thesis focused on labeled and unlabeled data with anomalous points. Therefore, we will empirically evaluate different aviation datasets using machine learning and deep approaches.

Anomalies Detection in Aviation:

Anomaly detection in aviation refers to identifying unexpected or unusual events or patterns that deviate from expected behavior in-flight data. This is a critical research area that aims to ensure operational efficiency and aviation safety. Aviation involves using various data-driven technologies and methods to analyze flight data, such as the flight data recorder (FDR) and flight data monitoring (FDM), to identify anomalies that may affect safety or operational issues.

Various techniques are employed to detect anomalies in aviation, including semi-supervised deep learning models, data-driven anomaly detection methods, stacked recurrent autoencoder methods with dynamic thresholding, and machine learning. These techniques compare the parameters of flights with predefined thresholds using different algorithms and apply other regression and classification techniques.

Anomaly detection in aviation is essential for improving safety and developing the next frontier of uncrewed aircraft. For this thesis, we will use machine learning and deep learning classification techniques to provide a detailed analysis of the approaches already used on different datasets. Additionally, we will explain further machine learning and deep learning approaches that will be part of this thesis. Figure 2 shows the authors who have majorly contributed to the domain of anomaly detection for aviation.

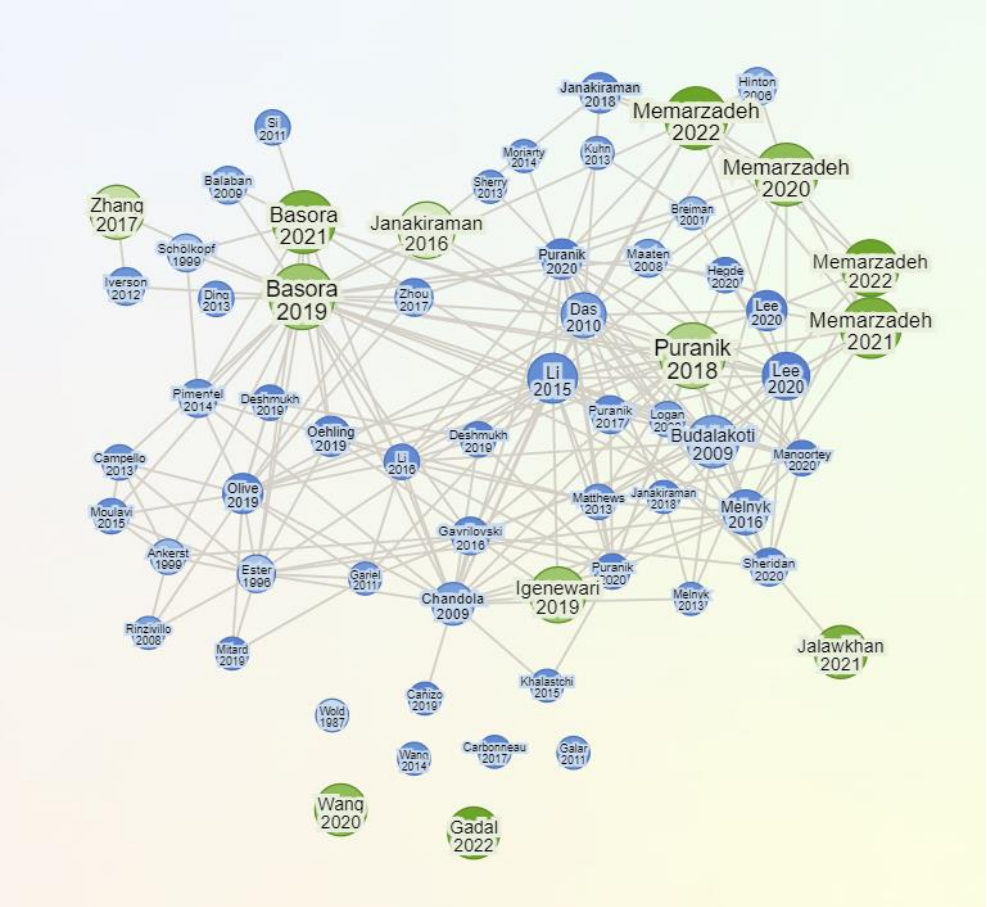


Figure 2 Author's contribution

Deep Learning Approaches:

Deep Learning is a powerful technique for building complex models that can learn from large amounts of data. In anomaly detection, deep learning can identify unusual patterns or outliers in data that deviate from normal behavior. The main idea behind deep learning for anomaly detection is to learn a model of standard data and then use this model to identify anomalies. This can be done using various techniques, such as autoencoders, generative adversarial networks (GANs), clustering techniques, and convolutional neural networks (CNNs).In deep learning, anomalies are detected by analyzing the patterns of logged events.

Approaches of deep learning used in the thesis are DBSCAN and Multiclass. Each of these approaches offers unique capabilities for detecting anomalies within diverse datasets.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise):

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a robust clustering algorithm well-suited for identifying clusters of varying shapes and sizes in data, including noise and outliers. Unlike traditional clustering methods, DBSCAN does not require the number of clusters to be specified in advance. Instead, it relies on two key parameters: EPS (the radius of a neighborhood around a point) and mints (the minimum number of points required to form a dense region). The algorithm classifies points as core points, border points, or noise. A core point has at least minPts points within its EPS radius, forming the basis of a cluster. Border points are within the EPS radius of a core point but do not meet the core point criteria. Points that are neither core nor border points are classified as noise. DBSCAN begins by selecting an unvisited point, checking its neighbors, and expanding clusters by recursively visiting and connecting neighboring core points. This approach effectively allows DBSCAN to discover clusters in large spatial databases and is particularly advantageous in detecting anomalies or noise, as these are naturally excluded from the clusters. Its ability to identify clusters of arbitrary shape and handle noise makes DBSCAN a versatile and widely used clustering method.

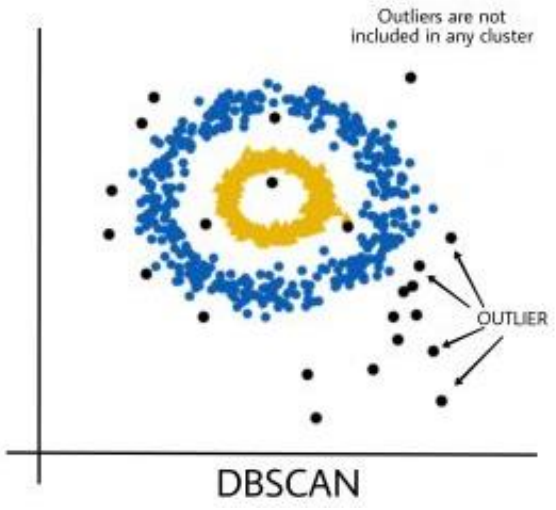


Figure 3 DBSCAN Representation

Multiclass Anomaly Detection:

Multiclass anomaly detection extends the traditional concept of anomaly detection by identifying unusual patterns or deviations within multiple predefined classes or

categories rather than treating the data as belonging to a single class. This approach is particularly beneficial in scenarios where data naturally falls into different categories, such as fraud detection across various transaction types, medical diagnostics across different diseases, or quality control in manufacturing with varying product types. In multiclass anomaly detection, the objective is to detect anomalies within each class and across the entire dataset. This typically involves two key steps: class-wise analysis and combined analysis.

In class-wise analysis, the model examines each class independently, learning the normal behavior specific to that class. This helps detect intra-class anomalies, where an instance deviates significantly from the norm within its class. Techniques such as supervised or semi-supervised learning can be employed, leveraging labeled data within each class to train the model.

The combined analysis then integrates the insights from individual classes to identify inter-class anomalies, where an instance might be misclassified or exhibit inconsistent behavior across multiple courses. This requires a holistic view of the data, often utilizing clustering algorithms, ensemble methods, or advanced deep-learning techniques to capture the complex relationships and dependencies between classes.

Multiclass anomaly detection assigns anomaly scores based on deviations within and between classes, providing a nuanced and comprehensive understanding of anomalies in the dataset. This dual approach detects subtle and context-specific anomalies, enhancing the model's accuracy and robustness. By addressing each class's unique characteristics and challenges while maintaining a global perspective, multiclass anomaly detection is highly effective in diverse applications, from detecting various cyber threats to ensuring product quality across different manufacturing lines.

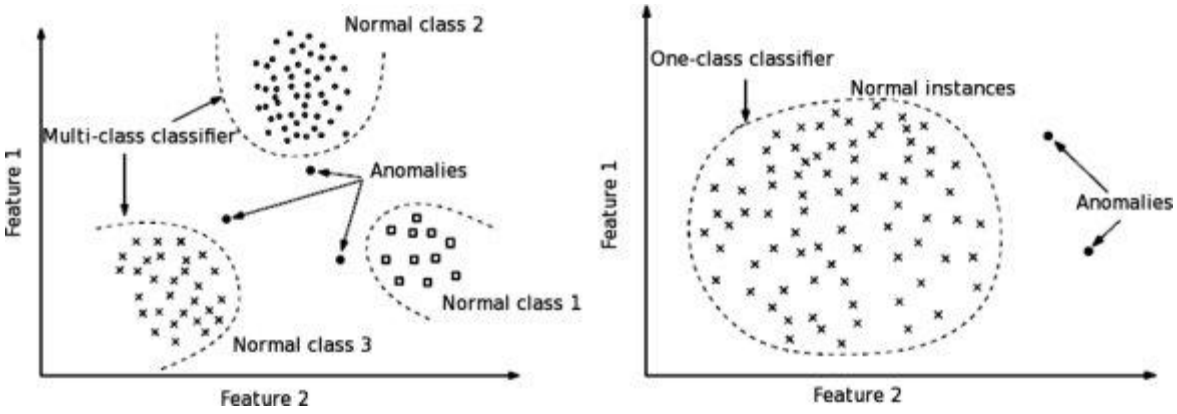


Figure 4 Multiclass Representation

Machine Learning Approaches:

Machine learning (ML) is becoming increasingly popular for detecting anomalies. ML is the process of automating knowledge acquisition from examples and is used to create models that can distinguish between abnormal and regular classes. Three main categories of machine learning are used for anomaly detection, and they are defined by the function used to train the model. These categories are:

1. Supervised Machine Learning: This approach involves training datasets containing labeled instances for anomalous and standard data. The challenge is that labeled anomalous data can be scarce in specific industries, such as aviation.

2. Semi-Supervised Learning: This approach uses a small amount of labeled data and a combination of labeled and unlabeled data to train the model. Most of the data used is typically unlabeled.

3. Unsupervised Learning: This type of machine learning does not require labeling and can learn associations and distributions to classify input without labels. However, supervised models are typically more straightforward to use for tasks such as anomaly detection.

Delving further into anomaly detection techniques used in this thesis. These techniques include LOF (Local Outlier Factor), LSTM (Long Short-Term Memory) for unsupervised machine learning, LSTM-AE (Long Short-Term Memory Autoencoder),

and Naïve Bayes for supervised machine learning. Each of these approaches offers unique capabilities for detecting anomalies within diverse datasets.

LOF:

Local Outlier Factor (LOF) is also an anomaly detection method that identifies outliers in a dataset by computing the local density deviation of a given data point concerning its neighbors. It considers a data point an outlier if it has a substantially lower density than its neighbors. The work of LOF can be summarized in the following steps.

- Calculate distances: For each data point, calculate the distances to its k-nearest neighbors.

- Determine reachability distance: For each data point, calculate the reachability distance, which is the maximum distance to its k-nearest neighbors.

- Calculate local reachability density (LRD): For each data point, calculate the LRD, which is the number of objects in the k-nearest neighbors divided by the reachability distance.

- Calculate LOF: For each data point, calculate the LOF by comparing its LRD to the LRD of its k-nearest neighbors. If the LOF exceeds 1, the data point is considered an outlier.

LOF is particularly useful for detecting outliers in datasets whose density is not constant. It can identify outliers in datasets where the local density deviates from the global density. However, it is essential to note that identifying an outlier depends on the problem and the user, as there is no specific threshold value above which a point is defined as an outlier.

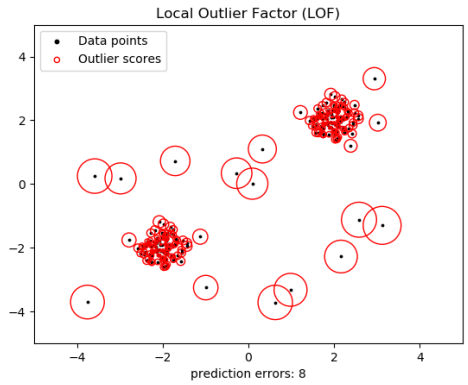


Figure 5 LOF Representation

LSTM:

Long Short-Term Memory (LSTM) is a well-suited and robust recurrent neural network (RNN) type for handling sequential data with long-term dependencies. LSTMs can learn long-term dependencies in sequential data, making them particularly effective for tasks involving time series and sequences, such as machine translation, speech recognition, and text summarization. The working of an LSTM can be summarized in the following steps.

- Memory Cell: LSTMs address the problem of learning long-term dependencies by introducing a memory cell, a container that can hold information for an extended period. This allows LSTMs to retain selected information in long-term memory, stored in the so-called Cell State, and short-term information in the hidden state, enabling them to capture long-term dependencies crucial for solving intricate tasks.
- Gates: LSTMs use a series of gates (input, output, and forget) to control how information is stored and retrieved in the memory cell. These gates enable LSTMs to selectively retain or discard information as needed, allowing them to avoid the vanishing gradient problem faced by traditional RNNs and process entire sequences of data without treating each point independently

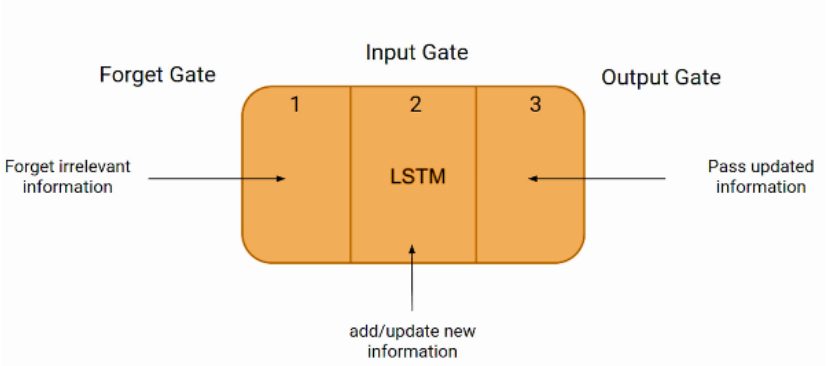


Figure 6 LSTM Representation

LSTM AE:

Long Short-Term Memory Autoencoder (LSTM-AE) is a type of autoencoder that uses LSTM-based architecture for autoencoder neural networks for unsupervised learning of data representations. The LSTM-AE is specifically designed to process, and Sequential data can be reconstructed, making it ideal for detecting anomalies in time series data. The working of an LSTM-AE can be explained as follows:

- Encoder Phase: In the encoder phase, the input sequence is fed into the LSTM network, which learns to encode the sequential information into a fixed-size internal representation. This internal representation captures the temporal dependencies and patterns present in the input sequence.
- Latent Space: The fixed-size internal representation from the encoder phase is often called the "latent space" or "code." This latent space encapsulates the input sequence's most important features and patterns, effectively compressing the information meaningfully.
- Decoder Phase: In the decoder phase, another LSTM network takes the latent space representation as input and learns to reconstruct the original input sequence. The network aims to generate an output sequence that closely matches the input sequence, effectively learning to decode the information from the latent space representation.
- Anomaly Detection: The LSTM-AE learns to reconstruct standard patterns in the input data during training. Subsequently, when it is presented with new data during the testing phase, the model's ability to accurately reconstruct the input

sequence is used to identify anomalies. Data points that cannot be well-reconstructed due to deviation from standard patterns are flagged as potential anomalies.

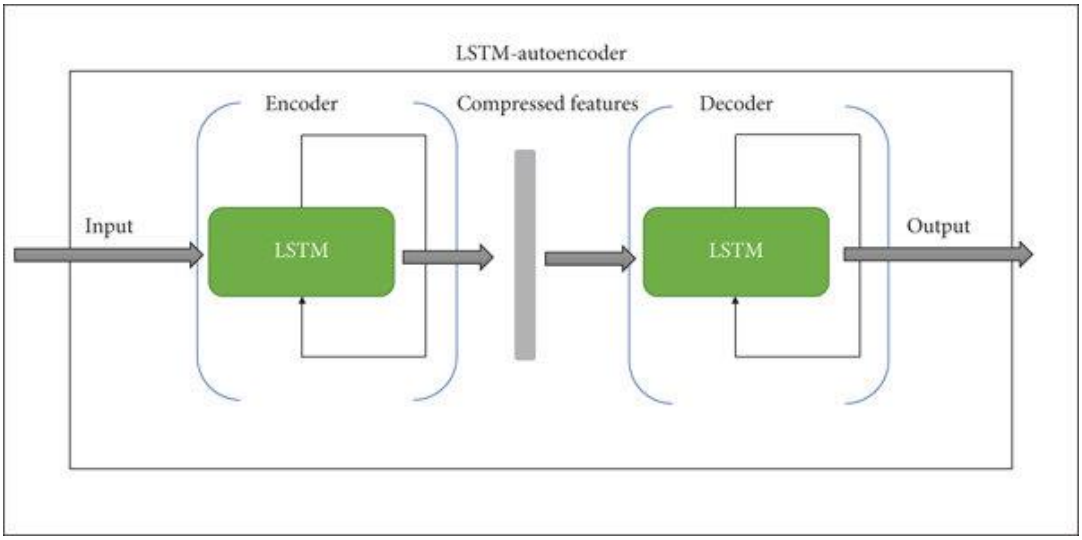


Figure 7 LSTM-AE Representation

Naïve Bayes:

Naive Bayes is a probabilistic classifier that is simple and effective. It is based on Bayes' Theorem and is commonly used for text classification and spam filtering tasks. The algorithm assumes that the features are conditionally independent, meaning that one feature's presence does not affect another's presence. However, this assumption needs to be more simplified., Naive Bayes classifiers are widely utilized for their simplicity and efficiency in machine learning. The algorithm is part of a family of generative learning algorithms and is known for its fast and reliable performance, particularly in natural language processing (NLP) problems. It is based on fundamental probability knowledge and makes predictions by calculating the conditional probability of a class given the features. Despite its simplicity, Naive Bayes often performs surprisingly well in practice, especially in scenarios where the independence assumption holds approximately the same.

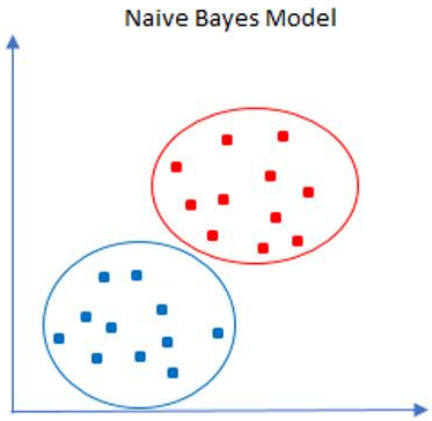


Figure 8 Naive Bayes Representation

Extreme Learning Machine:

An Extreme Learning Machine (ELM) is a feedforward neural network with a single layer of hidden nodes, where the weights between the input and hidden layer are randomly assigned and fixed. Huang et al. introduced ELM as a fast and efficient learning algorithm for single-layer feedforward neural networks (SLFNs). The critical characteristics of ELM include:

1. **Randomized Weights**: The input weights and biases are randomly assigned and not tuned during training.
2. **Analytical Solution**: The output weights are computed analytically using a simple linear system, often via the Moore-Penrose generalized inverse, making the training process extremely fast.
3. **Universal Approximation**: Despite the simplicity, ELM has been proven to have universal approximation capability, meaning it can approximate any continuous function given enough hidden nodes.

For anomaly detection, ELM can identify patterns that deviate significantly from the expected behavior in the data. Here's a step-by-step outline of how ELM can be applied to anomaly detection:

1. **Data Preparation**: Preprocess the data, including normalization and feature extraction, to make it suitable for the ELM model.

2. **Randomized Initialization**: Randomly initialize the weights and biases between the input and hidden layers. This step is crucial as it defines the input data's transformation.
3. **Hidden Layer Activation**: Compute the activation of the hidden layer. This is done by applying a non-linear activation function (e.g., sigmoid, ReLU) to the weighted sum of the inputs and biases.
4. **Output Weights Calculation**: Solve for the output weights analytically using the Moore-Penrose pseudoinverse of the hidden layer activations. The output weights link the hidden layer activations to the final output.
5. **Model Training**: Train the ELM model on the average (non-anomalous) data. The goal is to learn the representation of normal behavior.
6. **Anomaly Detection**: Apply the trained ELM model to new data points. Calculate the reconstruction error or use the model's output to determine if the data point is anomalous. High reconstruction error or significant deviation from the expected output indicates an anomaly.

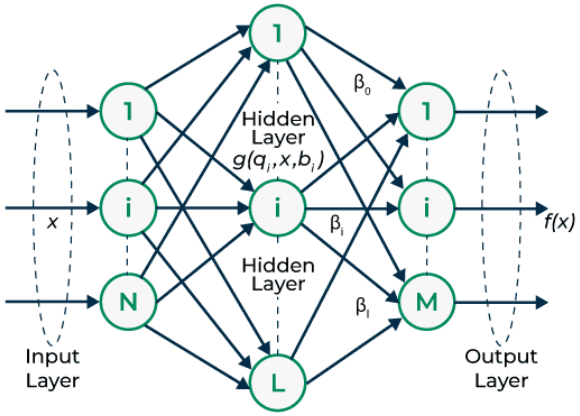


Figure 9 ELM Representation

Active Learning:

Active learning is a type of machine learning where the algorithm selectively queries the most informative or uncertain data points for labeling to improve model performance. It is beneficial when labeled data is relatively inexpensive for working on anomaly detection. Start by training a model on a small set of labeled data and following the steps below.

- **Query Strategy:** Use a query strategy to select the most informative or uncertain data points. Common strategies include:
  - **Uncertainty Sampling**: Selecting data points where the model is least confident.
  - **Query-by-Committee**: Using an ensemble of models to select data points where disagreement exists.
  - **Expected Model Change**: Choosing data points expected to cause the most significant change in the model.
- **Labeling**: An oracle (e.g., a human expert) labels the selected data points.
- **Model Update**: Retrain the model with the newly labeled data.
- **Iteration**: Repeat the process to improve the model's performance with minimal labeled data continuously.

Spatio-Temporal Anomaly Detection:

Spatio-temporal anomaly detection involves identifying anomalies in data with spatial (geographical) and temporal (time-based) components. This is useful in traffic monitoring, climate analysis, and sensor networks.

- **Spatial Analysis**: Analyze data across different spatial locations to identify unusual patterns or deviations.
- **Temporal Analysis**: Analyze data over time to detect temporal anomalies or trends.
- **Combination**: Combine spatial and temporal features to detect complex anomalies involving space and time changes.

- **Modeling Techniques**: Techniques like spatial-temporal autoregressive models, convolutional neural networks (CNNs), and recurrent neural networks (RNNs) are often used.

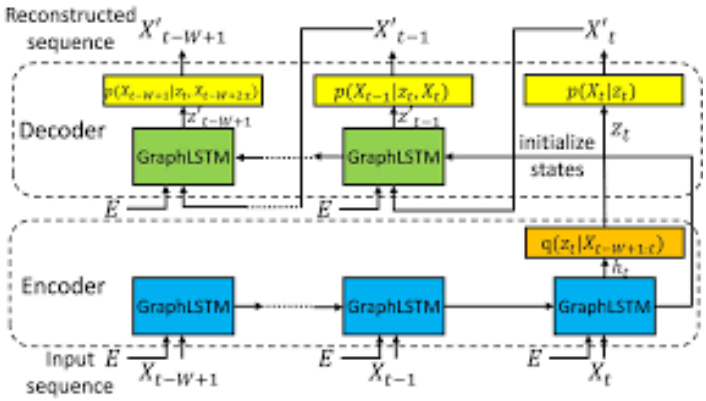


Figure 10 Active Learning Representation

GANomaly:

GANomaly is a model that uses Generative Adversarial Networks (GANs) for anomaly detection, leveraging the generative power of GANs to learn the distribution of standard data and identify anomalies based on reconstruction errors. The architecture consists of two main components: a generator and a discriminator. The generator in GANomaly has a unique encoder-decoder-encoder structure. The first encoder compresses the input data into a latent space, the decoder reconstructs the data from this latent representation, and the second encoder maps the reconstructed data back to the latent space. This structure ensures that the generator can produce realistic outputs matching the normal data distribution.

During training, the generator is trained to minimize the reconstruction error between the input and reconstructed data. This involves passing the average data through the first encoder to obtain a latent representation, reconstructing the data using the decoder, and then passing it through the second encoder to ensure it maps back to the same latent space. The discriminator is simultaneously trained to differentiate between actual and reconstructed data, helping the generator produce more realistic reconstructions.

After training, the model evaluates new data points for anomaly detection by reconstructing them through the generator. The reconstruction error is then calculated using metrics like mean squared error. Suppose the reconstruction error for a data point is high. In that case, the point significantly differs from the standard data the generator was trained on, flagging it as an anomaly. This method effectively detects anomalies because the generator struggles to accurately reconstruct data points that deviate from the normal data distribution, leading to higher reconstruction errors for anomalies.

