Independent University **Bangladesh (IUB)**

IUB Academic Repository

Computer Science and Engineering | Undergraduate Thesis

2026-04

Anomaly Detection in Petroleum Production Wells Using Machine Learning: A Comparative Study on the Petrobras 3W Dataset

Rahman, Sayed Rakinur

IUB

https://ar.iub.edu.bd/handle/11348/1198

Downloaded from IUB Academic Repository



Anomaly Detection in Petroleum Production Wells Using Machine Learning: A Comparative Study on the Petrobras 3W Dataset

April 2026

Prepared by:

Sayed Rakinur Rahman
ID: 2110690

Amanta Suraiya
ID: 2110965

Md. Turan Rafsan
ID: 2111319

Department of Computer Science and Engineering
Independent University, Bangladesh

Supervised by

Ashraful Islam, PhD
Assistant Professor
Department of Computer Science and Engineering
Independent University, Bangladesh

Attestation

We are aware of the fact that plagiarism is strictly prohibited and it conflicts with our university's rules and regulations. We guarantee the authenticity of our work. We have used some copyrighted material and others' models in our project, which have been properly cited in accordance with international standards and guidelines.

Author Name: Sayed Rakinur Rahman

Signature: .....

Author Name: Amanta Suraiya

Signature: .....

Author Name: Md. Turan Rafsan

Signature: .....

Evaluation Committee

Supervision Panel

.....
Supervisor

Examiners

.....
Dr. internal (Internal)

.....
Mr. internal 2 (Internal)

.....
Mr. external (External Examiner)

Office Use

.....
Program Coordinator

.....
Head of the Department

.....
Director, Office of the Graduate Studies, Research and Industry Relations

.....
Library

Acknowledgement

We would like to express our sincere gratitude to our respected supervisor, Ashraful Islam, PhD, from Independent University, Bangladesh. His dedicated guidance, insightful suggestions, remarkable patience, and the generous investment of his valuable time played a pivotal role in the successful completion of our project. Our Professor's expertise and mentorship greatly enhanced the quality and depth of our work, and we are truly appreciative of his unwavering support throughout our academic journey.

Additionally, our heartfelt thanks go to the Center for Computational & Data Sciences (CCDS) for their extraordinary help and consistent support. The collaborative and resourceful environment provided by CCDS significantly contributed to the development and execution of our project. Moreover, the encouragement and assistance have been instrumental in shaping our academic and research fields. We are grateful for the opportunities and resources extended to us during our research, which undoubtedly enriched our learning experience.

We acknowledge and appreciate the contributions of all individuals who supported us, both directly and indirectly, during this endeavor.

Abstract

Anomaly detection in offshore oil production systems is a safety-critical task, with undetected faults contributing directly to equipment failures, environmental incidents, and significant production losses. Machine Learning (ML) has shown significant potential for fault classification in industrial time-series data. However, few comparative studies systematically evaluate models across multiple learning paradigms, including classical, supervised deep learning, and unsupervised learning on real-world operational data. Existing studies tend to evaluate individual model architectures, offering limited analysis across fault types, class-imbalance conditions, and temporal data characteristics. Multi-class fault detection in oil wells presents multiple challenges, such as severe class imbalance, overlapping faults, and the risk of temporal data leakage as a result of splitting strategies applied to sensor time series. This report presents an extensive comparison study of machine learning approaches for multi-class anomaly detection using the Petrobras 3W dataset across eight fault types, a publicly available benchmark containing real-world sensor data from offshore oil wells. Five distinct models are evaluated: Random Forest (RF) as a classical machine-learning baseline, while Long Short-Term Memory (LSTM) and Transformer classifiers as supervised deep-learning approaches. An LSTM Autoencoder for unsupervised anomaly detection. This study proposes an Advanced Stacking Ensemble that combines all three supervised paradigms using cross-validated out-of-fold (OOF) prediction generation, meta-feature engineering including prediction entropy, inter-model agreement, confidence statistics and an eXtreme Gradient Boosting (XGBoost) meta-learner. A thorough preprocessing pipeline is implemented, including file-level data splitting to prevent temporal leakage, sliding-window segmentation, statistical feature engineering, and Synthetic Minority Over-sampling Technique (SMOTE) for class balancing. On the held-out test set, the Stacking Ensemble achieves the highest accuracy of 96.25%, while the RF has an accuracy of 95.98%. The RF also attains weighted F1-score of 0.9558. The Transformer (F1: 0.9295) and LSTM (F1: 0.9237) demonstrate competitive performance, with the Transformer offering the advantage of temporal modeling. The LSTM Autoencoder achieves an F1-score of 0.8588 for binary anomaly detection without requiring labeled anomaly data. Model interpretability is addressed through Shap-ley Additive exPlanations (SHAP) based feature-level importance analysis and Transformer attention-weight visualization for revealing fault-specific temporal interpretability attended to by the model.