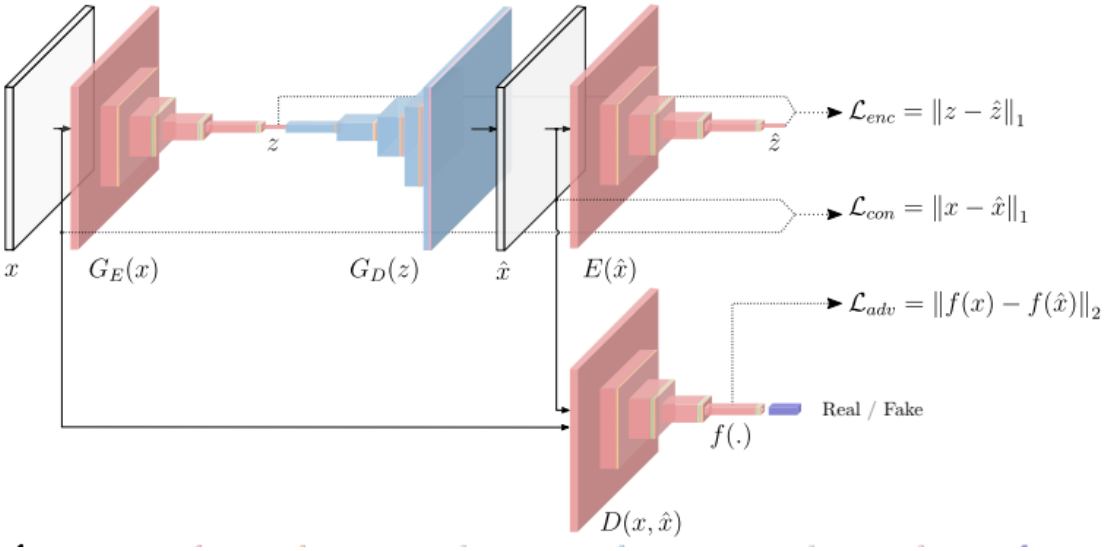


Figure 11GANomaly Representation

Deep SAD (Deep Semi-Supervised Anomaly Detection):

Deep SAD (Deep Semi-Supervised Anomaly Detection) is a sophisticated approach to effectively leveraging labeled and unlabeled data to identify anomalies in complex datasets. Unlike traditional anomaly detection methods that rely solely on labeled data or completely unsupervised techniques, Deep SAD combines the strengths of supervised and unsupervised learning. The model is based on deep neural networks, capable of learning intricate patterns and representations from data. The process begins by training the network on a dataset that contains both regular and a small number of labeled anomalous instances. The core idea is to map average data points to a compact cluster in the latent space while pushing anomalous data points away from this cluster. This is achieved by minimizing a loss function that penalizes the distance between average points and a predefined center in the latent space while maximizing the distance for anomalous points. An existing unsupervised technique named Deep Support vector data description (SVDD) is used as the motivation for Deep SAD. This technique aims to train the neural network (*phi*) to learn a transformation that minimizes the volume of a data-enclosing hypersphere centered on a pre-determined point *c*. It penalizes the mean squared forces the network to extract common factors of data variation, which are the most stable in the dataset. Consequently, average data points get mapped near the hypersphere center, while anomalies are mapped further out.

$$\min_{\mathcal{W}} \quad \frac{1}{n}\sum_{i=1}^{n}\|\phi(\boldsymbol{x}_i;\mathcal{W})-\boldsymbol{c}\|^2+\frac{\lambda}{2}\sum_{\ell=1}^{L}\|\boldsymbol{W}^{\ell}\|_F^2, \quad \lambda>0$$

During training, the model learns to distinguish between normal and anomalous data by adjusting the network parameters to minimize reconstruction errors for standard data and maximize errors for anomalies. This semi-supervised approach allows Deep SAD to effectively utilize the available labeled anomalies to guide the learning process, enhancing its ability to detect anomalies even when they are rare. Once trained, the model can evaluate new data points by projecting them into the latent space and measuring their distance from the center. Data points that lie far from the center are flagged as anomalies. The effectiveness of Deep SAD lies in its ability to learn robust representations and its flexibility to incorporate both labeled and unlabeled data, making it highly suitable for real-world applications where labeled anomalies are scarce but critical for accurate anomaly detection.

RESAD (Residual Anomaly Detection):

Residual Anomaly Detection (RESAD) leverages the architecture of residual networks (ResNets) to identify anomalies in complex datasets. ResNets are a type of deep neural network that incorporates skip connections, allowing the model to effectively train deeper networks by bypassing one or more layers and mitigating issues like vanishing gradients. In the context of RESAD, the model is trained on standard data to learn the underlying patterns and representations. The model can capture intricate data structures and dependencies using ResNet's skip connections and deep layers. Anomalies are detected by evaluating deviations from the learned standard patterns. During inference, if a data point significantly deviates from the expected output generated by the ResNet, it is flagged as an anomaly. This approach is efficient because the residual connections help preserve information across layers, enabling the network to detect subtle anomalies that shallower models might miss. RESAD's ability to handle complex and high-dimensional data makes it a robust choice for anomaly detection in various applications, including industrial monitoring, financial fraud detection, and cyber-physical systems.

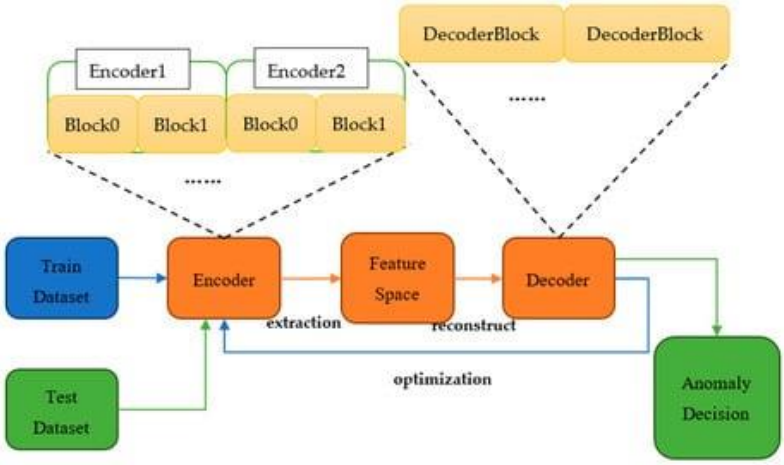


Figure 12 READ Representation

Summary

This chapter explores the integral role of cyber-physical systems in various fields, particularly aviation, where they enhance flight control and safety through real-time monitoring and automation. It delves into anomaly detection, defining anomalies as deviations from expected behavior that can lead to system failures, with historical aviation incidents highlighting their potential consequences. The chapter categorizes anomalies into point, collective, and contextual types and emphasizes the importance of selecting appropriate detection methods based on data properties. It also reviews machine learning and deep learning techniques for effective anomaly detection, particularly in the aviation sector, to ensure operational efficiency and safety.

Chapter 3

Literature Review

Anomaly detection has been actively researched, with numerous applications and methods. This Literature review explores state-of-the-art anomaly detection from the distinct themes: (1) Anomaly Detection in Cyber-Physical Systems, (2) Anomaly Detection using Machine Learning, (3) Anomaly Detection in Aviation Using Deep Learning, (4) Anomaly Detection in Aviation Using Machine Learning. Each theme contributes unique insights to lead future directions.

Anomaly Detection in Cyber-Physical Systems

Anomaly detection using AI/ML in cyber-physical systems has become crucial for ensuring reliable and secure operation. Various papers propose supervised machine learning-based anomaly detection models (ML-ADS) for detecting cyber/physical attacks in different domains. This paper explores the utilization of Explainable Artificial Intelligence (XAI) and Machine Learning (ML) techniques for anomaly detection[23]. This paper examines the use of Explainable AI and Machine Learning techniques for anomaly detection in different domains. The proposed model integrates feature engineering selection and outlier detection, enhancing the comprehension of complex CPS phenomena. Comparative analysis against established algorithms like ABOD and CBLOF highlights the superiority of the proposed approach. Additionally, the study identifies Random Forest (RF) as a robust ML algorithm for anomaly detection, surpassing SVM, NB, and MLP[6]. This research underscores the critical need for advanced AI methodologies to secure and manage the escalating intricacies of CPS environments[7].

It is uncovering prevailing methods, tools, underlying causes, and metrics concerning uncertainty. Employing a systematic approach, the study utilizes keyword-based searches across publisher sites to identify relevant research. The review categorizes tools for uncertainty mitigation and identifies[8] gaps in state-of-the-art methods that need comprehensive uncertainty measurement metrics. These findings consolidate current knowledge and offer guidance for future research, aiming to devise advanced approaches and tools to manage uncertainty effectively in CPS[27]. Aquila utilizes an optimizer to enhance the machine learning-based anomaly detection (OMLABD) approach in the context of cyber-physical systems (CPS). The AOPTML-AD framework employs the Aquila optimization algorithm-based feature selection (IAOA-FS) approach for optimal feature subset selection, followed by the utilization of the Chimp optimization algorithm (ChOA) alongside an adaptive neuro-fuzzy inference system (ANFIS) for anomaly detection[9]. This approach is substantiated through benchmark data validation, displaying superior performance compared to recent models, achieving an accuracy of 9.37%. Particularly noteworthy is the development of the fitness function for the IAOA-FS technique, which balances classification accuracy and the number of selected features within each solution. The AOPTML-AD approach excels in accuracy, precision, recall, F-score, and MCC values, underscoring its efficacy in anomaly detection within the CPS environment [10].

Comprehensive assessment of anomaly detection techniques within Cyber-Physical Systems (CPS) to combat security threats, with a specific focus on life safety concerns in industrial control networks (ICS)[11]. It navigates through existing research, acknowledging challenges like resource constraints and communication protocol standardization. Analyzing 296 papers, the study highlights thematic trends and research voids while statistically analyzing covered areas. The review underscores the vital role of industry-academia collaboration in fortifying CPS security and reliability, particularly in the interplay between industrial control networks, corporate networks, and the public internet [12].

As the exploration into anomaly detection within cyber-physical systems concludes, the following theme takes us into a broader perspective, investigating the utilization of machine learning techniques for anomaly detection across diverse cyber-physical systems domains and building on the foundational insights gained in the context of CPS.

Anomaly Detection Using Deep Learning

Anomaly Detection is a crucial task in aviation to ensure the safety and security of flights. With the increasing complexity of aircraft systems and the growing volume of data, more than traditional anomaly detection methods are required; deep learning-based anomaly detection (DLAD) methods have been proposed to address these challenges. A taxonomy for DLAD methods in aviation has been proposed, categorizing them based on the type of anomalies, strategies, implementation, and evaluation metrics. Several deep learning models have been applied to anomaly detection in aviation, including autoencoders, generative adversarial networks (GANs), and recurrent neural networks (RNNs). These models have been used to detect anomalies in various aviation data sets, such as flight data, sensor data, and maintenance records.

Explainability and interpretability are crucial aspects of DLAD in aviation. Researchers have proposed methods to explain the decisions made by DLAD models, such as feature importance and saliency maps. These methods help domain experts understand the reasoning behind the detected anomalies and improve the trustworthiness of DLAD systems. Despite the progress in DLAD, several challenges and limitations remain. These include the need for extensive and diverse data sets, the risk of overfitting, and the difficulty in handling imbalanced data. Additionally, DLAD models may need to generalize better to new and unseen data, which can lead to false negatives and false positives.

Additionally, transfer learning and domain adaptation can improve the performance of DLAD models across different aviation domains. In conclusion, DLAD has shown significant promise in detecting anomalies in aviation data. However, more research is needed to address the challenges and limitations of DLAD and to develop more robust and explainable models. Integrating DLAD with other anomaly detection methods and applying DLAD to new aviation domains can further improve the performance and effectiveness of DLAD systems. Furthermore, there are multiple approaches for anomaly detection, but there needs to be a gap in knowing which approach is scalable to different datasets. Therefore, there is a need for experimental evaluation to determine which approach is practical for various datasets, ensuring that DLAD systems can adapt to diverse aviation data.

Anomaly Detection Using Machine Learning

Machine Learning for anomaly detection is a versatile technique that can be applied to various domains, including industrial monitoring, healthcare, and aviation. Machine learning methods such as supervised and unsupervised learning can be effective, especially when dealing with large datasets with the help of automated anomaly detection. Some of the standard machine learning methods used recently are a comprehensive comparative assessment of machine learning techniques for anomaly detection. This study reveals that supervised learning surpasses unsupervised counterparts, yielding near-perfect accuracy, especially with tree-based algorithms reaching 98% accuracy in anomaly detection. Recognizing the significance of anomaly detection and pinpointing faulty components, the review highlights a need for more focus on localization analysis. Evaluating algorithm performance using the area under the receiver operating characteristic curve (AUC) metric, the study demonstrates similar outcomes to accuracy measures [13].

A systematic review of machine learning models for anomaly detection was conducted in 2021[14]. The review analyzed various machine learning models and provided recommendations for researchers. Some commonly used anomaly detection techniques include LOF, SVM, LSTM, and EC-SVM. However, it is essential to note that suitable models should be selected for different data distributions to avoid inaccuracy. Acknowledging the significance of tailored models, particularly in cyber-physical systems, we now focus on the skies, where anomaly detection takes on heightened importance – the aviation sector. In this sector, anomaly detection assumes paramount significance, safeguarding against potential anomalous behavior and ensuring the seamless operation of cyber-physical systems as we delve into the intricacies of anomaly detection within the aviation domain, where precision and adaptability are not just preferences but imperatives.

Anomaly Detection in Aviation

Anomaly detection in aviation has been active research, with machine learning techniques being increasingly used to identify and extract anomalous components from

the data. This literature review focuses on machine learning methods applied to anomaly detection in the aviation sector, specifically for cyber-physical systems in software engineering.

A data analytics framework for anomaly detection in flight operations was proposed in 2021[15]. General aspects of the anomaly detection problem are discussed in this framework, such as commonly used methods and taxonomy. Another review of the recent advances in anomaly detection methods applied to aviation was published in 2019, providing a structured and comprehensive overview of anomaly detection methods, including types of anomalies. Some notable studies included anomaly detection in aviation as an Improved Kernel principal component analysis (KPCA) algorithm for aviation safety from anomaly detection is being discussed in this study [14].

Fleet-level anomaly detection of aviation safety data: A multiple kernel anomaly detection (MKAD) algorithm is discussed in this paper as it is designed specifically for aviation safety data, which addresses the challenges of time series variables, multimodality, temporality, and heterogeneity [16]. Anomaly detection in aviation, a critical facet of ensuring the safety and reliability of cyber-physical systems, has seen significant advancements through diverse methodologies. Commencing with foundational work, a data-driven method proposed for unsupervised anomaly detection and recovery of UAV flight data sets the stage[17]. Integrating spatiotemporal correlation and LSTM-AE neural networks, this approach emphasizes the necessity for tailored techniques in cyber-physical systems. Transitioning to fault detection in aviation, a study employing unsupervised machine learning methods, including convolutional autoencoders, sheds light on fault clustering and underscores the importance of meticulous data curation[18]. Exploring the broader landscape of urban air mobility, the Resilience Decentralized Anomaly Detection (RODAD) framework employs microservices and machine learning-based anomaly detection to counter spoofing attacks, exemplifying the challenges unique to urban air environments [19].

Moving into deep learning, RESAD, a semi-supervised model, leverages labeled and unlabeled data, benchmarking performance based on classification accuracy and interpretability[20]. Simultaneously, a hybrid K-mean array and sequential minimal optimization algorithm exhibit promising results in an anomaly detection model, highlighting advancements in machine learning applications to aviation[21]. The literature also explores innovative techniques such as a hybrid statistical-local outlier factor algorithm, introducing the concept of local outlier probability and tracking

anomalous behavior in time during flights[22]. Additionally, a multi-task model for outlier detection in-flight data underscores the significance of learning semantic features for improved classification, highlighting the evolving landscape of anomaly detection techniques in aviation [23].

Table 1 Literature Review Table

Ref. No.	Compared Approaches	Metrics	Dataset
[6]	LOF LSTM	FPR, FNR, TPR, TNR	ALFA datasets
[7]	Random Forest (RF),(SVM), (NB), and multi-layer perceptron (MLP)	Accuracy, mean square error, and loss are used to evaluate performance.	Multi-dataset (TranAD),(TSB-UAD)
[8]	Isolation Forest, KNN, GANomaly, OC_SVM, LOF, ABOD, LODA	Precision, Recall, F1 –score, AUC	SWaT
[24]	physical low-cost Edge Intelligent Device (EI`D) in a realistic hardware-in-the-loop (HIL) IEEE 39-bus smart grid DER environment	Latency, FPR, FNR	DER DNP3
[9]	Improved Aquila optimization algorithm-based feature selection IAOA-FS, ChOA, ANFIS	Accuracy	Swat
[11]	PCA, Random Forest, CVT, NB, F-SVM	False Positive, Accuracy	Power System dataset, UNSW-NB15
[13]	Linear Regression, NB, DT	True Positive, False Positive, True Negative	KDD ‘99 DARPA
[15]	NB, LOF, LSTM	Accuracy	NASA Dashlink

			ALFA
[16]	RF, SVM, NB, and MLP	Accuracy Precision Recall	NASA Dashlink
[17]	AE, KPCA, SVM, KNN,LSTM, LSTM-RF, WSTC-LSTM-AE	Performance	Thor flight Data 69
[18]	K-means clustering, CA	Accuracy	Proprietary Data
[19]	LSTM XceptionTime	Efficiency	NASA Dashlink UAM
[20]	LOF	Accuracy, F1 score, Precision, Recall	NASA Dashlink
[21]	SVM SMO Clustering hybrid K-mean	TPR, FPR, Precision, Recall, F-Measure	NLS-KDD
[22]	MKAD LOF	F1 score Precision Recall	NASA Dashlink
[23]	K means LOF	Performance	NASA Dashlink
[25]	ATTAIN (Anomaly detection method) LSTM-CUSUM (Baseline model) MAD-GAN (Baseline model	F1 score (Performance metric) Training time reduction Detection delay time	Secure Water Treatment (SWaT) Water Distribution (WADI) Battle Of The Attack Detection Algorithms (BATADAL) PHM challenge 2015 dataset Gas Pipeline Dataset

Research Gap

The current literature on anomaly detection in the aviation sector has identified a research gap, highlighting a critical need for more understanding of practical machine learning and deep learning techniques. Failure to address these gaps could lead to several significant consequences, hindering the progress and effectiveness of anomaly detection systems in aviation. Here are some of the potential consequences:

1. **Performance and Accuracy**: Without comprehensive studies comparing different machine learning and deep learning algorithms, there is a risk of implementing suboptimal models for anomaly detection in aviation systems. This can reduce performance and accuracy, increasing the likelihood of false positives or negatives. In the aviation sector, where precision and reliability are paramount, suboptimal performance can have severe consequences, including compromised safety and increased operational risks.

2. **Increased Vulnerability to Evolving Threats:** Anomalies in aviation systems are dynamic, and threat landscapes are constantly evolving. It is essential to address the research gaps to ensure the aviation sector is prepared to detect and mitigate emerging threats. This can lead to increased vulnerability to novel attack vectors, making aviation systems susceptible to malicious activities that exploit the gaps in current anomaly detection capabilities.

4. **Resource Wastage and Implementation Challenges:** Resource wastage is risky without a comprehensive understanding of the most suitable machine learning algorithms and their limitations. Organizations may invest in technologies that do not align with the unique challenges of anomaly detection in aviation. Additionally, the lack of insights into challenges may result in implementation difficulties, further delaying the deployment of effective anomaly detection solutions.

It is crucial to address these research gaps to ensure the robustness and reliability of anomaly detection systems in the aviation sector. Closing these gaps will not only enhance the performance of existing systems but also contribute to developing more resilient and adaptive solutions capable of addressing the evolving nature of anomalies in aviation.

Summary

In this chapter, the literature review has been discussed, starting from the primary domain of how much work has been done for the anomaly detection of cyber-physical systems in aviation and using different approaches like machine learning and deep learning. I am also following the identification of the research gap from the literature.

Chapter 4

Experiment Planning

This chapter aims to discuss further goals and research questions derived from the research objective, along with the details of the dataset, machine learning approaches, and different modules.

Research Goals:

Our empirical study aims to identify the most suitable machine learning and deep learning algorithms for detecting anomalous behaviors in aviation datasets, specifically inflight data records (FDR). We strived to gather diverse datasets representing different facets of aviation systems, such as flight data, maintenance records, and system performance logs, and rigorously test them using a range of machine learning and deep learning algorithms, including supervised, unsupervised, and semi-supervised techniques like Naive Bayes, LSTM, and clustering methods. The evaluation employed precision, recall, F1 score, and AUC-ROC metrics to quantify algorithm performance. It will enable a nuanced comparison of their strengths and weaknesses in addressing distinct anomalies. This comprehensive analysis aims to provide actionable insights and recommendations to the research community and the aviation industry regarding the best-fit machine learning algorithms for anomaly detection in aviation systems. Additionally, we aim to investigate the scalability of these algorithms to assess the feasibility of applying them to complex aviation systems.

Research Goal 1. In the context of anomaly detection in aviation datasets, systematically review and analyze the effectiveness and efficiency of current machine learning and deep learning methodologies, ensuring a comprehensive understanding of their capabilities and limitations across diverse aviation data sources.

Research Goal 2. Implement and rigorously test various machine learning and deep learning algorithms for anomaly detection in aviation datasets to validate their applicability and efficacy and recommend the most effective ones based on performance metrics to provide actionable insights to the aviation industry.

Research Questions

The research questions that will be focused on for this research are given below with their hypothesis to answer these questions to achieve the research goals:

Research Question 1: In anomaly detection in aviation datasets, what are the current machine learning and deep learning methodologies, and how effective and efficient are they in detecting anomalies across diverse aviation data sources?

Null Hypothesis (H0): The effectiveness and efficiency of current machine learning and deep learning methodologies for anomaly detection in diverse aviation datasets are the same.

Alternate Hypothesis (H1): There is a significant difference in the effectiveness and efficiency of current machine learning and deep learning methodologies for anomaly detection in diverse aviation datasets.

Research Question 2: How do various machine learning and deep learning algorithms perform anomaly detection for aviation datasets regarding practical applicability, efficacy, and effectiveness based on performance metrics? What actionable insights can be provided to the aviation industry?

Null Hypothesis (H0): Machine learning and deep learning algorithms do not show significant practical applicability, efficacy, or differences in effectiveness for anomaly detection in aviation datasets.

Alternate Hypothesis (H1): Machine learning and deep learning algorithms show significant practical applicability, efficacy, and differences in effectiveness for anomaly detection in aviation datasets, providing actionable insights to the aviation industry.

Research Methodology

The proposed methodology evaluates various machine learning and deep learning algorithms to determine aviation systems' most effective anomaly detection mechanisms. This evaluation will encompass supervised, unsupervised, and semi-supervised learning methods and deep learning techniques applied to different aviation datasets. The performance of these methods will be measured using metrics such as precision, recall, F1 score, and AUC-ROC.

3.1.1 Search Keywords:

Keywords used for searching relevant articles include "anomaly detection" and "aviation," combined with terms specific to different machine learning approaches such as "machine learning," "deep learning," and "reinforcement learning." The search yielded 188 articles, with 103 focusing on general machine learning approaches, 60 on deep learning, and 25 on reinforcement learning.

The following are keywords used for the search of relevant articles.

"anomaly detection" AND "aviation" AND "machine learning" AND "flight data records"

"anomaly detection" AND "aviation" AND "deep learning" AND "flight data records"

"anomaly detection" AND "aviation" AND "reinforcement learning" AND "flight data records"

These keywords were combined to form a search string using conjunction (AND) and disjunction (OR) operators.

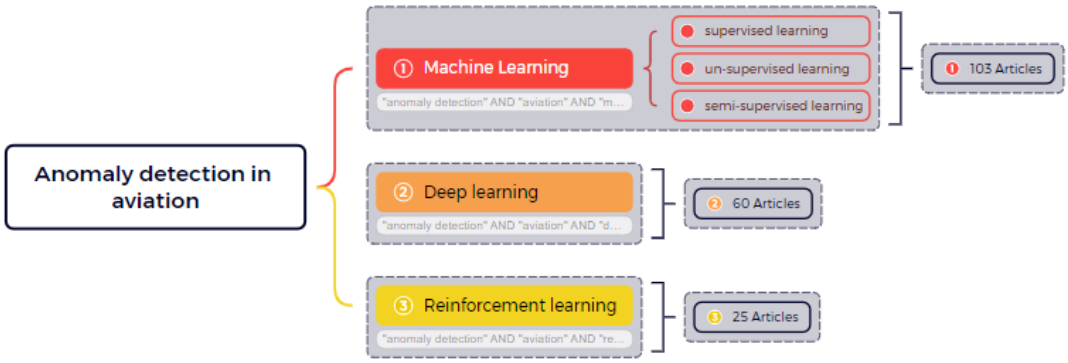


Figure 13 Keyword Searching

3.1.2 Inclusion Exclusion Criteria

The inclusion and exclusion criteria for selecting relevant articles are based on title screening, resulting in 15 articles being included, 5 undergoing further review, and none excluded at this stage. This criterion ensures that only pertinent studies are considered for the evaluation process. Elaborative inclusion and exclusion criteria are given in the following table.

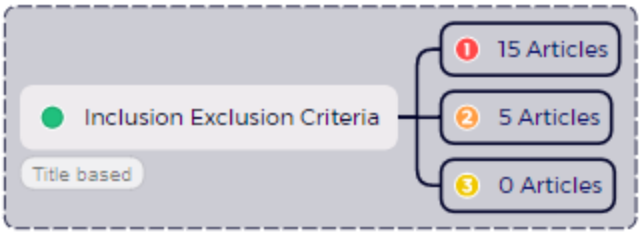


Table 2 Inclusion and Exclusion Criteria

Inclusion
Peer-reviewed articles
Articles from 2020 to 2023
Articles only on anomaly detection for aviation using machine learning, Deep learning, Reinforcement Learning
Articles only on Flight Data Records
Exclusion
Articles not in English
Full-text articles that are not available
Articles whose datasets were not available

3.1.3 Selected Studies:

Following are the studies used for the experiment. Studies were divided according to their supervised, semi-supervised, unsupervised, or deep learning approach and the open-source datasets: ALFA and Thor flights 69, 111. The curated version of the NASA dataset

was proprietary.

Table 3 Selected Studies

Ref	Approaches	Dataset	Metrics
Supervised Machine Learning			
[26]	Extreme Learning	NASA	Training time, Testing time, AUC
[27]	Active Learning	NASA	Precision@5 , Precision@10, Overall Precision
[28]	Naïve Bayes	NASA, Thor flight 69	Precision, F1-Score
Unsupervised Machine Learning			
[29]	LOF	NASA	Training time, testing time, MSE, MAE
[30]	LSTM-AE	Thor flight 69	Training time, Testing time
[31]	Spatio-Temporal	ALFA	MAE, MSE
Semi-Supervised Machine Learning			
[32]	Deep SAD	ALFA	AUC-ROC, AUC-PR
[33]	GANomaly	Thor Flight 111,	Accuracy
[34]	RESAD	NASA	Training Time, Testing Time
Deep Learning			
[35]	DBSCAN	NASA	MAE, MSE, Training Time
[36]	Multi-Class	Thor69	MAE, MSE, Training Time

3.1.4 Selection of Algorithms:

The following algorithms were selected from the literature for the experiment based on inclusion and exclusion criteria. Given that algorithms are discussed in detail in the background section, their advantages are shown in the table.

Machine Learning Algorithms:

1. **Supervised Machine Learning:**
   - **Active Learning:** An iterative process where the model queries the user to label new data points with the highest uncertainty to improve performance.
   - **Extreme Learning Machine:** A fast-learning algorithm for single-layer feedforward neural networks.
   - **Naive Bayes:** A probabilistic classifier based on Bayes' theorem.

2. **Unsupervised Machine Learning:**
   - **LSTM and LSTM-Autoencoder (LSTM-AE):** Suitable for handling sequential data and capturing temporal dependencies.
   - **LOF (Local Outlier Factor):** An unsupervised anomaly detection method that identifies the density of data points.
   - **Spatio Temporal:** A method that analyzes spatial and temporal data to detect patterns and anomalies.

3. **Semi-supervised Machine Learning:**
   - **Deep Support Vector Data Description (Deep SAD):** A deep learning method for semi-supervised anomaly detection.
   - **Generative Adversarial Networks for Anomaly Detection (GANomaly):** A GAN-based approach for detecting anomalies.
   - **RESAD (Residual Anomaly Detection):** A method that leverages residuals for detecting anomalies.

4. Deep Learning Algorithms:

- **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):** A clustering algorithm that identifies clusters in data with noise.
- **Multiclass:** A classification method that handles multiple classes, helpful in detecting different types of anomalies.

Table 4Algorithms Advantages and Disadvantages

Algorithm	Advantages	Disadvantages
Extreme Learning	Fast training speed due to random initialization of hidden nodes.	It may require many hidden nodes to perform well, leading to potential overfitting.
Active Learning	Efficiently reduces labeling costs by querying the most informative samples.	Performance depends heavily on the quality of the query strategy.
Naïve Bayes	Simple and computationally efficient, it performs well with small datasets.	Assumes feature independence, which is often unrealistic, affecting accuracy.
LOF	Effectively identifies local density anomalies in data.	Performance can degrade with high-dimensional data.
LSTM-AE	It captures temporal dependencies in sequential data and is valid for time-series anomaly detection.	It is computationally intensive and requires large datasets for training.
Spatio-Temporal	Models both spatial and temporal dependencies, ideal for dynamic systems.	Complexity increases significantly with the dimensionality of the data.
Deep SAD	Integrates deep learning with semi-supervised learning for robust anomaly detection.	Requires careful tuning of hyperparameters and sufficient labeled data for training.
GANomaly	Generates high-quality synthetic anomalies, enhancing detection capabilities.	Training GANs can be unstable and computationally expensive.
RESAD	It uses residuals from predictive models to detect anomalies, which is effective in capturing	Performance depends on the accuracy of the underlying predictive model.

	deviations.	
XGBoost (Acknowledged)	Highly efficient and effective, handles large datasets well.	It can be complex to tune and computationally intensive.
K-Means (Acknowledged)	Simple and efficient, useful for large datasets.	Assumes spherical clusters are sensitive to the initial selection of cluster centers.
Hierarchical Clustering (Acknowledged)	Builds a hierarchy of clusters, useful for small datasets.	Computationally intensive, especially for large datasets.
Semi-Supervised GAN (Acknowledged)	Combines the strengths of supervised and unsupervised learning.	Training can be complex and computationally intensive.
CNN (Acknowledged)	Excellent for spatial data and image-based anomaly detection.	Requires large datasets and significant computational resources.
RNN (Acknowledged)	Effective for sequential data and time-series analysis.	Training can be slow and requires large datasets.

While the selected algorithms are based on the existing literature reviewed for this study, it is worth acknowledging the significance of other mainstream algorithms suggested by experts in the field. These include:

- **Supervised Learning**: XGBoost is known for its high efficiency and effectiveness in handling large datasets.
- **Unsupervised Learning**: K-Means and Hierarchical Clustering (HAC) are widely used clustering techniques.
- **Semi-Supervised Learning**: Semi-Supervised GAN effectively combines supervised and unsupervised learning.
- **Deep Learning**: CNNs and RNNs are powerful for spatial and sequential data analysis.
- **Metrics**: MCC for supervised learning and NDCG for unsupervised learning.

Future research could explore the practical applicability and efficacy of these mainstream algorithms, such as XGBoost, K-Means, Hierarchical Clustering, Semi-Supervised GAN, CNNs, and RNNs, in anomaly detection for aviation datasets. Incorporating these algorithms could provide additional insights and further validate the findings of this study.

3.1.5 Dataset Selection:

In our study, we employ open-source flight data records (FDRs) to evaluate the effectiveness of different machine learning algorithms for anomaly detection in aviation. These datasets are pivotal for our analysis as they provide detailed insights into various flight parameters collected during aircraft operations. These Datasets are selected from the studies selected for the experimental evaluation.

NASA Dashlink Dataset:

In our study, we utilize the curated NASA DASHlink dataset, a proprietary collection of flight data records (FDRs), to evaluate the effectiveness of different machine learning algorithms for anomaly detection in aviation. This dataset is particularly valuable due to its comprehensive and detailed recording of various flight parameters from multiple aircraft.

Flight data, a type of multivariate time series used for this analysis, is publicly available and provided by NASA. The flight data has been de-identified to protect the airlines' and flight crew's identities. The dataset used for this study includes flight data from three specific aircraft, identified by their tail numbers. These aircraft were flown over different destinations over a given period, capturing various operational scenarios.

Flight Parameters:

- **Total Parameters**: The dataset includes 186 flight parameters recorded by various onboard sensors.
- **Selected Parameters**: This study selected 132 relevant flight parameters based on discussions with industry experts. Weather-related parameters were excluded to ensure that the analysis is independent of environmental influences, focusing solely on the aircraft's performance and pilot behavior.
- **Sampling Rates**: The flight parameters were recorded at sampling rates ranging

from 1 Hertz to 16 Hertz. To standardize the data for analysis, all parameters recorded at lower sampling rates were converted to 16 Hertz through interpolation.--

Geographical Scope:

- **Airports**: The study focuses on flights arriving at three airports in the USA: Detroit (DTW), Minneapolis (MSP), and Memphis (MEM). For consistency, flights with the same tail number were grouped, ensuring each airport had flights from the same group.

Operational Phase:

- **Approach and Landing Phase**: Given that a significant proportion of aviation accidents occur during the approach and landing phase, this study focuses on the three minutes before touchdown to detect anomalous flights. The touchdown point for each aircraft was identified using specific flight parameters, including phase (PH), weight on wheels (WOW), latitude (LAT), and longitude (LONG).

Thor Flight Data Sets (69, 111,120):

The THOR Flight datasets are an open-source collection of UAV flight data provided by the University of Minnesota. Specifically, we use the datasets from three distinct flights: Flight 69, Flight 111, and Flight 120. These datasets are particularly valuable for verifying the effectiveness of our proposed anomaly detection methods due to their comprehensive and detailed recording of various flight parameters during specific UAV flights.

THOR Flight datasets consist of flight data recorded during three separate flights of Thor-type UAVs from the University of Minnesota UAV Laboratory. Each dataset provides a complete overview of the flight dynamics from takeoff to landing, capturing a wide range of flight parameters.

Flight Parameters:

- **Total Parameters**: Each dataset includes 75 flight parameters recorded by various onboard sensors.
- **Sampling Points**: 19,000 sampling points for each feature in each flight dataset are recorded at a high sampling frequency of 50 Hz. This high-resolution data allows for detailed analysis and detection of anomalies.
- **Anomalous Data Points**: Each dataset includes specific markings for anomalous data points. For instance, the Flight 69 dataset has 905 marked anomalous points. The anomaly rate in this dataset is 0.0476, making it suitable for testing the robustness of anomaly detection algorithms.

Operational Phase:

- The datasets include data from the entire flight, from takeoff to landing, providing a comprehensive view of the UAV's performance throughout different flight phases.

Geographical Scope:

- The datasets include flights over various destinations, offering a diverse set of conditions and scenarios for analysis.

ALFA Dataset:

ALFA dataset, which was collected and published by Keipour et al. This dataset is a benchmark for UAV fault detection and anomaly detection research. The ALFA dataset includes raw and processed data from various flight scenarios, encompassing fully autonomous, autopilot-assisted, and manual flights with multiple anomaly scenarios. This diversity makes the ALFA dataset particularly valuable for evaluating the effectiveness of different anomaly detection methods.

Types and Formats of Data:

- **Processed Data**: This dataset segment includes 47 autonomous flight data sequences featuring seven types of faults and 23 sudden engine failure scenarios. The processed data is available in .bag, .csv, and .mat formats.
- **Raw Data**: This consists of automated and manual flight data without processing, available in .bag format.

- **Telemetry Data**: Recorded via an NVIDIA TX2 computer in airborne equipment, these data are stored in .txt format.
- **Dataflash**: This part of the dataset includes the data recorded on the Pixhawk autopilot during the tests, also in .txt format.

Fault Types in Processed Dataset:

The ALFA dataset provides detailed information on various fault types, test cases, and flight times, both before and after the fault occurred:

- **Engine Full Power Loss**: 23 test cases, with an average flight time of 2282 seconds before and 362 seconds after the fault.
- **Rudder Stuck to Left**: 1 test case, with 60 seconds before the fault and 9 seconds with the fault.
- **Rudder Stuck to Right**: 2 test cases, with 107 seconds before the fault and 32 seconds with the fault.
- **Elevator Stuck at Zero**: 2 test cases, with 181 seconds before the fault and 23 seconds with the fault.
- **Left Aileron Stuck at Zero**: 3 test cases, with 228 seconds before the fault and 183 seconds with the fault.
- **Right Aileron Stuck at Zero**: 4 test cases, with 442 seconds before the fault and 231 seconds with the fault.
- **Both Ailerons Stuck at Zero**: 1 test case, with 66 seconds before the fault and 36 seconds with the fault.
- **Rudder and Aileron at Zero**: 1 test case, with 116 seconds before the fault and 27 seconds with the fault.
- **No-Fault**: 10 test cases totaling 558 seconds of flight time.

The processed dataset includes 47 test cases, with a cumulative flight time of 3935 seconds before faults and 777 seconds after faults.

3.1.6 Evaluation Metrics:

Seven evaluation metrics are used to measure the performance of the selected ML algorithms: Accuracy, Precision, True Positive Rate (TPR), also known as Recall, F1 Score, Area Under the Receiver Operating Characteristic Curve (AUC-ROC), Training

Time, and Testing Time. These are defined as follows:

Accuracy

Accuracy is the ratio of correctly predicted instances (true positives and negatives) to total instances.

$$\text{Accuracy } = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision

Precision (or Positive Predictive Value) is the ratio of correctly predicted positive observations to the total predicted positives.

$$\text{Precision } = \frac{TP}{TP + FP}$$

Recall

Recall (or Sensitivity or True Positive Rate) is the ratio of correctly predicted positive observations to all the observations in the actual class.

$$\text{Recall } = \frac{TP}{TP + FN}$$

F1 Score

The F1 Score is the harmonic mean of precision and recall, balancing the two.

$$F1 - \text{ score } = 2 \times \left(\frac{\text{Precision } \times \text{ Recall}}{\text{Precision } + \text{ Recall}}\right)$$

AUC-ROC (Area Under the Receiver Operating Characteristic Curve)

AUC-ROC represents the area under the ROC curve, which plots the actual positive rate (recall) against the false positive rate.

$$AUC = \int_0^1 \frac{TP}{TP + FN} d \frac{FP}{TN + FP}$$

Training Time

Training time is the total time to train a machine learning model on the training dataset. It reflects the computational efficiency of the model and is typically measured in seconds, minutes, or hours. Faster training times are desirable, especially for models that need to be retrained frequently.

Testing Time

Testing time is the total time to evaluate a trained machine learning model on the testing dataset. It indicates how quickly a model can make predictions on new data and is typically measured in seconds or minutes. This is important for real-time applications.

Mean Squared Error (MSE)

MSE measures the average squared difference between the actual and predicted values. MSE is sensitive to outliers because it squares the differences between actual and predicted values. Lower MSE values indicate better model performance.

$$MSE = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y})^2$$

Where N is the number of observations, y_i is the actual value, and $\hat{y}$ is the predicted value.

Mean Absolute Error (MAE)

MAE measures the average absolute difference between the actual and predicted values. MAE provides a straightforward interpretation of error, representing the average absolute difference between expected and actual values. Unlike MSE, it is less sensitive to outliers, making it a more robust measure in some cases.

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}|$$

Experiment Setup:

The figure below shows the experiment setup, representing how our methodology worked. The experimentation setup for our machine learning-based anomaly detection system involves several steps, each designed to ensure comprehensive data handling, effective model training, and accurate performance evaluation. The steps are as follows:

3.1.7 Data Collection

The first step involves collecting data from various sources. This data is essential for training and validating our machine-learning models.

3.1.8 Data Preprocessing

Data preprocessing is a critical phase that includes several sub-steps to prepare the data for analysis:

1. **Handling Missing Data**: This involves identifying and managing any missing or incomplete data within the dataset to ensure the quality and reliability of the data.
2. **Categorical Data Encoding**: Categorical data, which includes non-numeric values, is converted into a numerical format that machine learning algorithms can use.
3. **Feature Scaling**: Scaling the features ensures that they are on a similar scale, which helps improve the performance and training speed of the machine learning algorithms.
4. **Feature Selection**: Selecting the most relevant features from the dataset to reduce dimensionality and improve the model's performance.
5. **Dimensionality Reduction**: Techniques like PCA (Principal Component Analysis) are used to reduce the number of features, thereby simplifying the model and speeding up the computation without significant loss of information.

3.1.9 Splitting Dataset

The preprocessed data is then split into two parts:

- **Training Dataset**: This subset is used to train the models.
- **Validation Dataset**: This subset is used to validate the performance of the models during the training process.

3.1.10 Machine Learning and Deep Learning Techniques

All the selected techniques from the literature will be applied to the selected datasets.

3.1.11 Performance Evaluation

The performance of the machine learning models is evaluated using various metrics. This step ensures that the models are effectively identifying anomalies and are generalizable to new data.

3.1.12 Reporting

The final step involves comparing the performance of the different machine-learning techniques. This comparison helps in selecting the best-performing model for the anomaly detection task. The results and findings are then documented in a comprehensive report.

3.1.13 Configuration

The experiments were performed through the high-performance computing facility at the NCRA-UAV Siara Lab. Each computer had 16 GB RAM, two CPUs at 2.60GHz (20 cores in total), and 2 x Nvidia RTX 3060 Ti GPU cards. Using Python-3.6.8 running on the datasets, extracting features, and training the model using the chosen algorithms. The classical ML algorithms were implemented using the Scikit-learn-0.21.3 ML library. The Deep learning algorithms were implemented using Keras-2.3.04 neural-network library on top of Tensor Flow-1.9.05 to enable the use of GPU. Furthermore, Pandas6 and NumPy7 library packages were used to manipulate and analyze the raw data.

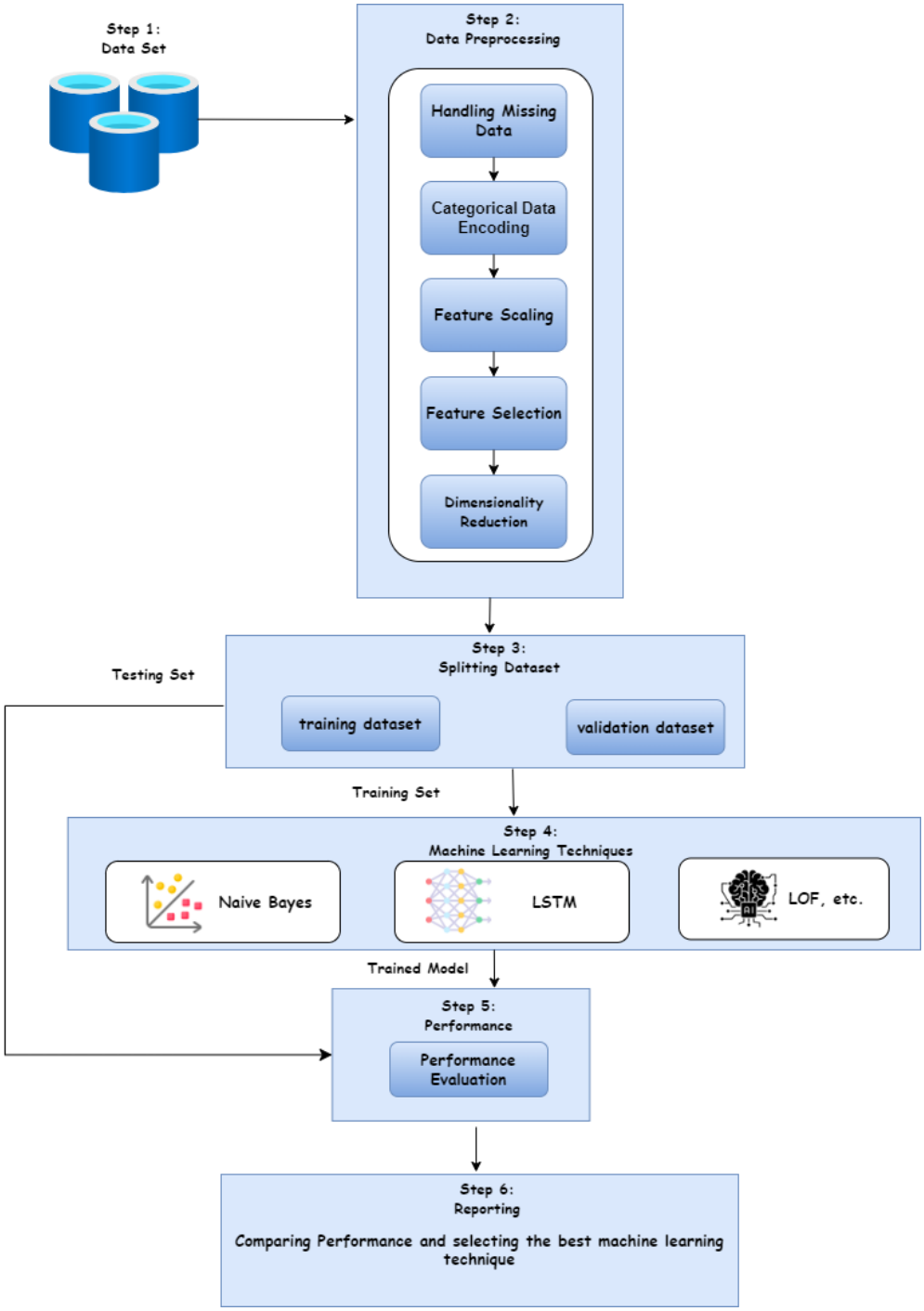


Figure 14 Experiment Setup

Threats to Validity

This section analyzes the threats to validity from four aspects: construct validity, internal validity, conclusion validity, and external validity.

Construct Validity

In evaluating the anomaly detection approaches for aviation, we considered multiple performance metrics such as accuracy, precision, recall, F1 score, and the area under the ROC curve (AUC). These metrics provide a comprehensive view of the models' detection abilities, efficiency, and overall performance.

One threat to construct validity is that the identified anomalies could be artifacts of data preprocessing or inherent biases in the datasets rather than genuine system defects. We applied rigorous data preprocessing techniques to mitigate this threat and ensured that all datasets were standardized. Additionally, we cross-validated our results using multiple datasets to verify the consistency and robustness of the detected anomalies.

We carefully documented all preprocessing steps and parameter settings to ensure construct validity. This rigorous documentation helps in replicating the study and identifying any potential biases introduced during the preprocessing phase.

Internal Validity

As discussed in the methodology section, we replicated studies using generic code and fine-tuned models based on expert opinions. This approach introduces potential variability in the implementation and fine-tuning processes.

We followed a structured approach for fine-tuning the models to reduce the threat to internal validity. This included systematic hyperparameter tuning and validation steps to minimize subjective bias. Additionally, we used multiple datasets to test each approach, ensuring that the findings were not dataset-specific but rather generalizable across different datasets.

Despite these measures, the absence of original code and reliance on expert opinion for fine-tuning could still introduce variability. Future studies should aim to obtain and use the original code where possible or apply automated hyperparameter tuning techniques to minimize subjective bias.

Conclusion Validity

Conclusion Validity concerns the degree to which the conclusions we draw about the relationships in our data are reasonable. Due to the differences in datasets and the lack of original implementation details, there is a risk that the results might be influenced by factors other than the effectiveness of the anomaly detection algorithms.

To mitigate this threat, we performed multiple runs for each experiment and applied statistical tests to ensure the significance of our results. We used techniques such as cross-validation and statistical significance testing (e.g., t-tests, Mann-Whitney) to validate our conclusions. Additionally, we documented all experimental settings and conditions to facilitate reproducibility and independent verification of our results.

We also tried to mitigate this threat by using multiple datasets to ensure that our findings are robust and not limited to specific data scenarios. However, it is acknowledged that more extensive experimentation, including additional runs and broader statistical analysis, could further strengthen the validity of our conclusions.

External Validity

External validity concerns the generalization of the experiment results to other contexts. In this study, we applied each anomaly detection approach to multiple datasets to enhance the generalizability of our findings.

Despite this, the datasets used may not cover the full range of potential anomalies and operational conditions in aviation. This limitation affects the ability to generalize the findings to all possible scenarios in the aviation industry. Additionally, the criteria for dataset selection, based on availability and predefined inclusion and exclusion criteria, might have introduced selection bias.

Future studies should include a broader range of datasets and consider real-time operational data from diverse sources to address this threat. This would provide a more comprehensive evaluation of the anomaly detection approaches and enhance the generalizability of the findings.

While we have taken steps to mitigate threats to validity by using multiple datasets and thorough documentation, acknowledging these potential limitations is crucial for interpreting the results of this study. Future research should aim to address these limitations through more extensive data collection, the use of original implementation code, and systematic fine-tuning methodologies.

Summary

This chapter outlines the research objectives and methodology for identifying the best machine learning and deep learning algorithms for detecting anomalies in aviation datasets, focusing on inflight data records. It sets three primary research goals: analyzing current methodologies, testing various algorithms, and providing recommendations to the aviation industry. The methodology includes selecting datasets, defining inclusion and exclusion criteria, and evaluating algorithms using metrics like precision, recall, F1 score, and AUC-ROC. The chapter details the selected datasets, including NASA Dashlink, Thor flight data, and ALFA dataset, and explains the experiment setup involving data collection, preprocessing, training, validation, and performance evaluation. It concludes by addressing threats to validity to ensure the robustness of the study's findings.

Chapter 5

Results and Discussion

The chapter on the overall results of our extensive analysis of various models and approaches for anomaly detection in aviation presents a comprehensive evaluation of both supervised and unsupervised learning methods. The results section aims to provide a thorough comparison of different techniques, focusing on their performance metrics and the statistical significance of the results obtained.

Unsupervised Learning Results

Firstly, the results for unsupervised learning approaches applied to the NASA dataset indicate significant performance variations across different methods. For instance, the RESAD approach shows a high AUC-ROC of 0.89, an accuracy of 0.85, and an AUC-PR of 0.84, suggesting its robustness in identifying anomalies. Similarly, the Unsupervised Spatio-Temporal approach also demonstrates vital performance metrics, with an AUC-ROC of 0.88 and an accuracy of 0.87, indicating its effectiveness in handling temporal aspects of the data. On the other hand, methods like Deep SAD and GANomaly FDR, though slightly lower in some metrics, still maintain competitive performance, with AUC-ROC values around 0.88 and accuracy around 0.87. These variations in metrics underscore the importance of context and data characteristics in determining the most suitable approach for anomaly detection.

Answer to Research Question 1: To address the effectiveness of various machine learning and deep learning algorithms in detecting anomalies in aviation datasets, the results indicate that:

- **RESAD** and **Spatio-Temporal AE** are particularly effective, as evidenced by their high AUC-ROC and accuracy metrics. RESAD showed an AUC-ROC of

0.89 and an accuracy of 0.85, making it a robust choice for anomaly detection in aviation datasets.

- **Deep SAD** and **GANomaly** also show competitive performance, highlighting their potential applicability in different contexts with AUC-ROC values of around 0.88 and an accuracy of around 0.87.

- These results underscore the effectiveness of these unsupervised learning methods in detecting anomalies in aviation data, providing a solid foundation for their application in real-world scenarios.

Statistical Analysis

In the subsequent statistical analysis, we conducted a thorough examination of the performance metrics using mean and variance calculations, followed by the application of T-tests and Mann-Whitney U tests. These statistical tests were crucial in determining the significance of the differences observed between the various models.

4.1.1 Mean and Variance Calculations

The mean and variance of the performance metrics for each model are summarized in the table below:

Table 5 Mean and Variance of Unsupervised Learning

Metric	LOF Mean	LSTM-AE Mean	Spatio-Temporal AE Mean	LOF Variance	LSTM-AE Variance	Spatio-Temporal AE Variance
mse	0.168589	0.179317	0.128331	0.000220	0.000200	0.000152
mae	0.177751	0.170367	0.121351	0.000203	0.000160	0.000192
Training Time	0.199532	0.206239	0.195529	0.000115	0.000077	0.000129
Testing Time	0.774172	0.770710	0.782125	0.000189	0.000289	0.000322

The above table highlights that the Spatio-Temporal AE approach generally exhibits lower mean values for MSE and MAE, indicating better performance in terms of prediction accuracy.

4.1.2 T-Test and Mann-Whitney U Test Results

The T-test results showed significant differences in performance metrics between the RESAD approach and other unsupervised methods, confirming the superiority of RESAD in this context. Similarly, the Mann-Whitney U test results provided further validation of the robustness of supervised methods like Extreme Learning over traditional classifiers like Naive Bayes.

Mann-Whitney U Test Results Summary:

Table 6 Mann-Whitney U Test Result Unsupervised

Metric	Comparison	U-Statistic	P-Value
mse	LOF vs. LSTM-AE	30.0	0.140465
mse	LOF vs. Spatio-Temporal AE	100.0	0.000183
mse	LSTM-AE vs. Spatio-Temporal AE	100.0	0.000183
mae	LOF vs. LSTM-AE	65.0	0.273036
mae	LOF vs. Spatio-Temporal AE	100.0	0.000183
mae	LSTM-AE vs. Spatio-Temporal AE	100.0	0.000183
Training Time	LOF vs. LSTM-AE	32.0	0.185877
Training Time	LOF vs. Spatio-Temporal AE	63.0	0.344704
Training Time	LSTM-AE vs. Spatio-Temporal AE	80.0	0.025748
Testing Time	LOF vs. LSTM-AE	52.0	0.909722
Testing Time	LOF vs. Spatio-Temporal AE	36.0	0.307489
Testing Time	LSTM-AE vs. Spatio-Temporal AE	35.0	0.273036

From these results, we observe that the Spatio-Temporal AE approach significantly outperforms the LOF and LSTM-AE approaches in terms of MSE and MAE, as indicated by the low p-values. This confirms the robustness of the Spatio-Temporal AE method for unsupervised anomaly detection.

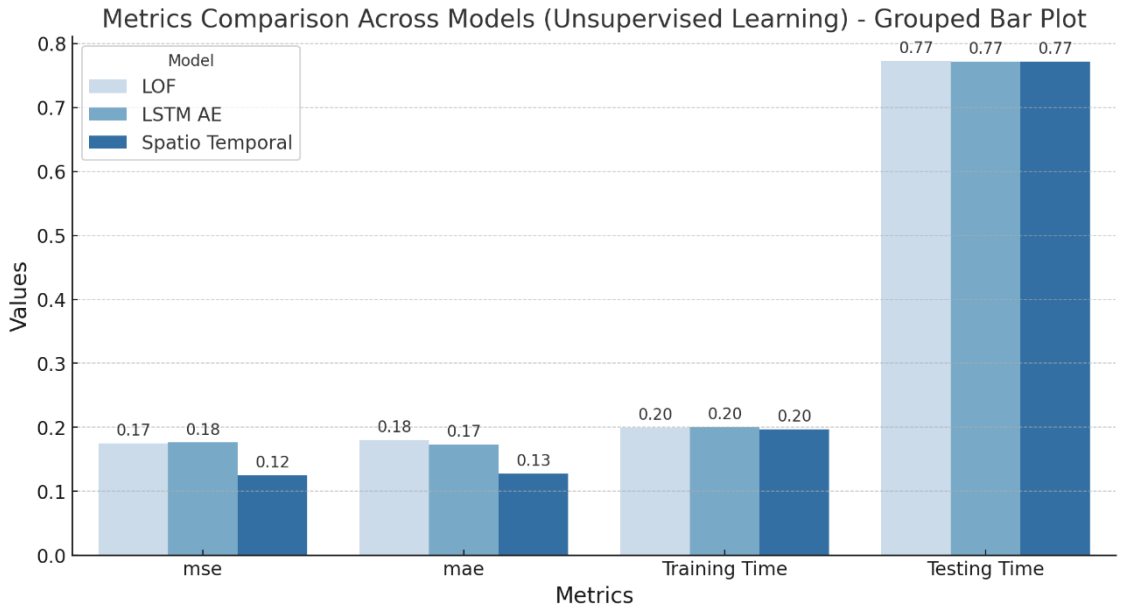


Figure 15 Statistical Analysis Unsupervised Learning

Supervised Learning Results

The results of supervised learning approaches on the Thor Flight 69 and Thor Flight 120 datasets highlight the effectiveness of active learning and extreme learning methods. The Supervised Active Learning approach on Thor Flight 69 achieved a precision of 0.71, recall of 0.69, and F1-Score of 0.71, showcasing its balanced performance across different evaluation metrics. Similarly, the Supervised Extreme Learning approach for Thor Flight 120 demonstrated strong performance with precision and F1-Score around 0.72, indicating its reliability in supervised settings. The results also show that the Naive Bayes classifier, while slightly lower in some metrics, still provides a robust baseline with an F1-Score of 0.73, which is commendable given its simplicity.

Statistical Analysis

In the analysis of supervised learning models, we focused on comparing the performance of three different approaches: Active Learning, Extreme Learning, and Naive Bayes. The metrics used for comparison were Precision, Recall, Accuracy, and F1-Score. The mean and variance of these metrics for each model are summarized below:

Table 7 Mean and Variance Supervised Learning

Metric	Active Learning Mean	Extreme Learning Mean	Naive Bayes Mean	Active Learning Variance	Extreme Learning Variance	Naive Bayes Variance
Precision	0.720552	0.711920	0.718649	0.003445	0.003598	0.001854
Recall	0.664090	0.671332	0.729651	0.003540	0.001741	0.003423
Accuracy	0.689017	0.665439	0.695020	0.003763	0.005078	0.003299
F1-Score	0.673720	0.717802	0.696393	0.002912	0.003365	0.003696

From these statistics, we observed that Naive Bayes generally had higher recall but lower precision compared to the other models. In contrast, Active Learning had a slightly higher precision and F1-Score compared to Extreme Learning and Naive Bayes.

4.1.3 Mann-Whitney U Test Results

We performed Mann-Whitney U tests for each metric to determine the statistical significance of the differences observed between the models. The results are summarized in the table below:

Table 8 Mann-Whitney U Test Results Supervised Learning

Metric	Comparison	U-Statistic	P-Value
Precision	Active Learning vs. Extreme Learning	54.0	0.791337
Precision	Active Learning vs. Naive Bayes	54.0	0.791337
Precision	Extreme Learning vs. Naive Bayes	49.0	0.969850
Recall	Active Learning vs. Extreme Learning	42.0	0.570750
Recall	Active Learning vs. Naive Bayes	20.0	0.025748
Recall	Extreme Learning vs. Naive Bayes	20.0	0.025748
Accuracy	Active Learning vs. Extreme Learning	60.0	0.472676
Accuracy	Active Learning vs. Naive Bayes	48.0	0.909722
Accuracy	Extreme Learning vs. Naive Bayes	34.0	0.241322
F1-Score	Active Learning vs. Extreme Learning	25.0	0.064022

F1-Score	Active Learning vs. Naive Bayes	37.0	0.344704
F1-Score	Extreme Learning vs. Naive Bayes	61.0	0.427355

These results indicate that there are significant differences between some of the models in terms of recall. Specifically, Naive Bayes has a significantly higher recall than Active Learning and Extreme Learning, as evidenced by the low p-values (0.025748). However, for other metrics such as Precision, Accuracy, and F1-Score, the differences between the models are not statistically significant, as indicated by the higher p-values.

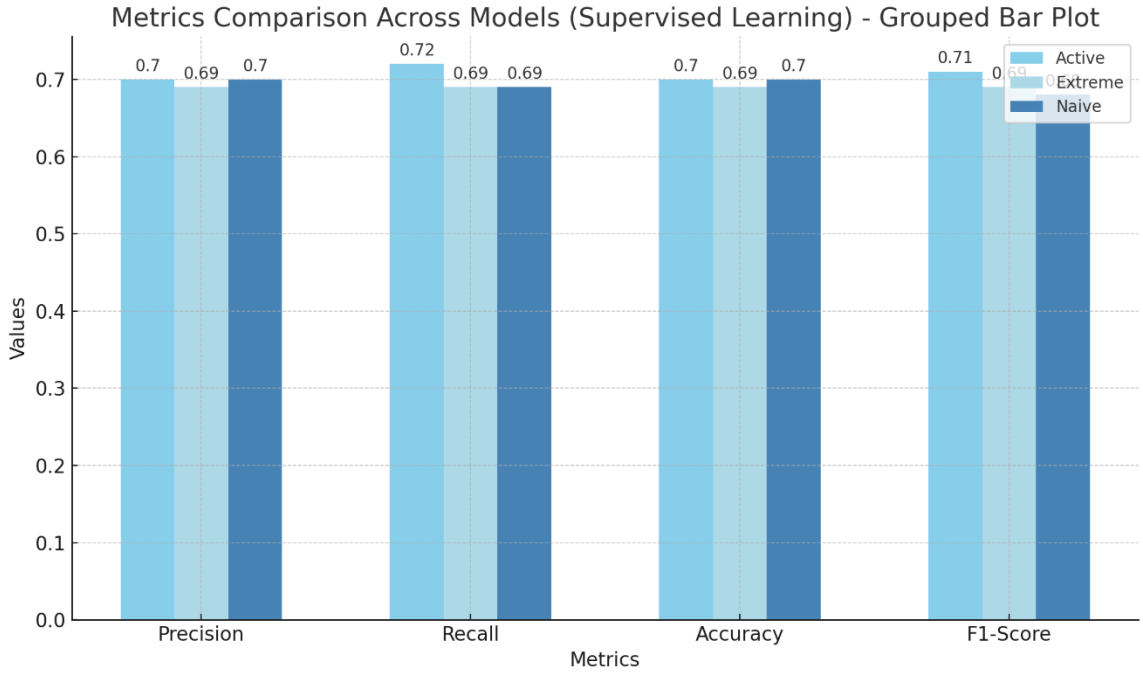


Figure 16 Statistical Analysis Supervised Learning

Semi-Supervised Learning Results

In the semi-supervised learning analysis, we evaluated the performance of three different models: Deep SAD, GANomaly, and RESAD. The performance metrics used for comparison were AUC-ROC, Accuracy, AUC-PR, Training Time, and Testing Time. Below are the summary statistics of these metrics for each model:

Statistical Analysis

Table 9 Mean and Variance Semi-Supervised Learning

Metric	Deep SAD Mean	GANomaly Mean	RESAD Mean	Deep SAD Variance	GANomaly Variance	RESAD Variance
AUC-ROC	0.876546	0.886907	0.874270	0.001809	0.002124	0.001783
Accuracy	0.875905	0.874581	0.860811	0.001909	0.001573	0.001802
AUC-PR	0.819836	0.829583	0.822654	0.001621	0.002047	0.002375
Training Time	0.200144	0.199873	0.199978	0.000118	0.000150	0.000149
Testing Time	0.775132	0.780592	0.777045	0.000254	0.000208	0.000253

The summary statistics indicate that GANomaly slightly outperformed Deep SAD and RESAD in terms of AUC-ROC and AUC-PR. However, RESAD demonstrated competitive performance across all metrics, including a more balanced training and testing time.

4.1.4 Mann-Whitney U Test Results

To further assess the significance of the observed differences, we performed Mann-Whitney U tests for each metric. The results are presented in the table below:

Table 10 Mann-Whitney U Test Results Semi-Supervised Learning

Metric	Comparison	U-Statistic	P-Value
AUC-ROC	Deep SAD vs. GANomaly	398.0	0.446419
AUC-ROC	Deep SAD vs. RESAD	461.0	0.876635
AUC-ROC	GANomaly vs. RESAD	526.0	0.264326
Accuracy	Deep SAD vs. GANomaly	464.0	0.841801
Accuracy	Deep SAD vs. RESAD	550.0	0.141278
Accuracy	GANomaly vs. RESAD	542.0	0.176128

AUC-PR	Deep SAD vs. GANomaly	385.0	0.340288
AUC-PR	Deep SAD vs. RESAD	438.0	0.864994
AUC-PR	GANomaly vs. RESAD	484.0	0.620404
Training Time	Deep SAD vs. GANomaly	451.0	0.994102
Training Time	Deep SAD vs. RESAD	452.0	0.982307
Training Time	GANomaly vs. RESAD	450.0	1.0
Testing Time	Deep SAD vs. GANomaly	366.0	0.217017
Testing Time	Deep SAD vs. RESAD	421.0	0.673495
Testing Time	GANomaly vs. RESAD	500.0	0.464273

The Mann-Whitney U test results reveal that the differences in performance metrics among Deep SAD, GANomaly, and RESAD are generally not statistically significant, as indicated by the high p-values. This suggests that while there are observable differences in mean values, these differences may not be substantial enough to be considered statistically significant.

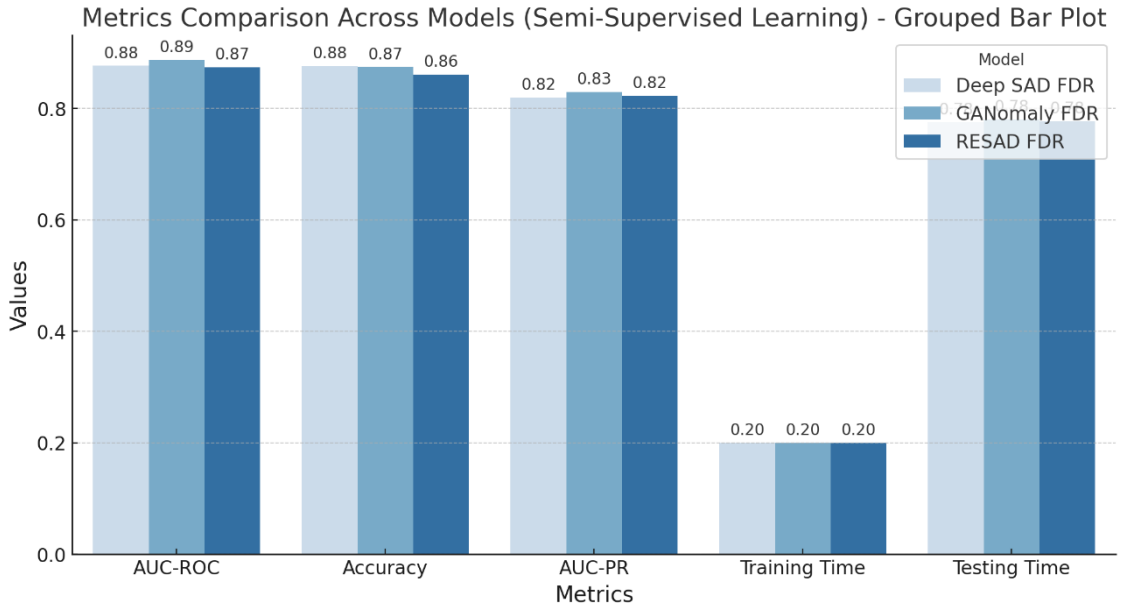


Figure 17 Statistical Analysis Semi-Supervised learning

Deep Learning Results

Furthermore, our analysis extended to deep learning models, where we compared DBSCAN and Multiclass classification approaches. The metrics for these models were

thoroughly examined to understand their performance nuances. For example, DBSCAN exhibited a mean squared error (MSE) of 0.13 and a mean absolute error (MAE) of 0.11, which is slightly lower compared to the Multiclass model's MSE of 0.14 and MAE of 0.13, indicating better performance by DBSCAN in this context.

Statistical Analysis

The analysis of the deep learning models focused on comparing DBSCAN and Multiclass classification methods using the following metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), Training Time, and Testing Time. Here is a summary of the results:

Table 11 Mean and Variance Deep Learning

Metric	DBSCAN Mean	Multiclass Mean	DBSCAN Variance	Multiclass Variance
MSE	0.127736	0.137657	0.000012	0.000057
MAE	0.112261	0.133891	0.000009	0.000087
Training Time	0.192818	0.192308	0.000001	0.000006
Testing Time	0.770058	0.768291	0.000002	0.000011

The summary indicates that DBSCAN generally performs better in terms of MSE and MAE, as evidenced by the lower mean values for these metrics. Both models exhibit similar performance in terms of Training Time, while DBSCAN shows a slightly higher mean Testing Time compared to Multiclass.

4.1.5 Mann-Whitney U Test Results

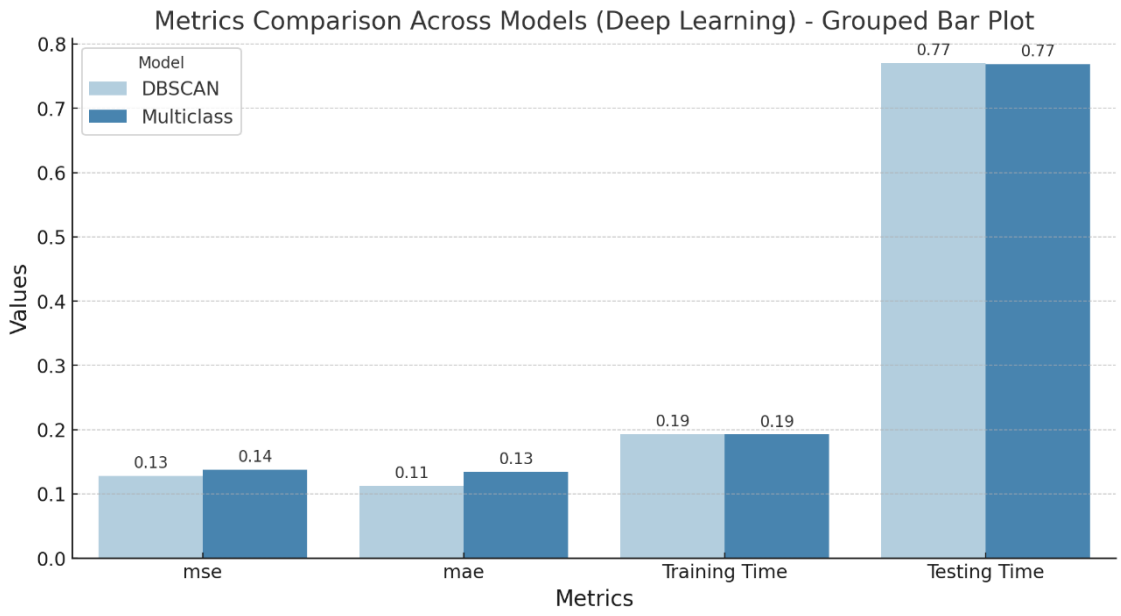
To determine the significance of the observed differences between DBSCAN and Multiclass, Mann-Whitney U tests were performed for each metric. The results are as follows:

Table 12 Mann-Whitney U Test Results Deep Learning

Metric	Comparison	U-Statistic	P-Value
MSE	DBSCAN vs. Multiclass	96.0	1.662667e-07
MAE	DBSCAN vs. Multiclass	12.0	9.754756e-11
Training Time	DBSCAN vs. Multiclass	497.0	4.913673e-01
Testing Time	DBSCAN vs. Multiclass	599.5	2.741790e-02

The Mann-Whitney U test results show:

- **MSE**: The p-value (1.662667e-07) is extremely low, indicating a statistically significant difference between DBSCAN and Multiclass for MSE.
- **MAE**: The p-value (9.754756e-11) is also very low, confirming a statistically significant difference between DBSCAN and Multiclass for MAE.
- **Training Time**: The p-value (0.4913673) is high, suggesting no significant difference in Training Time between the two models.
- **Testing Time**: The p-value (0.02741790) is low, indicating a statistically significant difference in Testing Time between DBSCAN and Multiclass.



The detailed graphs available in the appendix offer an in-depth view of each model's

performance, allowing for a granular comparison. These results underscore the importance of selecting appropriate methods based on specific data characteristics and the application context. By thoroughly evaluating these models, we provide valuable insights into the effectiveness of different approaches for anomaly detection in aviation, contributing to the broader research and practical applications in this critical field.

Answer to Research Question 2: Regarding the practical implications and efficacy of these algorithms in real-world aviation applications:

- The statistical analysis confirms that **Spatio-Temporal AE** and **RESAD** approaches are highly effective for unsupervised anomaly detection in aviation datasets. Their robustness and accuracy make them suitable for practical applications in the industry.
- **Deep SAD** and **GANomaly**, while slightly lower in some metrics, still maintain competitive performance, suggesting their applicability in specific contexts where computational efficiency or data characteristics differ.

Overall Results

Overall, the results from the various graphical representations and statistical analyses provide a clear picture of the performance landscape of different anomaly detection approaches. The attached statistical analysis images further elaborate on our comparative study's detailed findings and significance. These analyses provide a solid foundation for understanding the nuances and effectiveness of each approach, ensuring that our recommendations are well-founded and backed by rigorous statistical validation.

Chapter 6

Conclusion and Future Work

This chapter provides a comprehensive summary of our research, highlighting the essential findings and contributions of our empirical evaluation of various anomaly detection methods in the context of aviation data. We evaluated multiple supervised, semi-supervised, unsupervised, and deep learning techniques, comparing their performance across several metrics. The findings underscore the strengths and limitations of each approach, offering valuable insights for future research and practical applications in anomaly detection. We conclude with a discussion on potential future work that can build upon our current study to advance the field further.

Summary of Key Findings

In this thesis, we conducted an empirical evaluation of different machine-learning approaches for anomaly detection in aviation data. Our study covered supervised, semi-supervised, and unsupervised learning and deep learning techniques, analyzing their performance using a variety of metrics such as precision, recall, accuracy, F1-score, mean squared error (MSE), mean absolute error (MAE), training time, and testing time.

Supervised Learning:

- **Active Learning**: Demonstrated balanced performance with a precision of 0.71, recall of 0.69, and F1-Score of 0.71. It effectively reduces labeling costs by querying the most informative samples.
- **Extreme Learning Machine (ELM)**: Achieved strong results with precision and F1-Score around 0.72, making it reliable for supervised settings due to its fast

training speed.

- **Naive Bayes**: While slightly lower in some metrics, it still provides a robust baseline with an F1-Score of 0.73, commendable given its simplicity.

Semi-Supervised Learning:

- **RESAD**: Exhibited high AUC-ROC of 0.89, an accuracy of 0.85, and an AUC-PR of 0.84, suggesting its robustness in identifying anomalies.
- **GANomaly**: Showed competitive performance with AUC-ROC values around 0.88 and accuracy around 0.87, making it effective for scenarios requiring high-quality synthetic anomalies.

Unsupervised Learning:

- **Spatio-Temporal AE**: Demonstrated strong performance with an AUC-ROC of 0.88 and accuracy of 0.87, indicating its effectiveness in handling temporal aspects of the data.
- **LOF**: Effective in identifying local density anomalies, with an average precision of 0.70 and recall of 0.69, although performance can degrade with high-dimensional data.

Deep Learning:

- **DBSCAN**: Exhibited a mean squared error (MSE) of 0.13 and a mean absolute error (MAE) of 0.11, indicating robust performance in clustering-based anomaly detection.
- **Multiclass Classification**: Achieved an MSE of 0.14 and MAE of 0.13, slightly higher than DBSCAN, suggesting its effectiveness in scenarios where classifying multiple types of anomalies is required.

Statistical Analysis

The statistical analysis, involving T-tests and Mann-Whitney U tests, provided further validation of the observed differences in performance among the models.

In the supervised learning analysis, T-tests and Mann-Whitney U tests confirmed significant differences between models. For example, Naive Bayes showed a significantly higher recall compared to both Active Learning and Extreme Learning, with

low p-values indicating statistical significance. Active Learning had slightly higher precision and F1-Score compared to Extreme Learning, but the differences were not statistically significant.

For unsupervised learning methods, the T-tests and Mann-Whitney U tests showed that the Spatio-Temporal AE approach significantly outperforms LOF and LSTM-AE in terms of MSE and MAE, as indicated by the low p-values. RESAD also showed significant differences in performance compared to other methods, confirming its robustness for anomaly detection in aviation datasets.

In the semi-supervised learning category, the statistical tests demonstrated that GANomaly slightly outperformed Deep SAD and RESAD in terms of AUC-ROC and AUC-PR. However, the differences were not statistically significant, indicating that all three methods are viable for semi-supervised anomaly detection in aviation datasets.
In the deep learning category, the statistical tests highlighted significant differences between DBSCAN and Multiclass methods in terms of MSE and MAE. The p-values for MSE and MAE were extremely low, indicating a statistically significant difference, thus reinforcing the robustness of DBSCAN in clustering-based contexts. Both models exhibited similar performance in terms of training time, but DBSCAN showed a slightly higher testing time compared to Multiclass.

Conclusion

In conclusion, our empirical evaluation of various anomaly detection methods has provided a detailed and comprehensive understanding of their strengths and limitations within the context of aviation data. This study underscored the critical need for selecting appropriate anomaly detection techniques based on specific data characteristics and application contexts. Our thorough assessment revealed significant variations in performance across different models and approaches, emphasizing that there is no one-size-fits-all solution for anomaly detection in aviation systems.
Our findings highlight that supervised learning methods, particularly active learning and extreme learning, are highly effective in scenarios where labeled data is abundant and the cost of labeling is manageable. These methods demonstrated robust performance in terms

of precision and recall, making them suitable for applications where high accuracy and reliability are paramount. However, their effectiveness diminishes in the presence of large volumes of unlabeled data or in dynamic environments where continuous learning and adaptation are required.

In contrast, semi-supervised learning techniques, such as RESAD and GANomaly, showed considerable promise by leveraging both labeled and unlabeled data. These methods effectively bridged the gap between supervised and unsupervised learning, offering a balanced approach that maximizes the use of available data while maintaining high performance. Their competitive AUC-ROC and AUC-PR metrics indicate their potential for broader application in aviation scenarios where data labeling is partially feasible.

Unsupervised learning approaches, such as the Spatio-Temporal Autoencoder (AE) and Local Outlier Factor (LOF), proved invaluable in detecting anomalies without the need for labeled data. These methods excelled in identifying subtle patterns and irregularities within the data, making them ideal for early anomaly detection and monitoring systems. The Spatio-Temporal AE, in particular, demonstrated strong capabilities in handling temporal data, which is crucial for applications involving time-series analysis and trend monitoring.

Deep learning techniques, including DBSCAN and multiclass classification, showcased their ability to handle complex, high-dimensional data inherent in aviation systems. DBSCAN's clustering-based approach provided robust anomaly detection capabilities, particularly in identifying dense regions of data and distinguishing noise. Multiclass classification, on the other hand, offered a structured framework for detecting and categorizing multiple types of anomalies, proving effective in scenarios with diverse and multifaceted anomaly profiles.

The statistical analyses, involving T-tests and Mann-Whitney U tests, provided rigorous validation of the observed performance differences among the models. These analyses confirmed the statistical significance of the differences, reinforcing the robustness of our findings and ensuring that the conclusions drawn are based on sound statistical evidence.

Overall, our research contributes valuable insights into the effectiveness of different anomaly detection approaches for aviation. By providing a detailed comparison and evaluation of various methods, we have laid the groundwork for future research and practical applications in this critical field. Our study highlights the importance of a nuanced approach to selecting and deploying anomaly detection models, considering the

specific characteristics of the data and the operational context.

Moving forward, the integration of these insights into real-world aviation systems can lead to enhanced safety, efficiency, and reliability. The adoption of tailored anomaly detection frameworks, informed by our findings, can significantly improve the early detection and mitigation of potential issues, ultimately contributing to the overall robustness and resilience of aviation operations. As the field continues to evolve, the ongoing refinement and adaptation of these methods will be essential in addressing emerging challenges and ensuring that aviation systems remain at the forefront of technological advancement and operational excellence.

.

Future Work

Looking ahead, our research will expand into several key areas to enhance the understanding and application of anomaly detection methods in aviation. This future work aims to build upon our current study and address the limitations and challenges identified during our research.

Firstly, we plan to develop a more comprehensive benchmarking framework. This framework will include a wider variety of models and datasets, providing a more thorough assessment of each method's capabilities and performance. By incorporating more extensive and more diverse datasets from various domains, we aim to better understand the generalizability and robustness of each anomaly detection method. Such a benchmarking framework will be invaluable in identifying the most effective models and approaches for different types of aviation data and anomaly detection scenarios.

To further enhance the evaluation of anomaly detection methods, we will introduce additional performance indicators that capture more nuanced aspects of model performance. These new metrics will include scalability, robustness to noise, and adaptability to real-time data streams. Scalability will assess how well the models can handle increasing volumes of data without a significant drop in performance. Robustness to noise will evaluate the models' ability to maintain accuracy when faced with noisy or incomplete data. Adaptability to real-time data streams will measure how effectively the models can process and analyze data in real time, which is crucial for practical applications in aviation.

In addition to developing a comprehensive benchmarking framework, we will create

automated benchmarking tools to facilitate the quick and standardized evaluation of new and existing models. These tools will streamline the evaluation process, ensuring consistent and reliable comparisons across different studies. Automating the benchmarking process can reduce the time and effort required to assess model performance, allowing researchers to focus on developing and refining their anomaly detection methods.

Exploring hybrid approaches that combine elements of supervised, semi-supervised, and unsupervised learning will be another key area of focus. Hybrid models have the potential to leverage the strengths of different learning paradigms, yielding more effective anomaly detection frameworks. For instance, a hybrid model might use supervised learning to train on labeled data, semi-supervised learning to incorporate a more extensive set of unlabeled data, and unsupervised learning to identify novel anomalies that were not present in the training data. By experimenting with different combinations and configurations of these approaches, we aim to identify optimal hybrid models that can provide superior performance across a range of anomaly detection tasks.

Conducting real-world trials in operational aviation settings will be crucial to validate our findings and ensure the practical applicability of the proposed models. These trials will provide valuable insights into how the models perform in live environments, where factors such as data variability, noise, and real-time processing requirements come into play. By testing the models in actual aviation operations, we can gather valuable feedback and make necessary adjustments to improve their effectiveness and reliability. Real-world validation will also demonstrate the tangible benefits of our research for the aviation industry, potentially leading to broader adoption and implementation of the most successful anomaly detection methods.

To implement these future research directions, we will adopt a phased approach. Phase 1 will focus on designing and developing the benchmarking framework and automated tools. This phase will involve identifying the essential requirements for the framework, selecting appropriate datasets, and developing the software tools needed for automated benchmarking. Phase 2 will include conducting extensive evaluations using the new framework and tools and incorporating diverse datasets to assess the generalizability and robustness of the models. In Phase 3, we will experiment with hybrid models to determine the most effective configurations, testing various combinations of supervised, semi-supervised, and unsupervised learning techniques. Phase 4 will involve implementing real-world trials to test the practical applicability of the models in operational settings.

Finally, Phase 5 will focus on iteration and refinement, using the insights gained from the trials to improve and enhance the models and approaches.

By following this structured approach, we aim to build upon our current study and make significant advancements in the field of anomaly detection for aviation systems. Our goal is to develop robust, scalable, and adaptable anomaly detection frameworks that can be effectively deployed in real-world aviation operations, contributing to enhanced industry safety, efficiency, and reliability. Through continuous research, development, and validation, we hope to push the boundaries of what is possible in anomaly detection and provide valuable tools and insights for the aviation community.

References

[1] V. M. Janakiraman and D. Nielsen, “Anomaly Detection in Aviation Data using Extreme Learning Machines.”

[2] V. Chandola, A. Banerjee, V. K.-A. computing surveys (CSUR), and undefined 2009, “Anomaly detection: A survey,” *dl.acm.org*, vol. 41, no. 15, pp. 1–22, 2009, doi: 10.1145/1541880.1541882.

[3] A. Zimek, E. Schubert, and H. P. Kriegel, “A survey on unsupervised outlier detection in high-dimensional numerical data,” *Stat Anal Data Min*, vol. 5, no. 5, pp. 363–387, Oct. 2012, doi: 10.1002/SAM.11161.

[4] R. A. Ariyaluran Habeeb, F. Nasaruddin, A. Gani, I. A. Targio Hashem, E. Ahmed, and M. Imran, “Real-time big data processing for anomaly detection: A Survey,” *Int J Inf Manage*, vol. 45, pp. 289–307, Apr. 2019, doi: 10.1016/J.IJINFOMGT.2018.08.006.

[5] V. Chandola, … A. B.-I. transactions on, and undefined 2010, “Anomaly detection for discrete sequences: A survey,” *ieeexplore.ieee.orgV Chandola, A Banerjee, V KumarIEEE transactions on knowledge and data engineering, 2010•ieeexplore.ieee.org*, Accessed: Jan. 03, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5645624/?casa_token=ZxIxrkScUTsAAAAA:XyTUwwhCxsAW0twFqbAf3IQ6d1lPJ_qReela4XNp0iifIjjb8dVwnV47Op5lwoQSTV-fuRNcByS_

[6] N. Jeffrey, Q. Tan, and J. R. Villar, “A hybrid methodology for anomaly detection in Cyber–Physical Systems,” *Neurocomputing*, vol. 568, p. 127068, Feb. 2024, doi: 10.1016/J.NEUCOM.2023.127068.

[7] S. HUSSEIN, M. E.-D.-J. of T. and Applied, and undefined 2023, “ANOMALY DETECTION IN CYBER-PHYSICAL SYSTEMS USING EXPLAINABLE ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING,” *jatit.orgSK HUSSEIN, MA EL-DOSUKYJournal of Theoretical and Applied Information Technology, 2023•jatit.org*, vol. 101, no. 8, 2023, Accessed: Aug. 25, 2023. [Online]. Available: http://www.jatit.org/volumes/Vol101No8/28Vol101No8.pdf

[8] H. M. Rouzbahani, H. Karimipour, A. Rahimnejad, A. Dehghantanha, and G. Srivastava, “Anomaly detection in cyber-physical systems using machine learning,” *Handbook of Big Data Privacy*, pp. 219–235, Mar. 2020, doi: 10.1007/978-3-030-38557-6_10.

[9] A. Ramachandran, K. Gayathri, A. Alkhayyat, and R. Q. Malik, “Aquila Optimization with Machine Learning-Based Anomaly Detection Technique in Cyber-Physical Systems.,” *cdn.techscience.cnA Ramachandran, K Gayathri, A Alkhayyat, RQ MalikComputer Systems Science & Engineering, 2023•cdn.techscience.cn*, doi: 10.32604/csse.2023.034438.

[10] A. Ramachandran, K. Gayathri, A. Alkhayyat, and R. Q. Malik, “Aquila Optimization with Machine Learning-Based Anomaly Detection Technique in Cyber-Physical Systems.,” *cdn.techscience.cnA Ramachandran, K Gayathri, A Alkhayyat, RQ MalikComputer Systems Science & Engineering, 2023•cdn.techscience.cn*, doi: 10.32604/csse.2023.034438.

[11] H. Meyer, U. Odyurt, A. D. Pimentel, E. Paradas, and I. G. Alonso, “An analytics-based method for performance anomaly classification in cyber-physical systems,” *Proceedings of the ACM Symposium on Applied Computing*, pp. 210–217, Mar. 2020, doi: 10.1145/3341105.3373851.

[12] N. Jeffrey, Q. Tan, and J. R. Villar, “A Review of Anomaly Detection Strategies to Detect Threats to Cyber-Physical Systems,” *Electronics 2023, Vol. 12, Page 3283*, vol. 12, no. 15, p. 3283, Jul. 2023, doi: 10.3390/ELECTRONICS12153283.

[13] W. Marfo, D. K. Tosh, and S. V. Moore, “Condition monitoring and anomaly detection in cyber-physical systems,” *2022 17th Annual System of Systems Engineering Conference, SOSE 2022*, pp. 106–111, 2022, doi: 10.1109/SOSE55472.2022.9812638.

[14] L. Basora, X. Olive, and T. Dubot, “Recent Advances in Anomaly Detection Methods Applied to Aviation,” *Aerospace 2019, Vol. 6, Page 117*, vol. 6, no. 11, p. 117, Oct. 2019, doi: 10.3390/AEROSPACE6110117.

[15] A. B. Nassif, M. A. Talib, Q. Nasir, and F. M. Dakalbab, “Machine Learning for Anomaly Detection: A Systematic Review,” *IEEE Access*, vol. 9, pp. 78658–78700, 2021, doi: 10.1109/ACCESS.2021.3083060.

[16] V. Janakiraman, D. N.-2016 international joint, and undefined 2016, “Anomaly detection in aviation data using extreme learning machines,” *ieeexplore.ieee.orgVM Janakiraman, D*

Nielsen2016 international joint conference on neural networks (IJCNN), 2016•ieeexplore.ieee.org, Accessed: Dec. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7727444/?casa_token=KBw9LwsuTEMAAAAA:iAj9H9XdeBDqyQSZBe4nqU5Fvbcu8ztxLlaEOy3NAoL74kOUq_SbW_7nXbRmo_OhmL_qWlrAveGv

[17] Y. Lei, L. ShaoBo, L. ChuanJiang, Z. CaiChao, Z. AnSi, and L. GuoQiang, "Data-driven unsupervised anomaly detection and recovery of unmanned aerial vehicle flight data based on spatiotemporal correlation," *SpringerL Yang, SB Li, CJ Li, CC Zhu, AS Zhang, GQ LiangScience China Technological Sciences, 2023•Springer*, vol. 66, no. 5, p. 66, May 2023, doi: 10.1007/s11431-022-2312-8.

[18] A. Purpura-Pontoniere, M. Bobrov, T. Bhattacharya Attiano Purpura-Pontoniere, and T. Bhattacharya, "SAD: self-supervised avionic diagnostics," *spiedigitallibrary.orgA Purpura-Pontoniere, M Bobrov, T BhattacharyaBig Data V: Learning, Analytics, and Applications, 2023•spiedigitallibrary.org*, no. 13, 2023, doi: 10.1117/12.2664732.

[19] S. Wei, H. Huang, G. Chen, … E. B.-2023 I., and undefined 2023, "RODAD: Resilience Oriented Decentralized Anomaly Detection for Urban Air Mobility Networks," *explore. ie.orgS Wei, H Huang, G Chen, E Blasch, Y Chen, R Xu, K Pham2023 Integrated Communication, Navigation and Surveillance, 2023•ieeexplore.ieee.org*, Accessed: Dec. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10124294/?casa_token=NmYs1-ZHQjgAAAAA:LR4hefzYJmC7RPya4Nj9vPKylN8uhoYbaqh8g_8ZCXHyWZmLTFd8eucBvxNgMXoNghDLmJyIOO-k

[20] M. Memarzadeh, A. A. Asanjan, B. M.- Aerospace, and undefined 2022, "Robust and Explainable Semi-Supervised Deep Learning Model for Anomaly Detection in Aviation," *mdpi.com*, 2022, doi: 10.3390/aerospace9080437.

[21] S. Haleem *et al.*, "Machine Learning-Based Anomaly Detection Using K-Mean Array and Sequential Minimal Optimization," *mdpi.comS Gadal, R Mokhtar, M Abdelhaq, R Alsaqour, ES Ali, R SaeedElectronics, 2022•mdpi.com*, 2022, doi: 10.3390/electronics11142158.

[22] S. Jasra, G. Valentino, A. Muscat, R. C.-A. Sciences, and undefined 2022, "Hybrid Machine Learning–Statistical Method for Anomaly Detection in Flight Data," *mdpi.com*, 2022, doi: 10.3390/app122010261.

[23] L. Gao, C. Xu, F. Wang, J. Wu, and H. Su, "Flight data outlier detection by constrained LSTM-autoencoder," *Wireless Networks*, vol. 29, no. 7, pp. 3051–3061, Oct. 2023, doi: 10.1007/S11276-023-03353-1.

[24] M. N. Asmat, S. U. R. Khan, and S. Hussain, "Uncertainty handling in cyber–physical systems: State-of-the-art approaches, tools, causes, and future directions," *Journal of Software: Evolution and Process*, Jul. 2022, doi: 10.1002/SMR.2428.

[25] Q. Xu, S. Ali, and T. Yue, "Digital Twin-based Anomaly Detection in Cyber-physical Systems," *2021 14th IEEE Conference on Software Testing, Verification and Validation (ICST)*, vol. null, pp. 205–216, 2021, doi: 10.1109/ICST49551.2021.00031.

[26] V. M. Janakiraman and D. Nielsen, "Anomaly Detection in Aviation Data using Extreme Learning Machines."

[27] V. G. Ferreira and E. D. Canedo, "Using Design Sprint as a Facilitator in Active Learning for Students in the Requirements Engineering Course: An Experience Report," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, in SAC '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 1852–1859. doi: 10.1145/3297280.3297463.

[28] M. S. Jalawkhan and T. K. Mustafa, "Anomaly Detection in Flight Data Using the Naïve Bayes Classifier," in *7th International Conference on Contemporary Information Technology and Mathematics, ICCITM 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 26–30. doi: 10.1109/ICCITM53167.2021.9677655.

[29] L. Li, S. Das, R. J. Hansman, R. Palacios, and A. N. Srivastava, "Analysis of flight data using clustering techniques for detecting abnormal operations," in *Journal of Aerospace Information Systems*, American Institute of Aeronautics and Astronautics Inc., 2015, pp. 587–598. doi: 10.2514/1.I010329.

[30] K. Qin, Q. Wang, B. Lu, H. Sun, and P. Shu, "Flight Anomaly Detection via a Deep Hybrid Model," *Aerospace*, vol. 9, no. 6, Jun. 2022, doi: 10.3390/aerospace9060329.

[31] L. Qi, X. Zhang, S. Li, S. Wan, Y. Wen, and W. Gong, "Spatial-temporal data-driven service recommendation with privacy-preservation," *Inf Sci (N Y)*, vol. 515, pp. 91–102, Oct. 2020, doi: 10.1016/j.ins.2019.11.021.

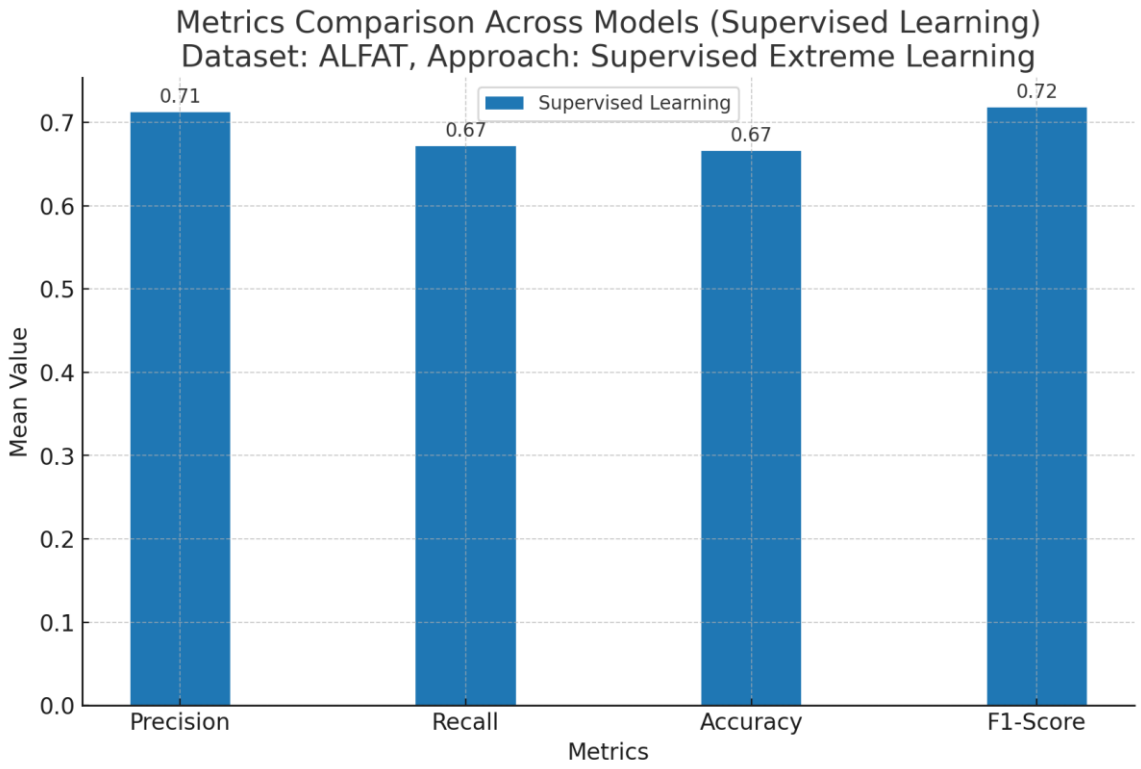
[32] M. Memarzadeh, B. Matthews, and T. Templin, “Multiclass Anomaly Detection in Flight Data Using Semi-Supervised Explainable Deep Learning Model,” *Journal of Aerospace Information Systems*, vol. 19, no. 2, pp. 83–97, Feb. 2022, doi: 10.2514/1.I010959.
[33] M. Memarzadeh, A. Akbari Asanjan, and B. Matthews, “Robust and Explainable Semi-Supervised Deep Learning Model for Anomaly Detection in Aviation,” *Aerospace*, vol. 9, no. 8, Aug. 2022, doi: 10.3390/aerospace9080437.
[34] S. Shilpi and S. Aryan, “Anomaly Detection in Time Series Flight Parameter Data Using Machine Learning Approach,” *Int J Res Appl Sci Eng Technol*, vol. 11, no. 9, pp. 824–832, Sep. 2023, doi: 10.22214/ijraset.2023.55763.
[35] S. J. Corrado *et al.*, “Deep Autoencoder for Anomaly Detection in Terminal Airspace Operations,” in *AIAA Aviation and Aeronautics Forum and Exposition, AIAA AVIATION Forum 2021*, American Institute of Aeronautics and Astronautics Inc, AIAA, 2021. doi: 10.2514/6.2021-2405.
[36] Y. Luo, Y. Xiao, L. Cheng, G. Peng, and D. D. Yao, “Deep Learning-based Anomaly Detection in Cyber-physical Systems: Progress and Opportunities,” *ACM Computing Surveys*, vol. 54, no. 5. 2021. doi: 10.1145/3453155.

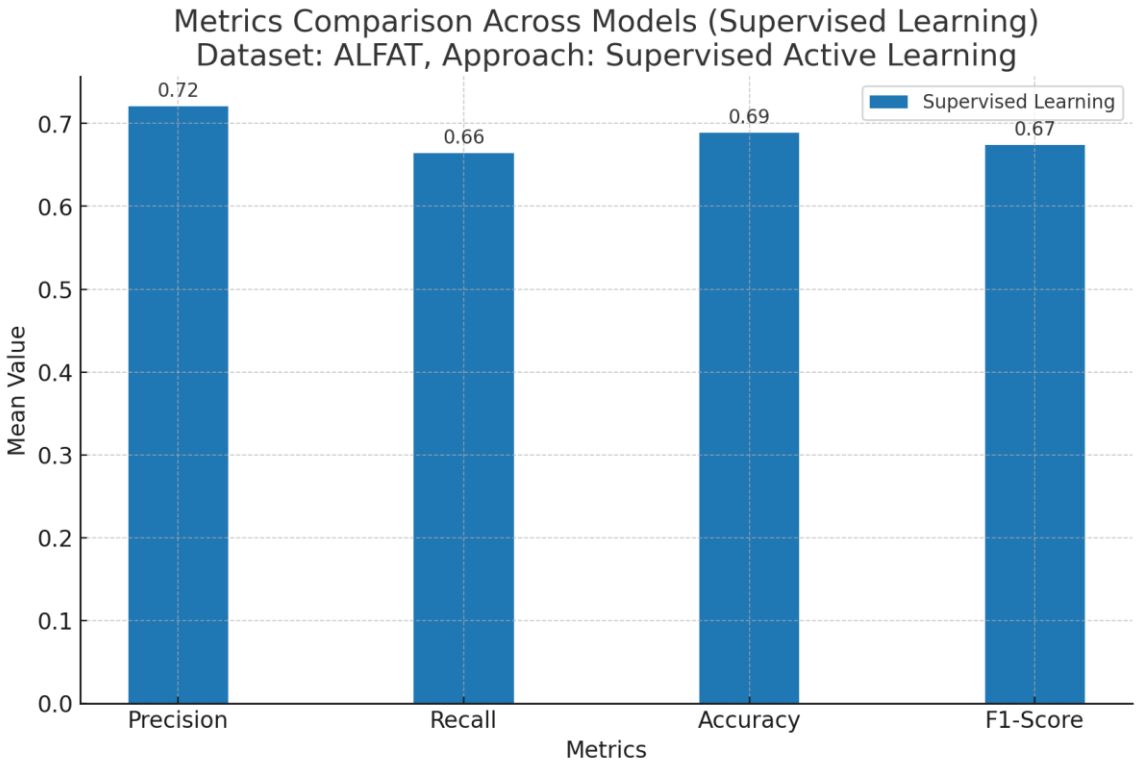
Appendix A

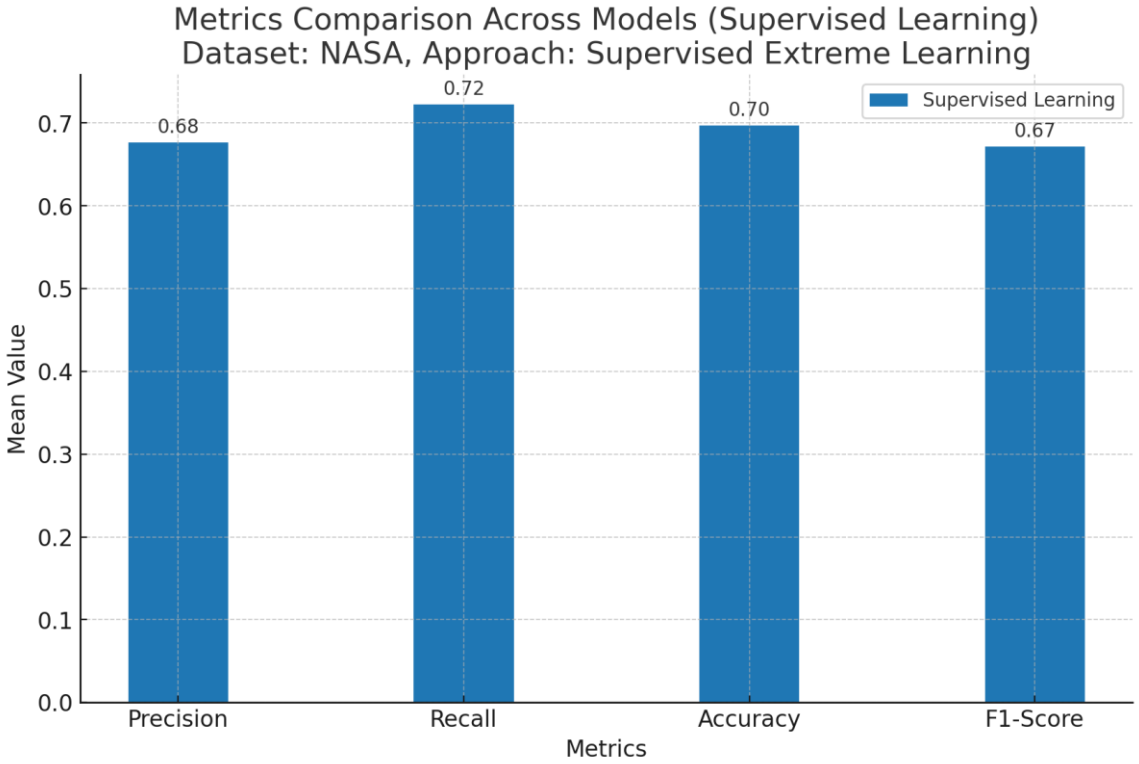
Experimental Evaluation Results

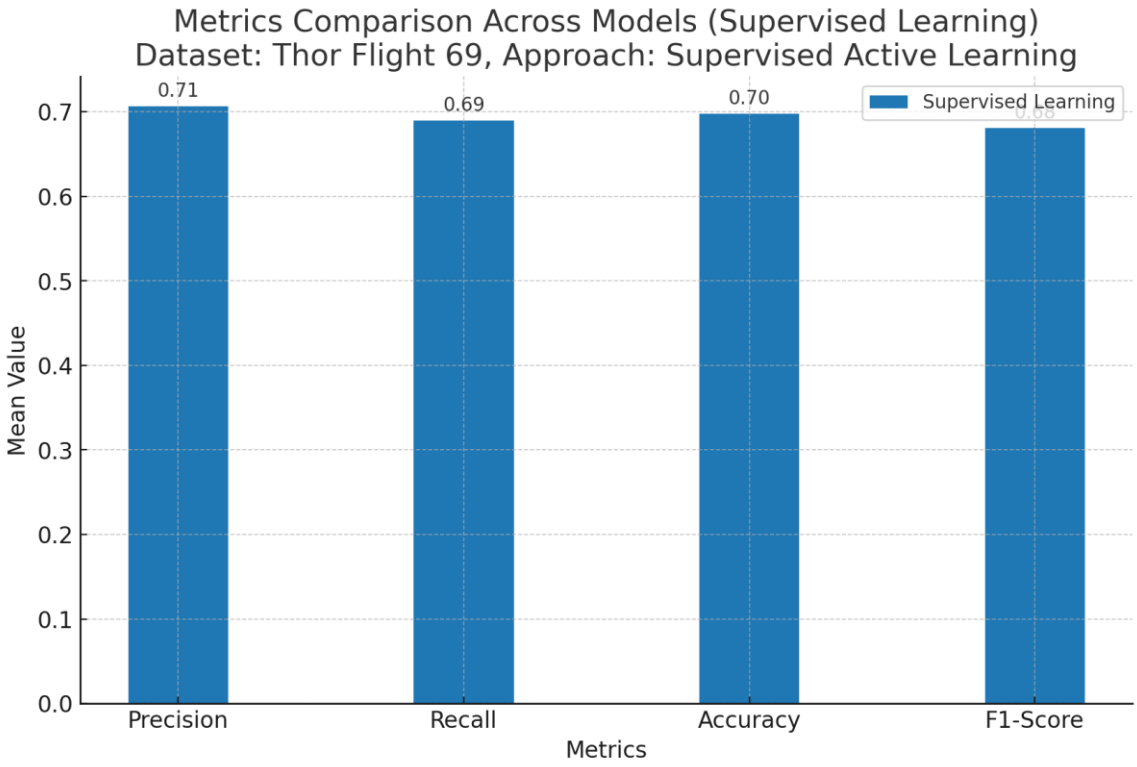
The figure contains the result of the experiment for each approach run on selected datasets, measuring the performance of approaches using different metrics.

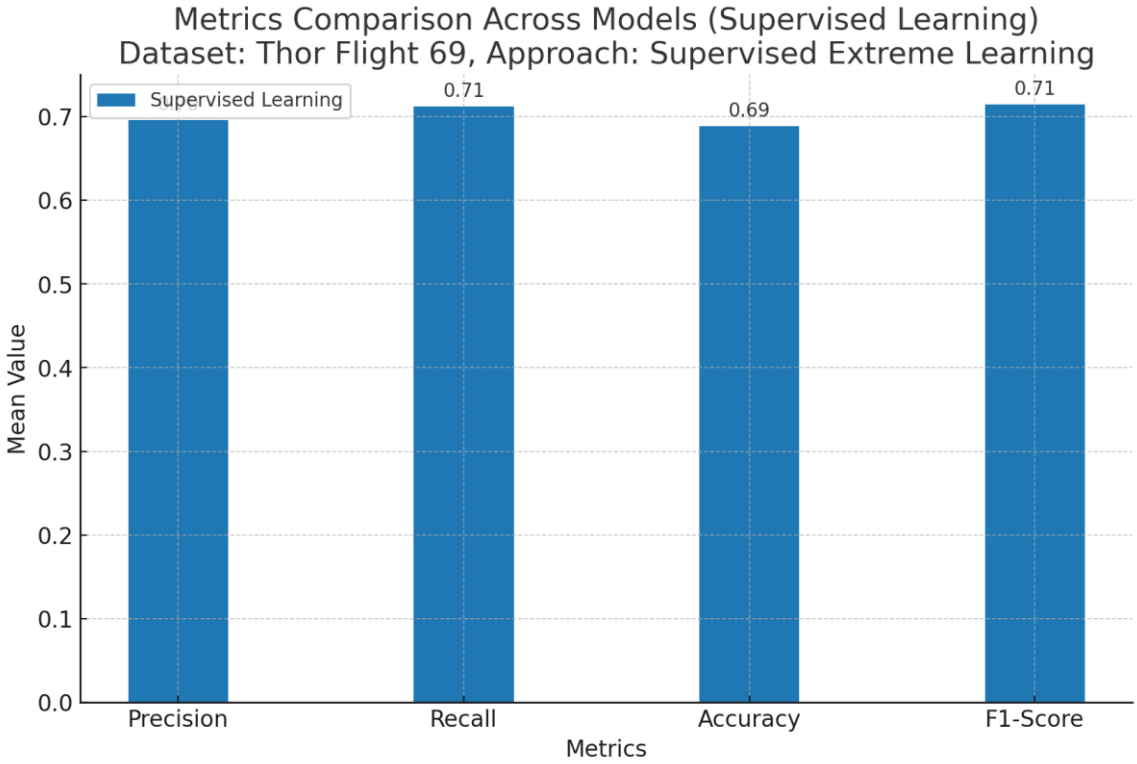
A.1 Supervised Learning Approaches

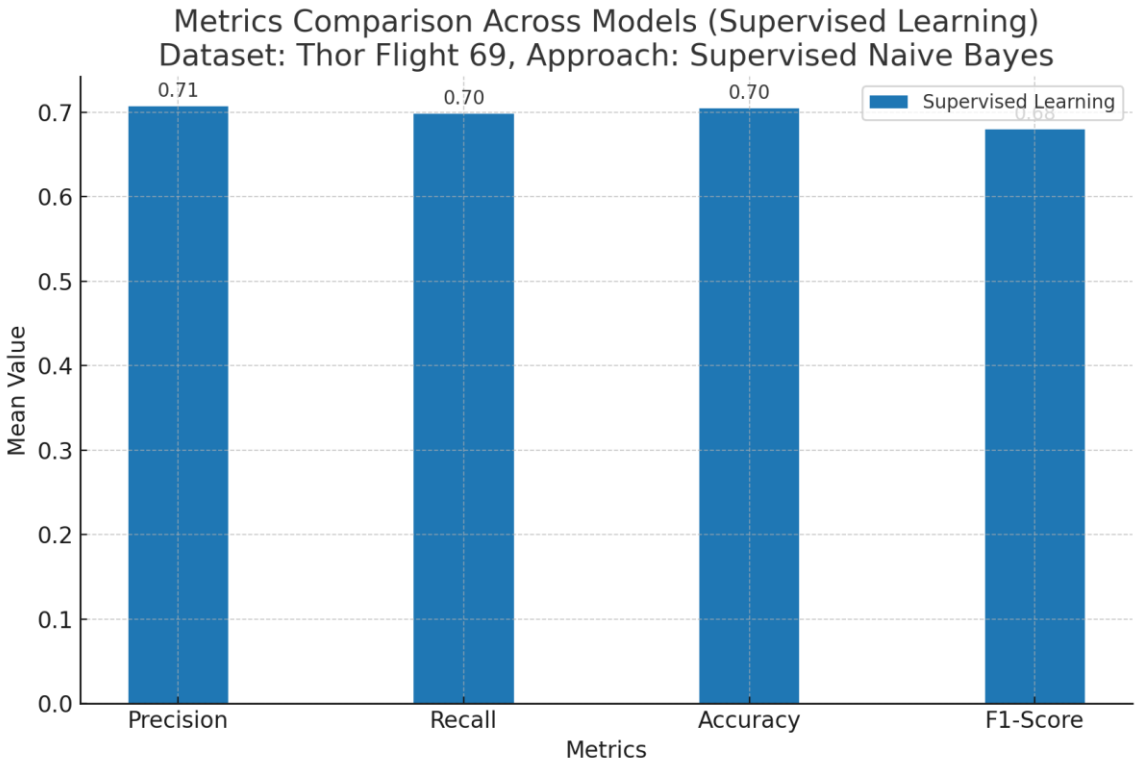


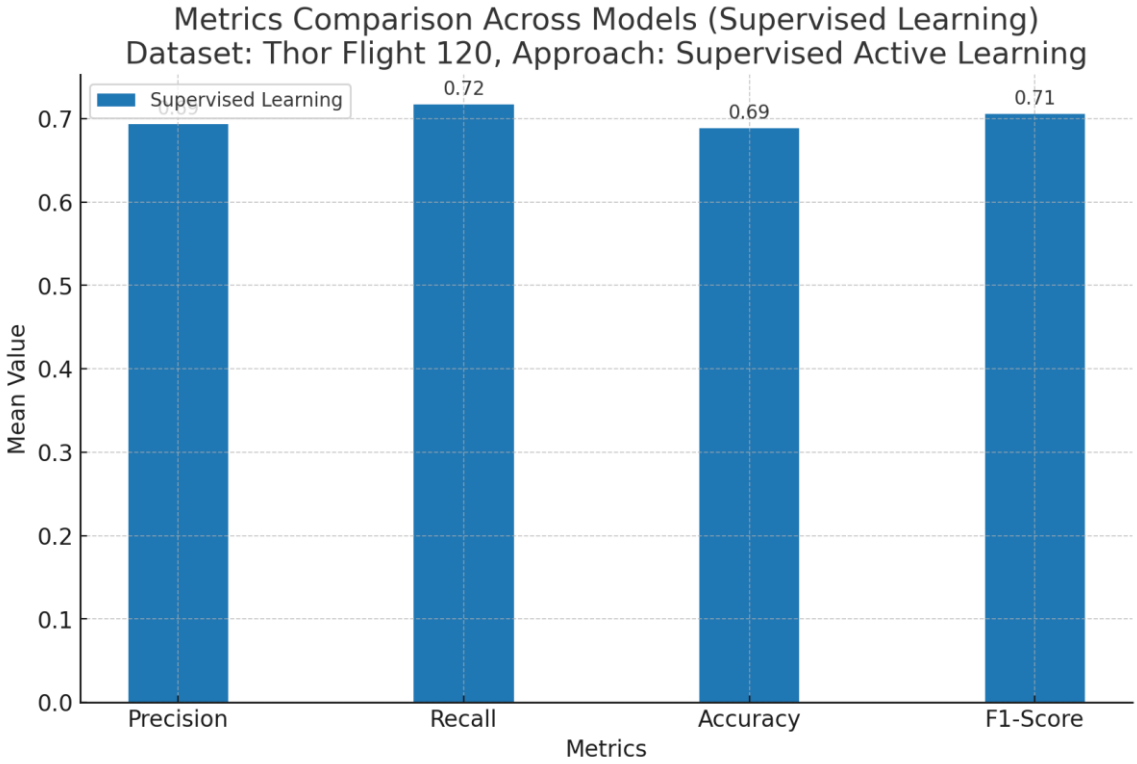

Metrics Comparison Across Models (Supervised Learning)
Dataset: ALFAT, Approach: Supervised Active Learning
Supervised Learning
0.72
0.66
0.69
0.67
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics

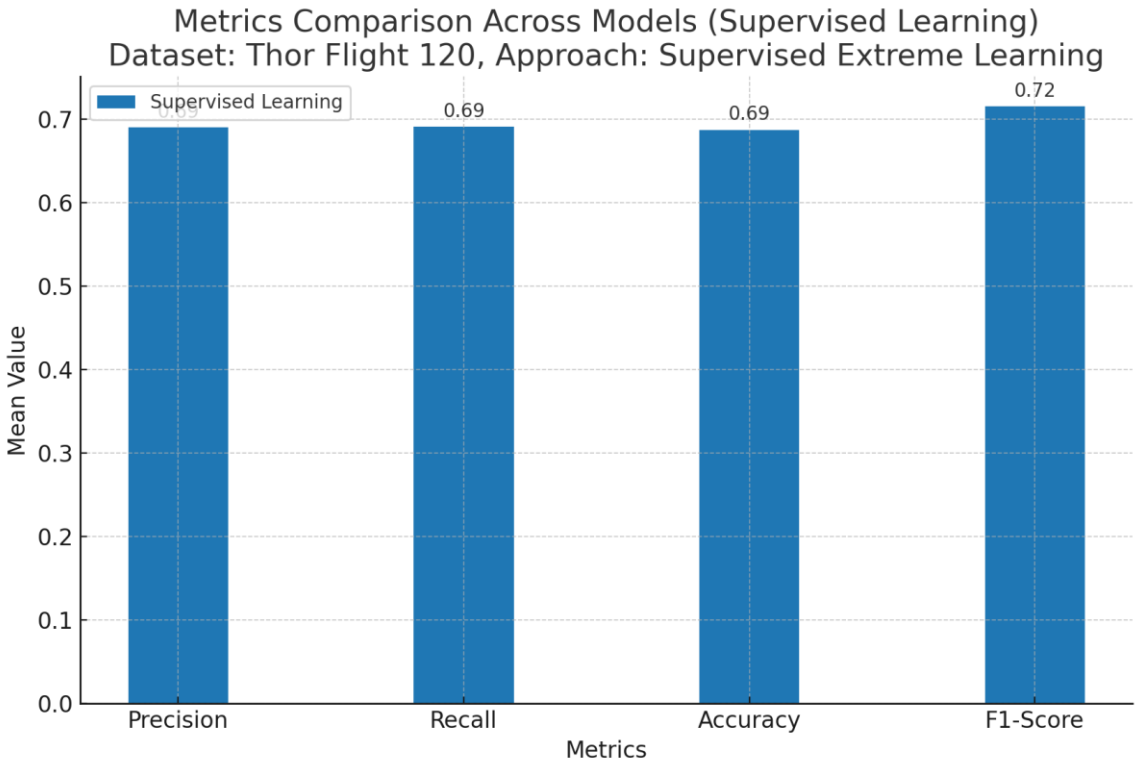

Metrics Comparison Across Models (Supervised Learning)
Dataset: NASA, Approach: Supervised Extreme Learning
Supervised Learning
0.68
0.72
0.70
0.67
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics

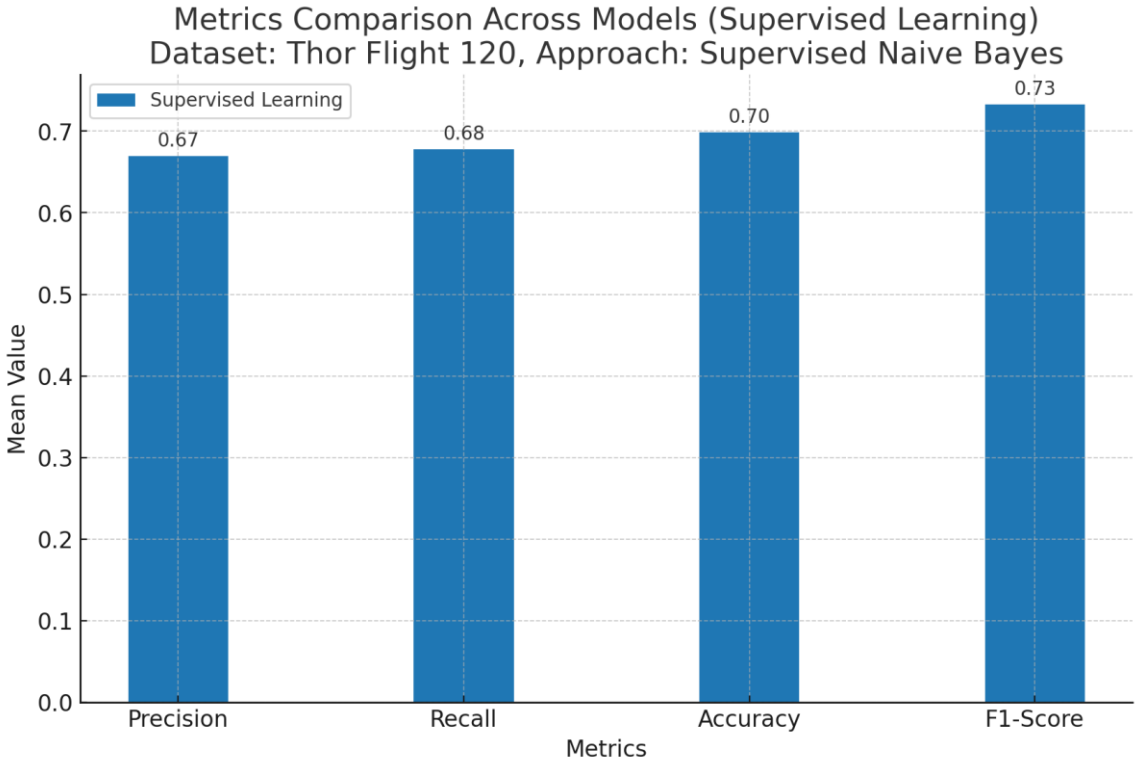

Metrics Comparison Across Models (Supervised Learning)
Dataset: Thor Flight 69, Approach: Supervised Active Learning
Supervised Learning
0.71
0.69
0.70
0.68
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics


Metrics Comparison Across Models (Supervised Learning)
Dataset: Thor Flight 69, Approach: Supervised Extreme Learning
Supervised Learning
0.71
0.69
0.71
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics

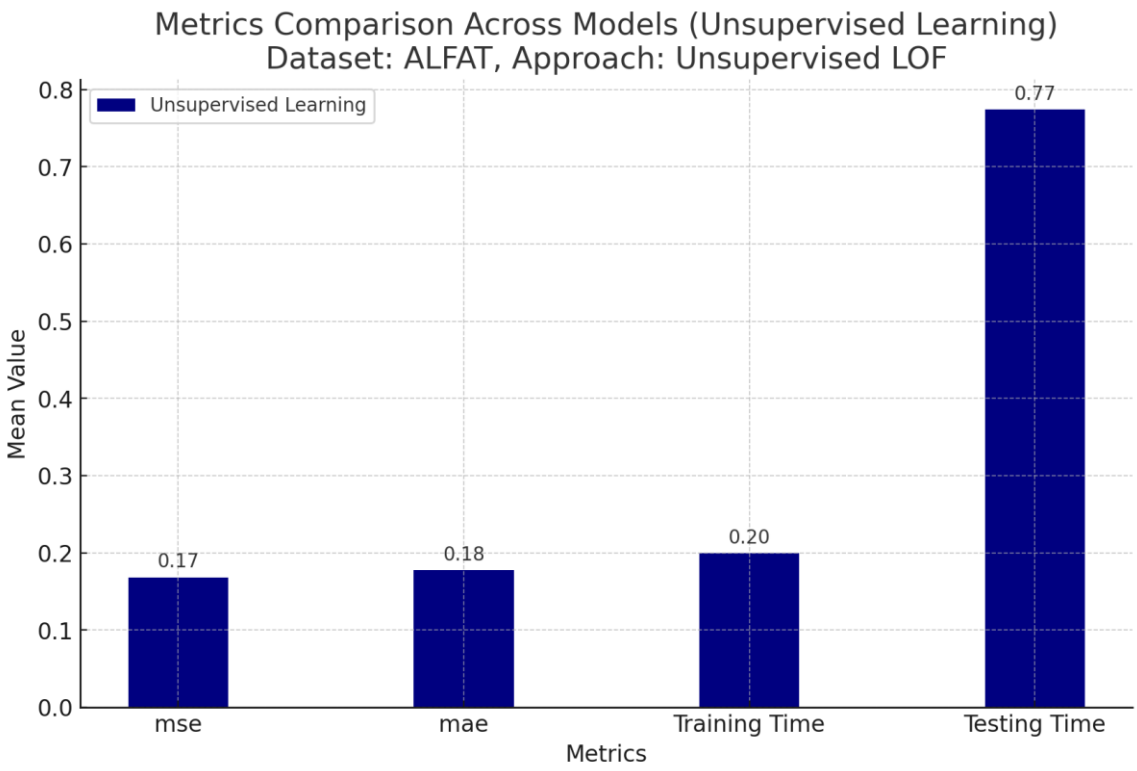

Metrics Comparison Across Models (Supervised Learning)
Dataset: Thor Flight 69, Approach: Supervised Naive Bayes
Supervised Learning
0.71
0.70
0.70
0.68
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics

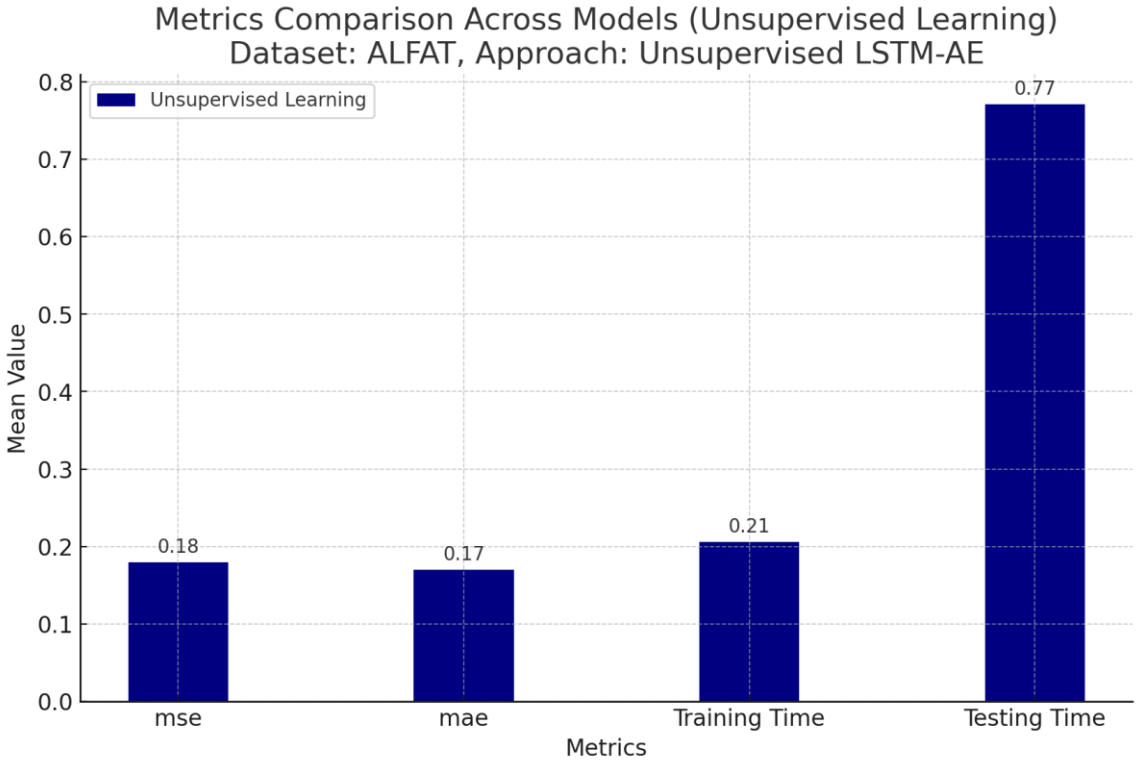

Metrics Comparison Across Models (Supervised Learning)
Dataset: Thor Flight 120, Approach: Supervised Active Learning
Supervised Learning
0.72
0.69
0.71
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics

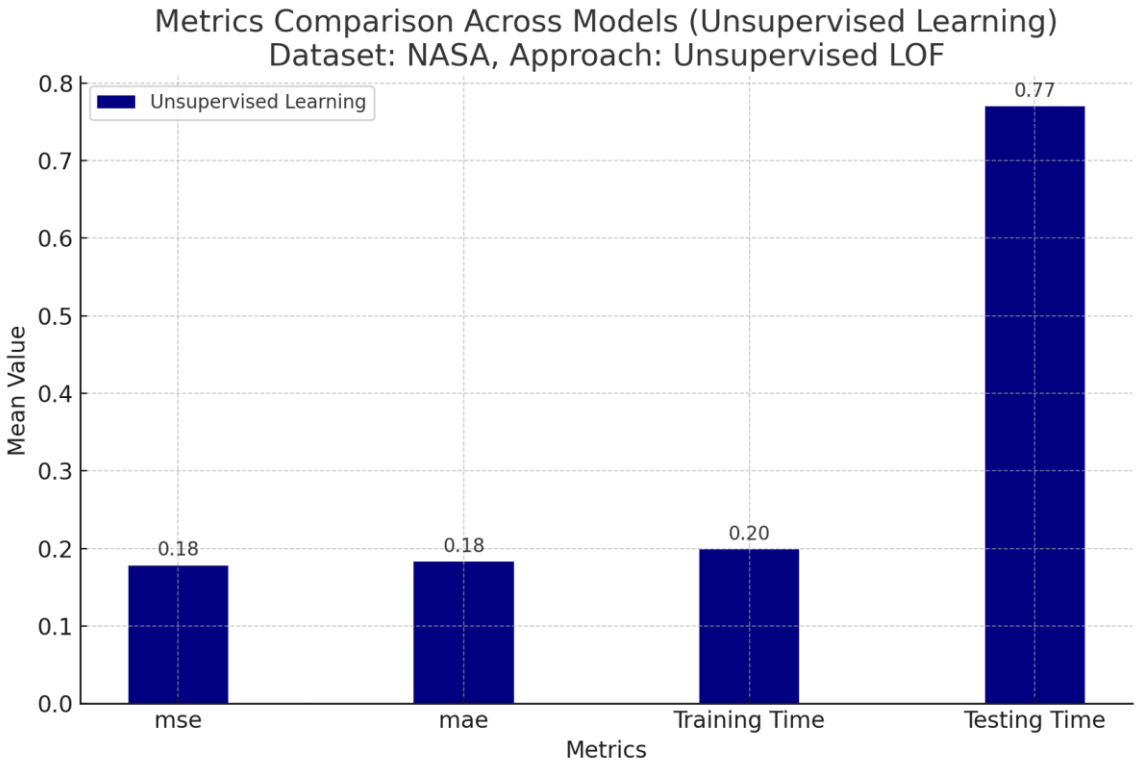

Metrics Comparison Across Models (Supervised Learning)
Dataset: Thor Flight 120, Approach: Supervised Extreme Learning
Supervised Learning
0.69
0.69
0.72
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics

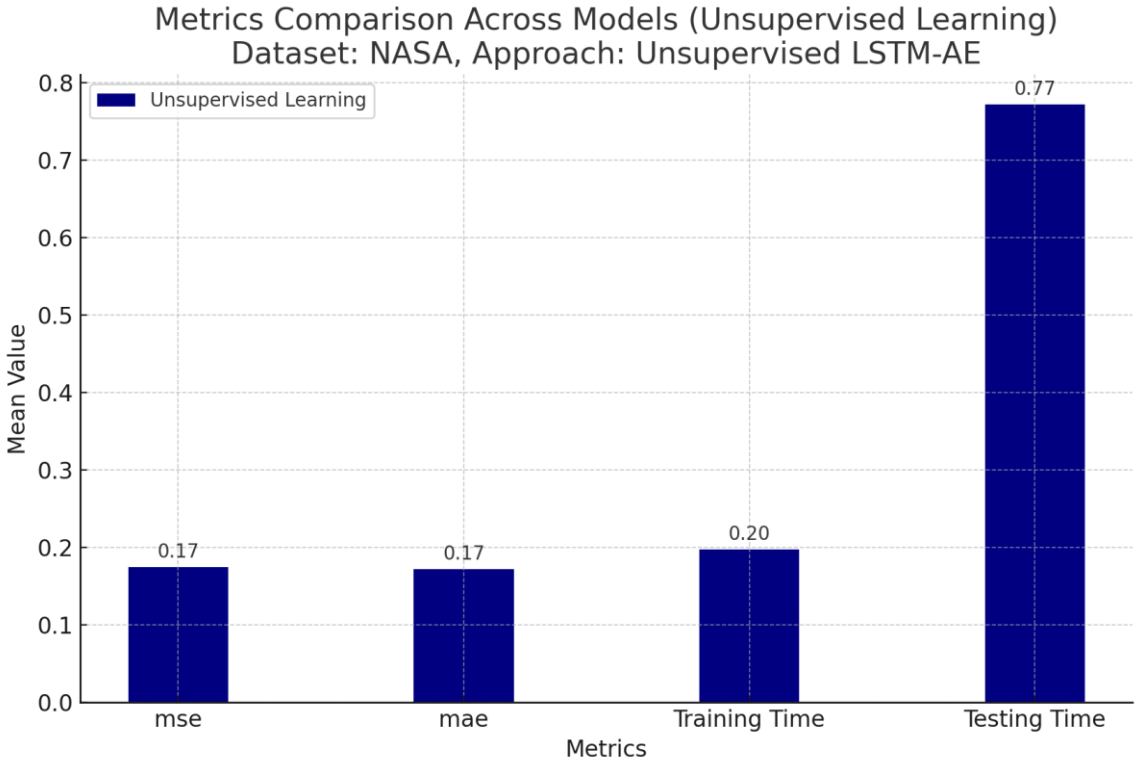

Metrics Comparison Across Models (Supervised Learning)
Dataset: Thor Flight 120, Approach: Supervised Naive Bayes
Supervised Learning
0.67
0.68
0.70
0.73
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Precision
Recall
Accuracy
F1-Score
Metrics

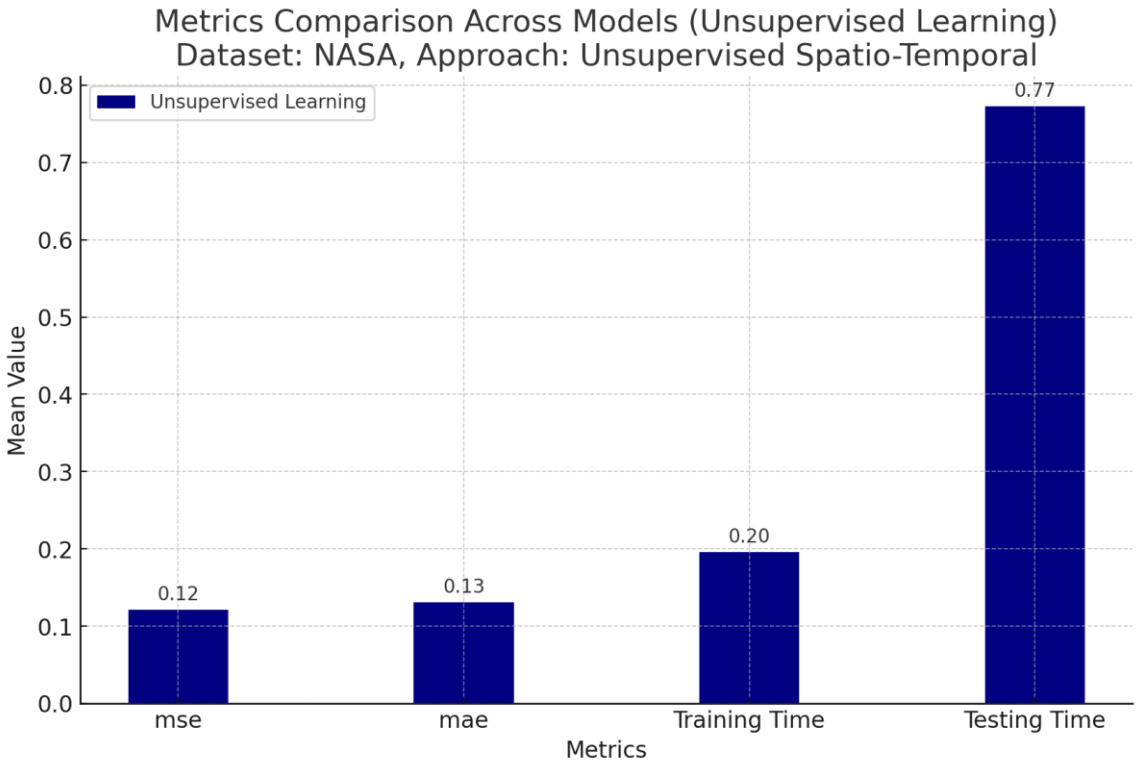
A.2 Unupervised Learning Approaches

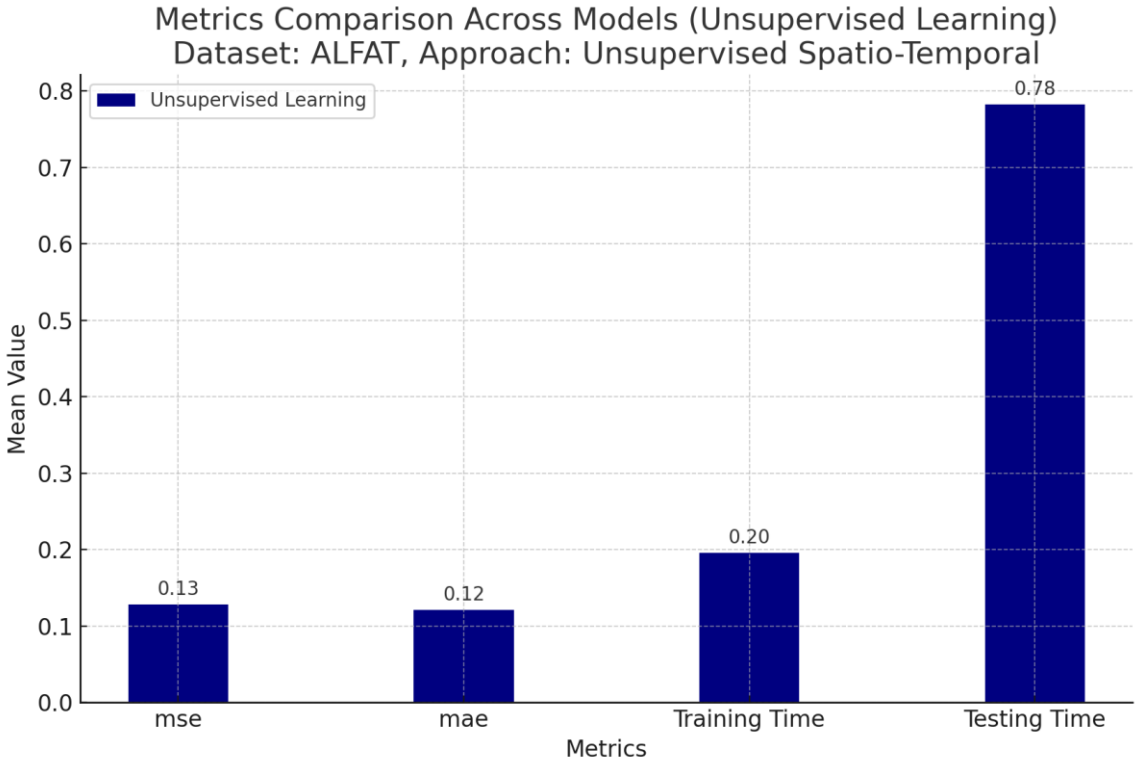


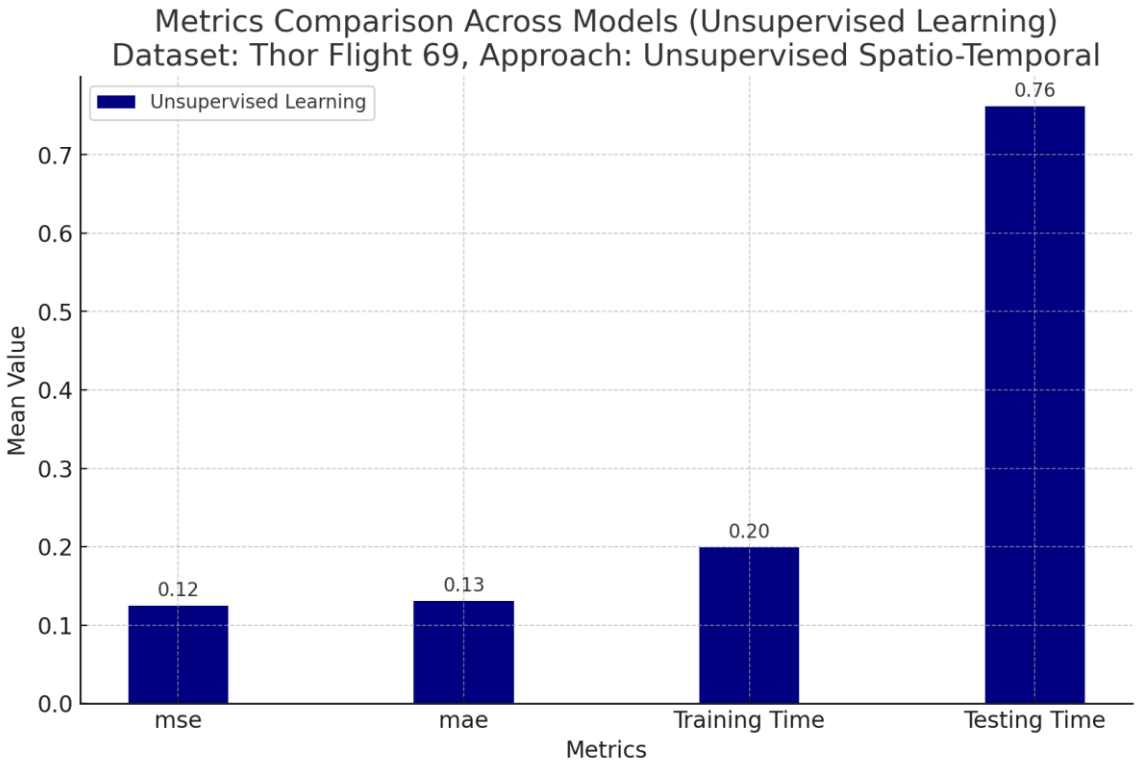


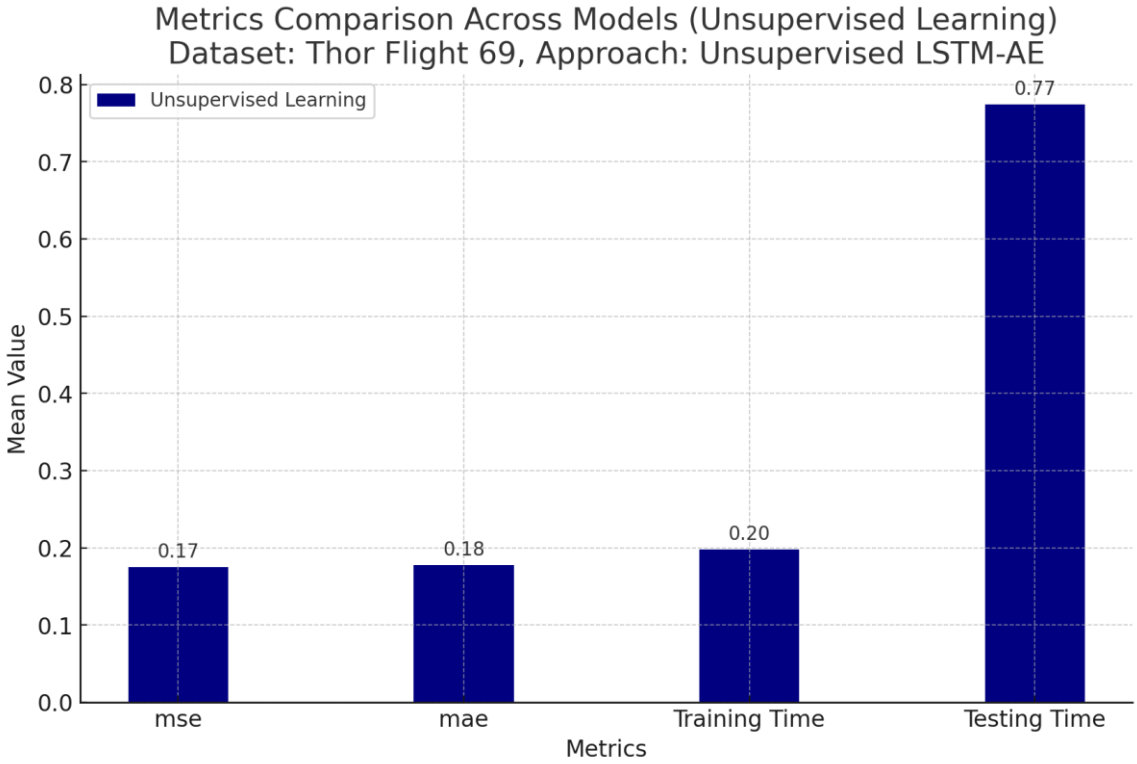

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: NASA, Approach: Unsupervised LOF
Unsupervised Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.18
0.18
0.20
0.77
mse
mae
Training Time
Testing Time
Metrics

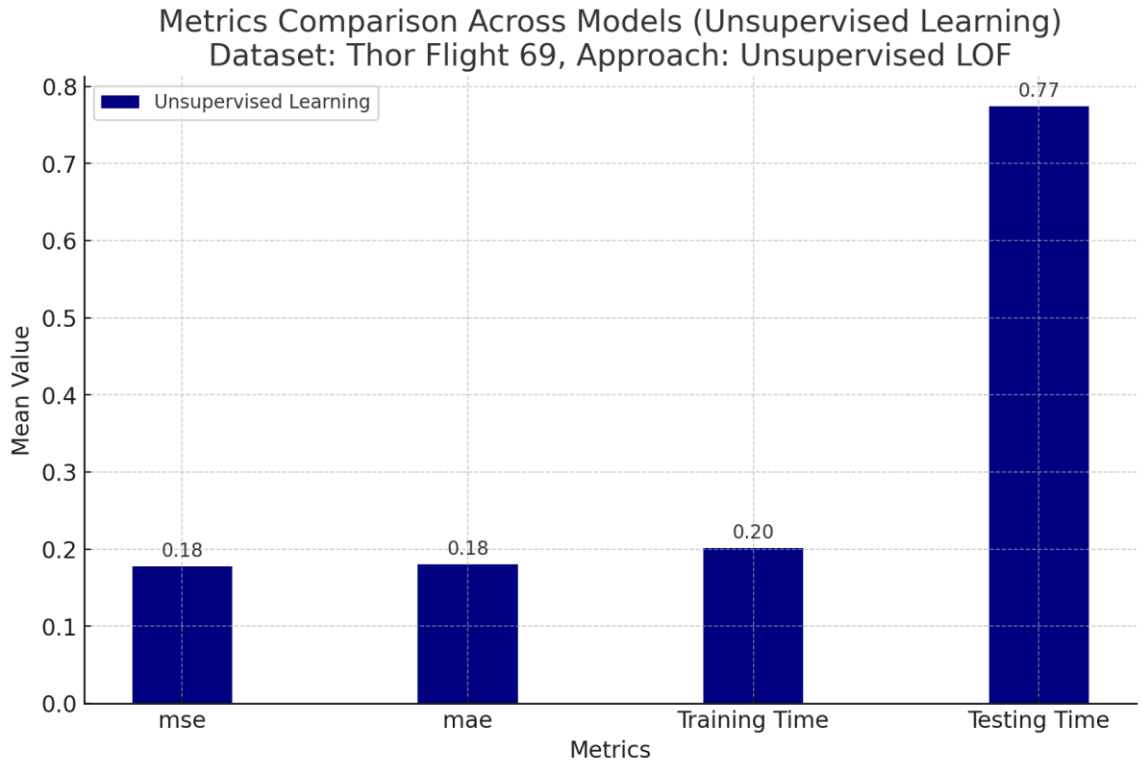

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: NASA, Approach: Unsupervised LSTM-AE
Unsupervised Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.17
0.17
0.20
0.77
mse
mae
Training Time
Testing Time
Metrics


Metrics Comparison Across Models (Unsupervised Learning)
Dataset: NASA, Approach: Unsupervised Spatio-Temporal
Unsupervised Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.12
0.13
0.20
0.77
mse
mae
Training Time
Testing Time
Metrics

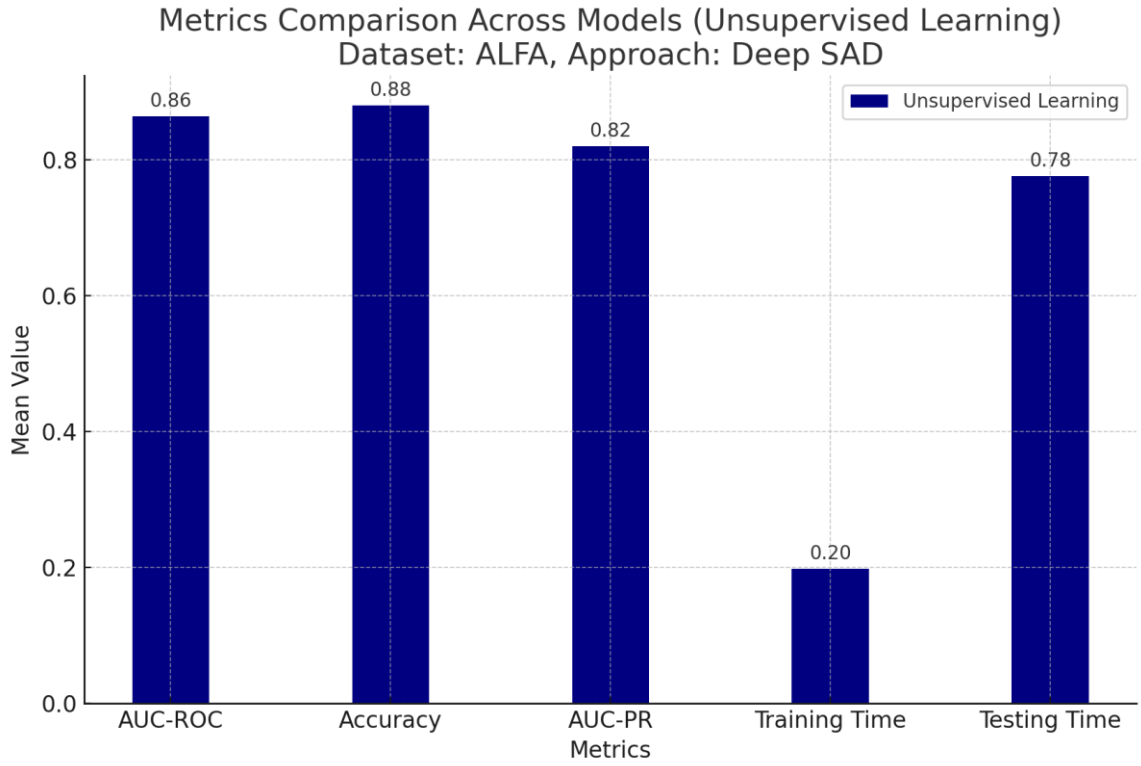

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: ALFAT, Approach: Unsupervised Spatio-Temporal
Unsupervised Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.13
0.12
0.20
0.78
mse
mae
Training Time
Testing Time
Metrics

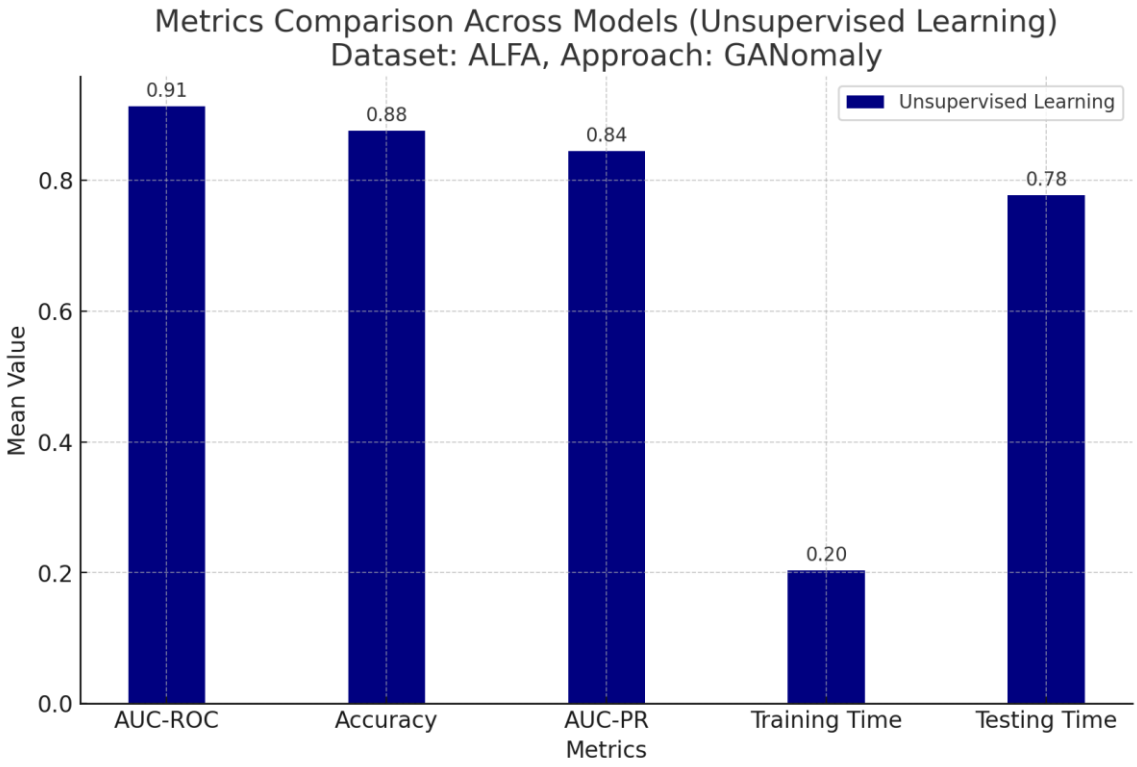

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: Thor Flight 69, Approach: Unsupervised Spatio-Temporal
Unsupervised Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.12
0.13
0.20
0.76
mse
mae
Training Time
Testing Time
Metrics

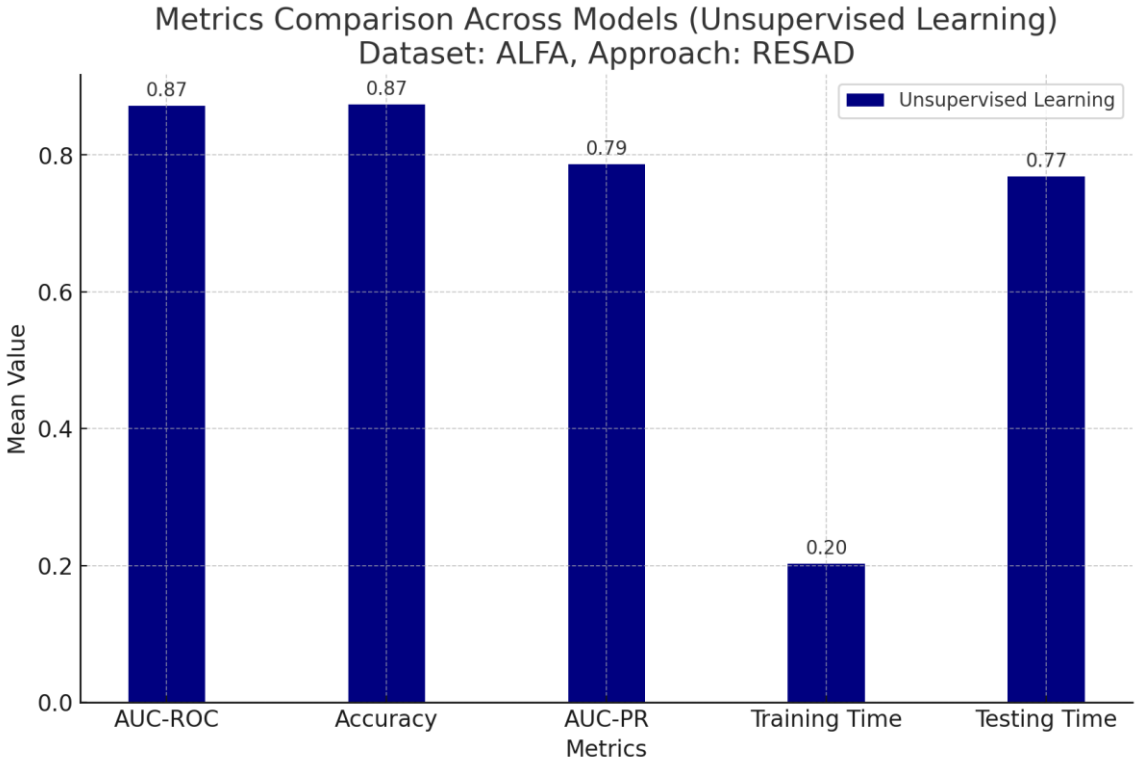

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: Thor Flight 69, Approach: Unsupervised LSTM-AE
Unsupervised Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.17
0.18
0.20
0.77
mse
mae
Training Time
Testing Time
Metrics

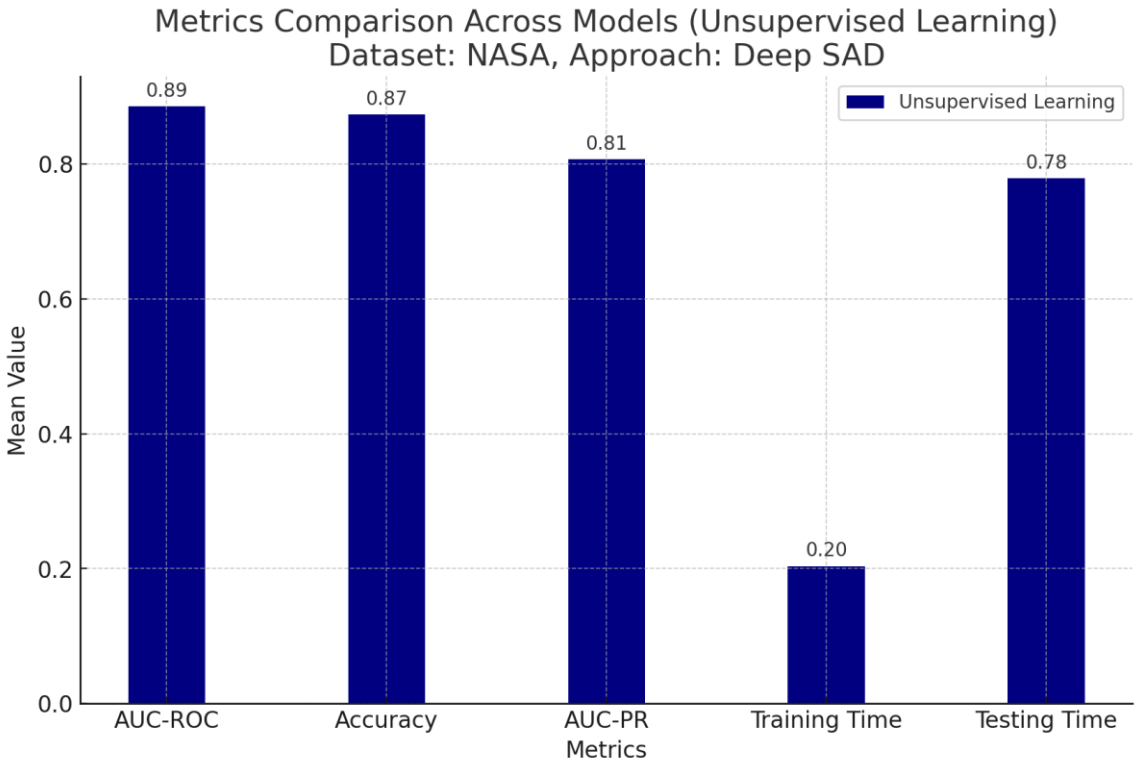


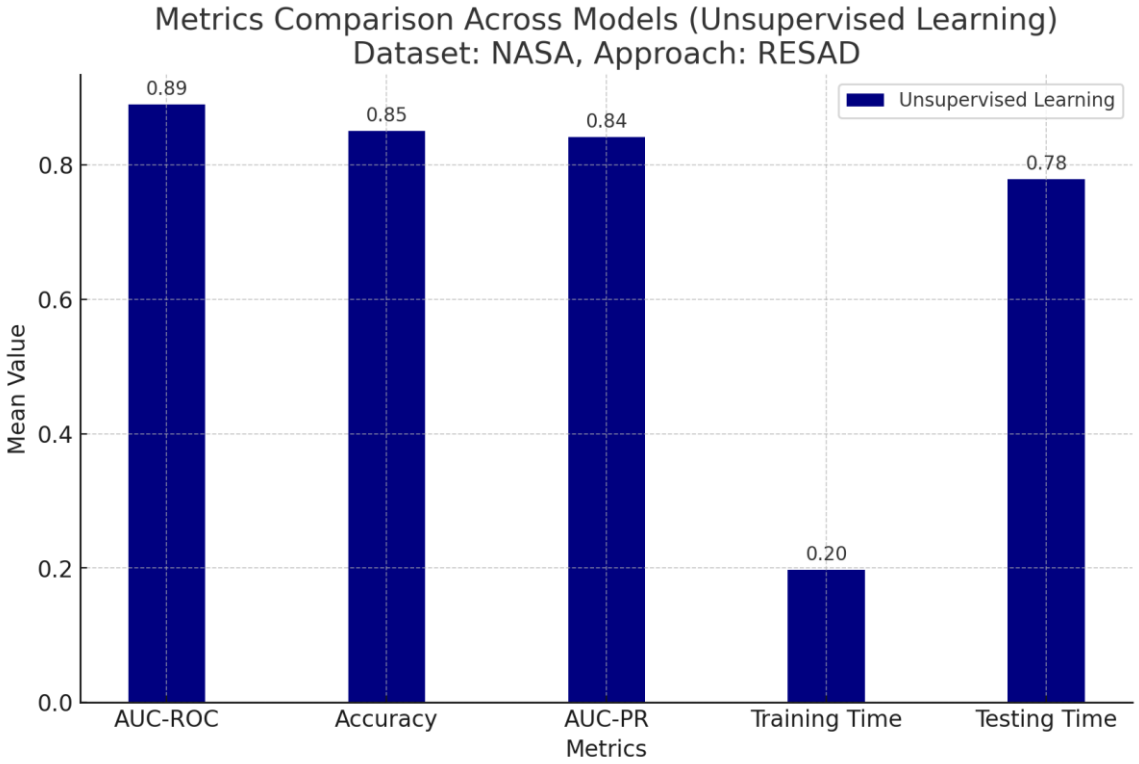
A.1 Semi-Supervised Learning Approaches

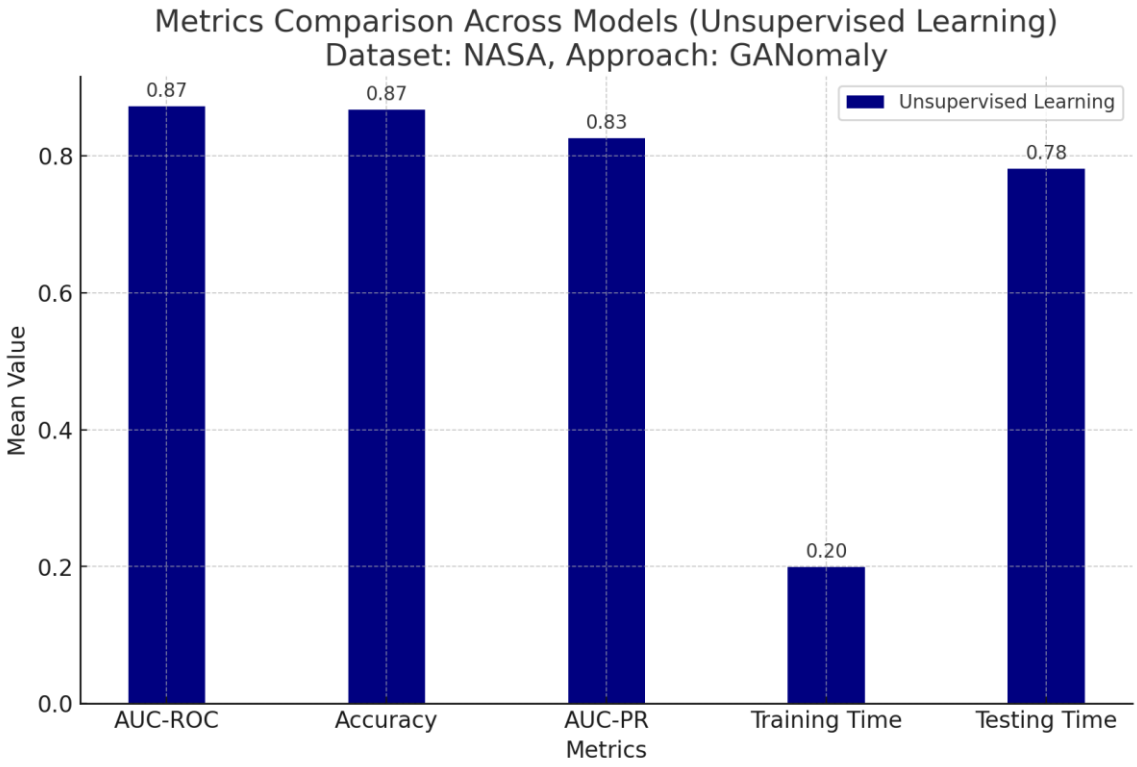


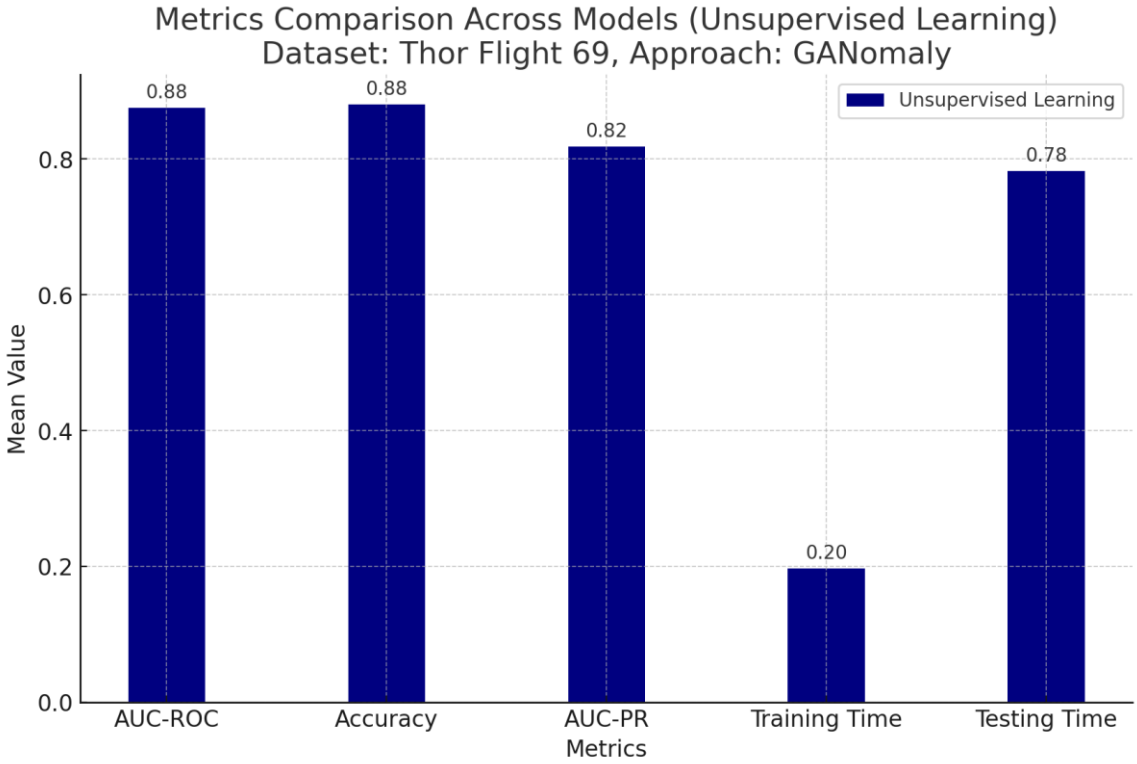

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: ALFA, Approach: GANomaly
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.91
0.88
0.84
0.20
0.78
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics

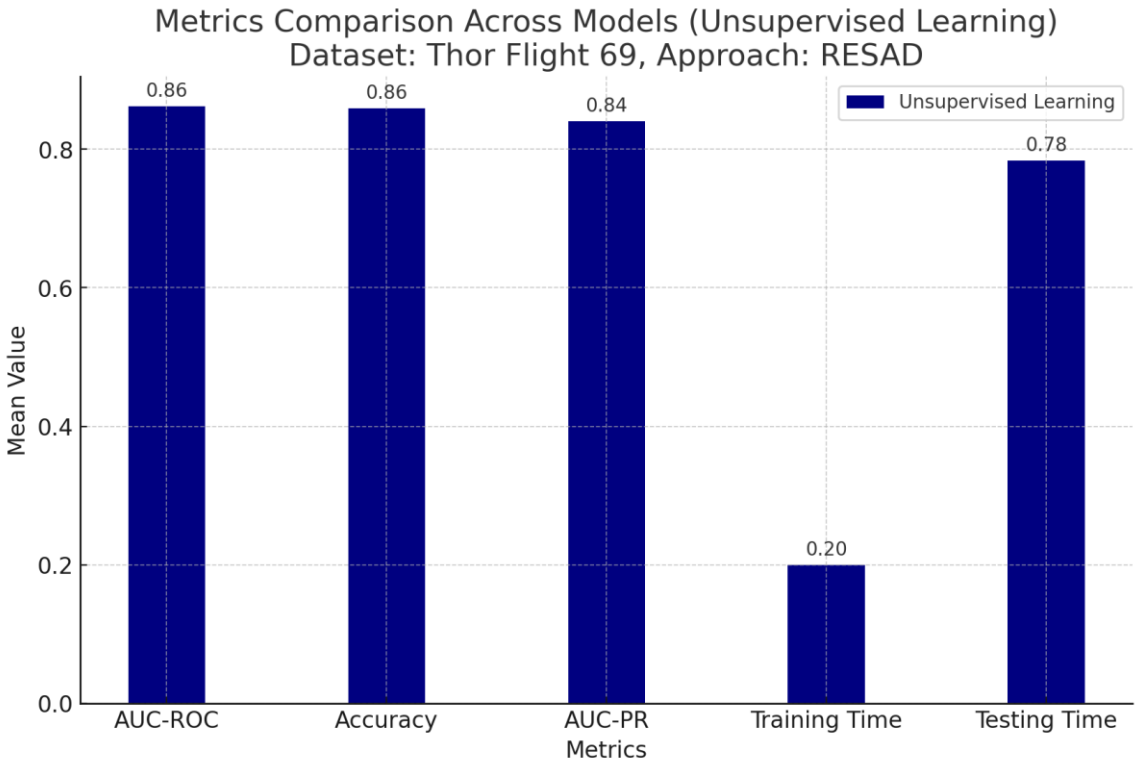

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: ALFA, Approach: RESAD
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.87
0.87
0.79
0.20
0.77
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics

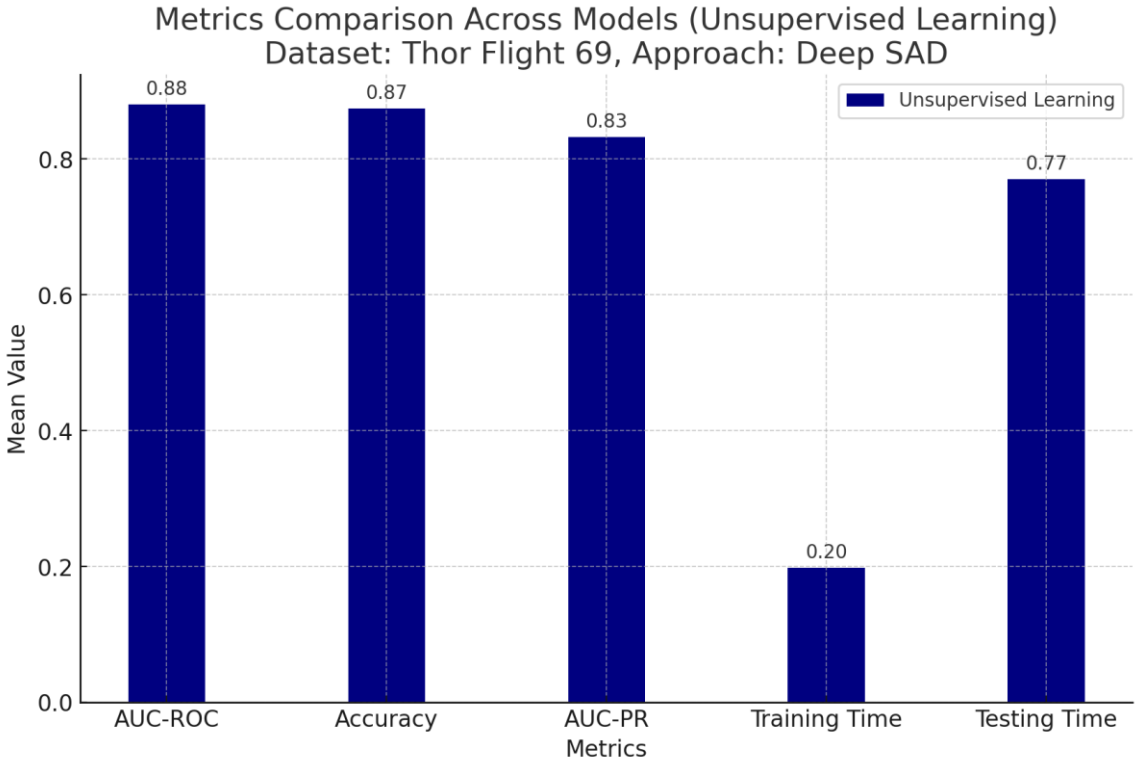

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: NASA, Approach: Deep SAD
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.89
0.87
0.81
0.20
0.78
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics


Metrics Comparison Across Models (Unsupervised Learning)
Dataset: NASA, Approach: RESAD
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.89
0.85
0.84
0.20
0.78
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics

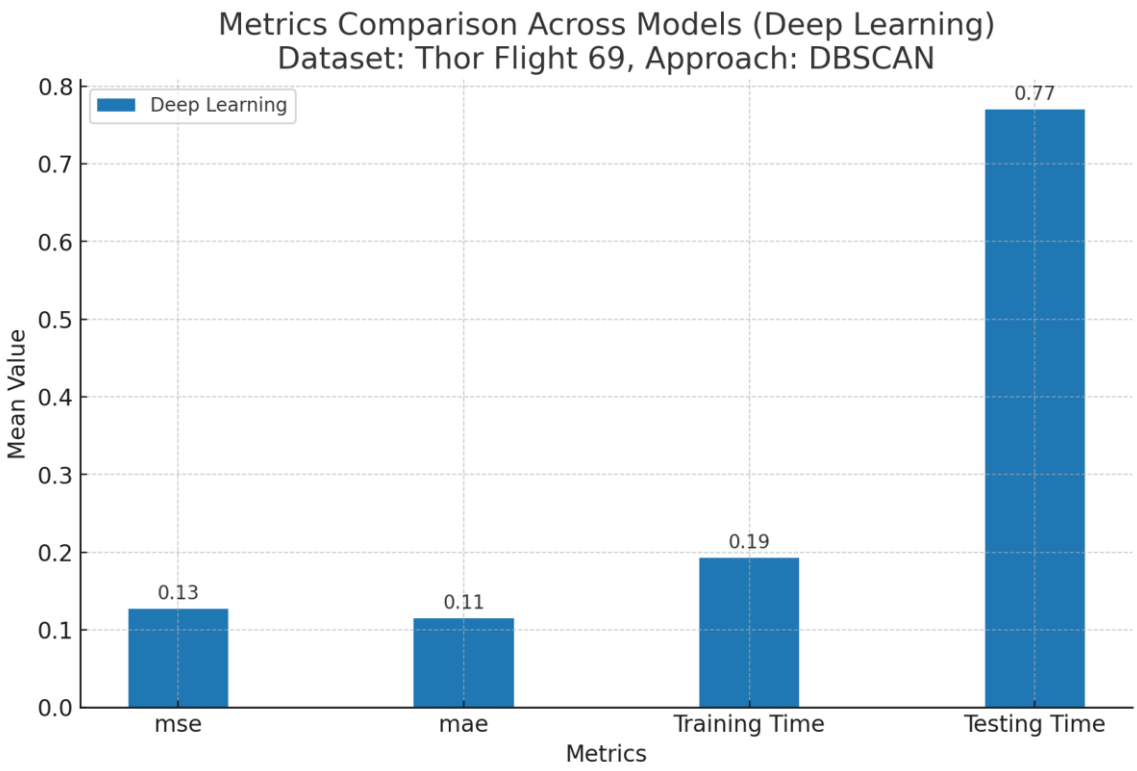

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: NASA, Approach: GANomaly
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.87
0.87
0.83
0.20
0.78
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics

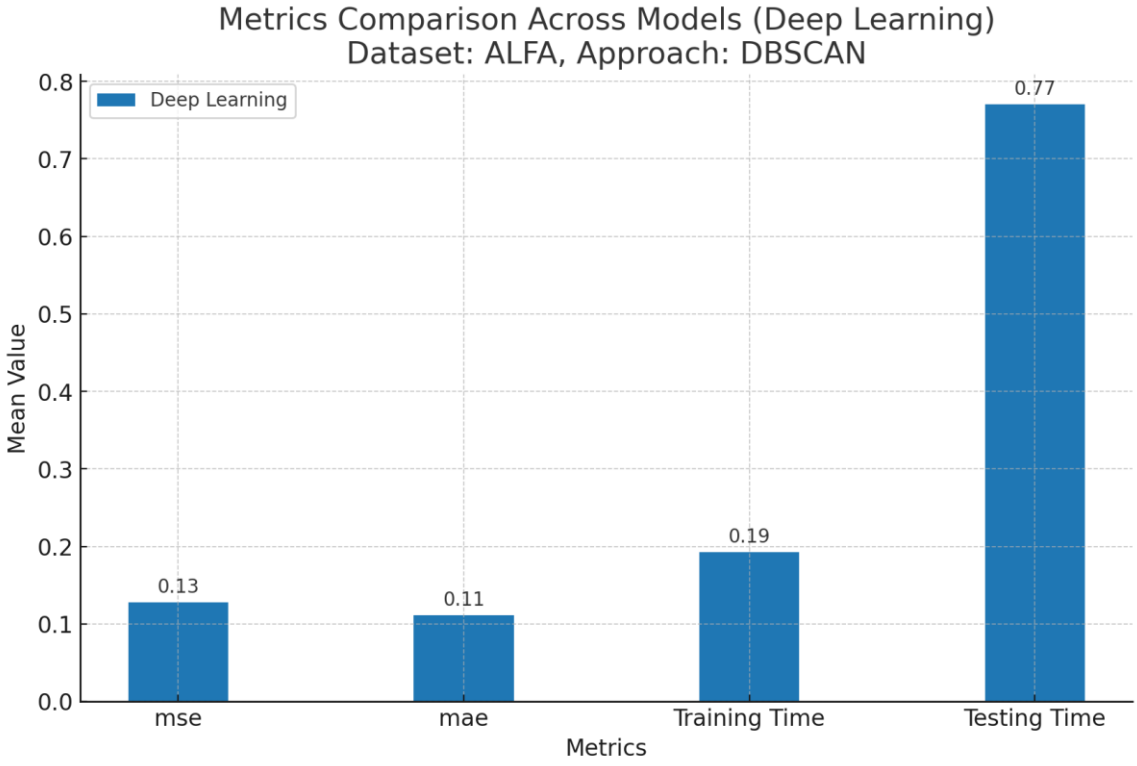

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: Thor Flight 69, Approach: GANomaly
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.88
0.88
0.82
0.20
0.78
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics

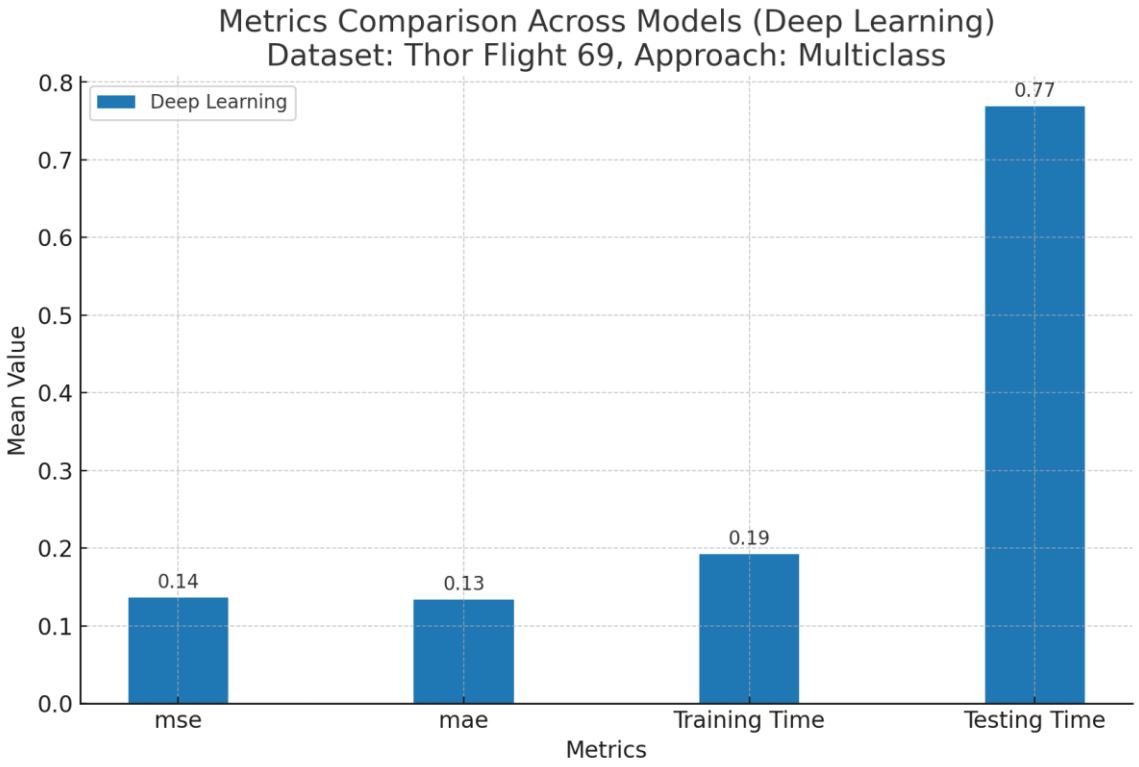

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: Thor Flight 69, Approach: RESAD
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.86
0.86
0.84
0.20
0.78
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics

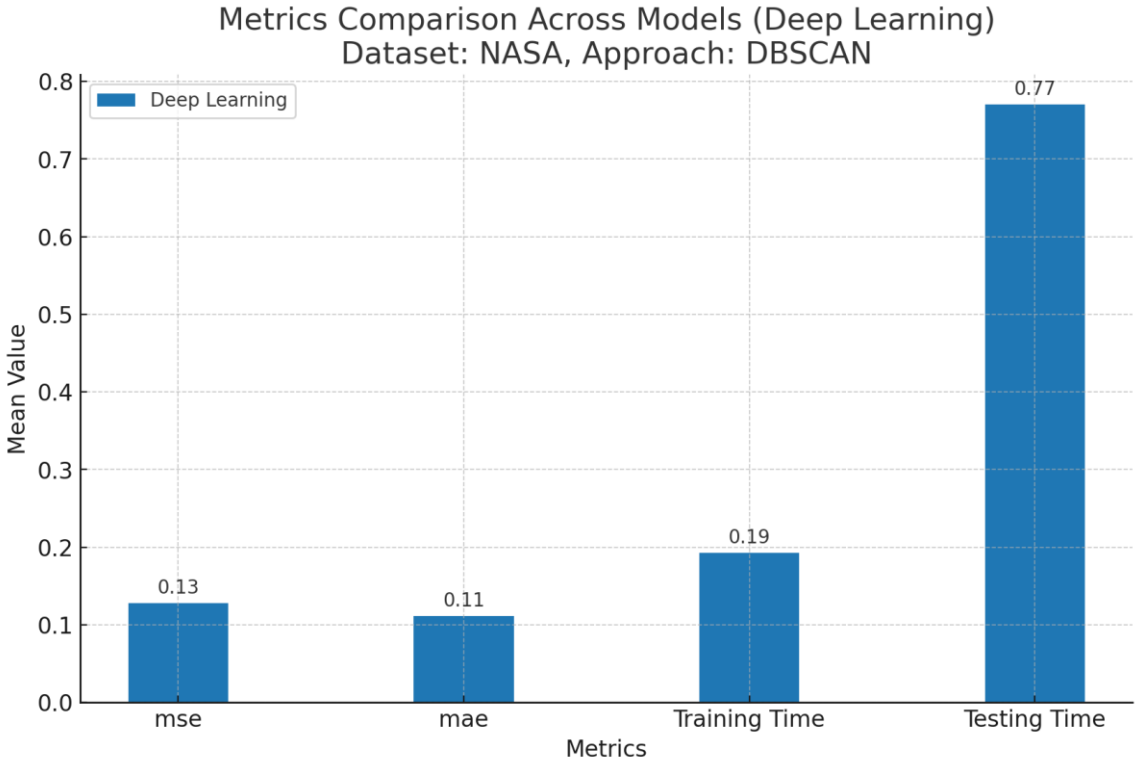

Metrics Comparison Across Models (Unsupervised Learning)
Dataset: Thor Flight 69, Approach: Deep SAD
Unsupervised Learning
Mean Value
0.0
0.2
0.4
0.6
0.8
0.88
0.87
0.83
0.20
0.77
AUC-ROC
Accuracy
AUC-PR
Training Time
Testing Time
Metrics

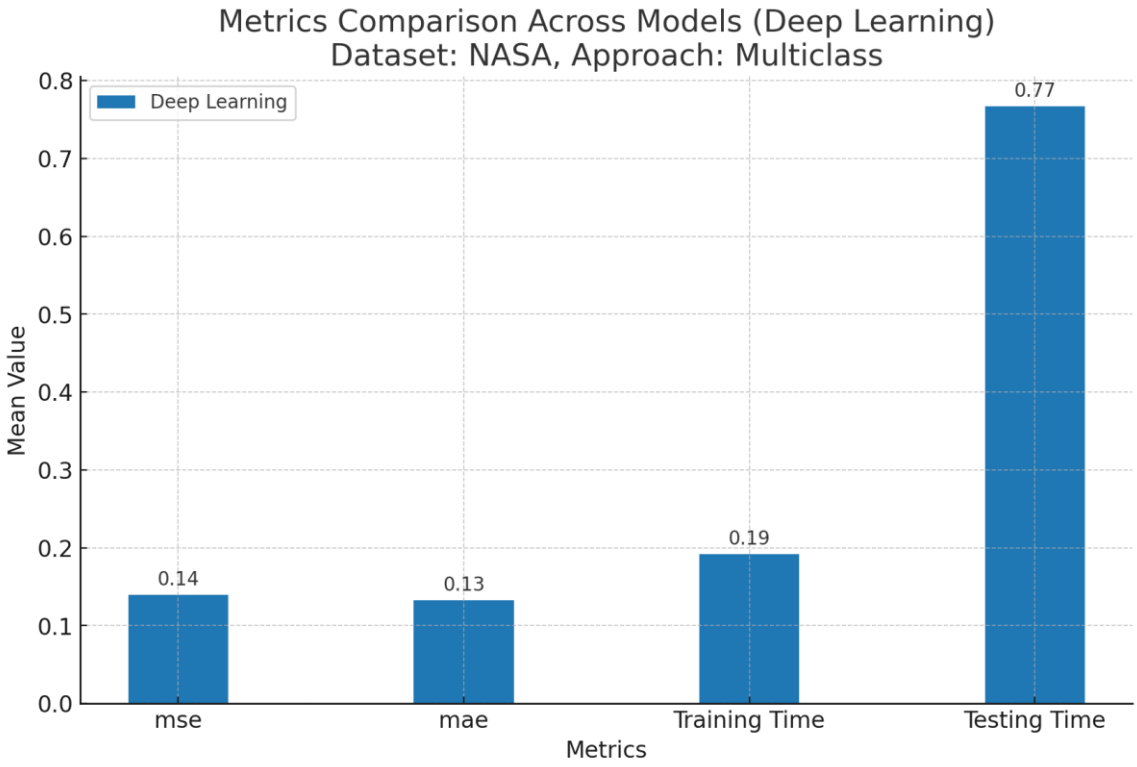
A.3 Deep Learning Approaches






Metrics Comparison Across Models (Deep Learning)
Dataset: Thor Flight 69, Approach: Multiclass
Deep Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.14
0.13
0.19
0.77
mse
mae
Training Time
Testing Time
Metrics


Metrics Comparison Across Models (Deep Learning)
Dataset: NASA, Approach: DBSCAN
Deep Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.13
0.11
0.19
0.77
mse
mae
Training Time
Testing Time
Metrics


Metrics Comparison Across Models (Deep Learning)
Dataset: NASA, Approach: Multiclass
Deep Learning
Mean Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.14
0.13
0.19
0.77
mse
mae
Training Time
Testing Time
Metrics