Contents

1 Introduction **9**
- 1.1 Background of the Project . . . 9
- 1.2 Problem Statement . . . 9
- 1.3 Research Objectives . . . 10
- 1.4 Contributions . . . 10
- 1.5 Report Organization . . . 11

2 Related Works **12**
- 2.1 Anomaly Detection in Petroleum Systems . . . 12
- 2.2 The Petrobras 3W Dataset . . . 12
- 2.3 Deep Learning for Time-Series Anomaly Detection . . . 13
  - 2.3.1 Recurrent Neural Networks . . . 13
  - 2.3.2 Transformer-Based Models . . . 13
  - 2.3.3 Autoencoder-Based Methods . . . 14
- 2.4 Ensemble Methods . . . 14
- 2.5 Model Explainability . . . 14
- 2.6 Outlier Detection and Systematic Reviews . . . 15
- 2.7 Research Gaps and Motivation . . . 15

3 Dataset Materials and Preprocessing **16**
- 3.1 The Petrobras 3W Dataset . . . 16
- 3.2 Exploratory Data Analysis . . . 17
- 3.3 Preprocessing Pipeline . . . 18
  - 3.3.1 Missing Value Handling . . . 18
  - 3.3.2 File-Level Data Splitting . . . 18
  - 3.3.3 Sliding Window Segmentation . . . 20
  - 3.3.4 Statistical Feature Engineering . . . 20
  - 3.3.5 Normalization . . . 21
  - 3.3.6 Class Imbalance Handling . . . 21
- 3.4 Dimensionality Reduction Visualization . . . 21

4 Proposed Methodology **24**
- 4.1 System Overview . . . 24
- 4.2 Random Forest Classifier (Baseline) . . . 24
- 4.3 LSTM Classifier . . . 24
- 4.4 LSTM Autoencoder (Unsupervised) . . . 25
- 4.5 Transformer Classifier . . . 26
- 4.6 Advanced Stacking Ensemble . . . 26
  - 4.6.1 Cross-Validated Out-of-Fold Stacking . . . 26
  - 4.6.2 Meta-Feature Engineering . . . 26

4.6.3 XGBoost Meta-Learner . . . 27
4.7 Loss Functions . . . 27

5 System Implementation **29**
5.1 Implementation Details . . . 29
5.2 Training Configuration . . . 29
5.3 Evaluation Metrics . . . 30

6 Results and Analysis **31**
6.1 Model Comparison . . . 31
6.2 Literature Comparison . . . 33
6.3 Per-Model Analysis . . . 33
6.3.1 Random Forest . . . 33
6.3.2 LSTM Classifier . . . 33
6.3.3 Transformer Classifier . . . 34
6.3.4 LSTM Autoencoder . . . 34
6.3.5 Stacking Ensemble . . . 34
6.4 Ensemble Diversity Analysis . . . 34
6.5 Training Dynamics . . . 38
6.6 Explainability Analysis . . . 38
6.6.1 SHAP Feature Importance . . . 38
6.6.2 Transformer Attention Visualization . . . 39
6.7 Precision-Recall Analysis . . . 41

7 Discussion of Key Findings **43**
7.1 Comparative Model Performance Analysis . . . 43
7.2 The Stacking Ensemble Bringing Models Together . . . 43
7.3 Impact of Focal Loss and Adaptive Thresholding . . . 44
7.4 Data Leakage Prevention via File-Level Splitting . . . 44
7.5 Statistical Feature Engineering Effectiveness . . . 44
7.6 Implications for Operational Deployment . . . 44
7.7 Generalization and Robustness . . . 45

8 Limitations **46**

9 Conclusion and Future Work **47**
9.1 Conclusion . . . 47
9.2 Future Work . . . 47

Bibliography **48**

List of Figures

3.1 Distribution of anomaly classes in the 3W dataset, showing class imbalance. 17
3.2 Correlation heatmap of the eight sensor features, showing inter-feature connections that report feature engineering decisions. 18
3.3 Distribution of classes across train, validation, and test splits. 19
3.4 Bar chart with log scale showing class sample counts across splits 19
3.5 Bar chart of normalized percentage distribution by class 20
3.6 Class distribution before using SMOTE. 22
3.7 Class distribution SMOTE oversampling to achieve balanced class representation with fabricated data. 22
3.8 PCA and t-SNE projections of the preprocessed feature space, showing the class separability landscape that the models must navigate. 23

4.1 Pipeline for anomaly detection in petroleum production wells using the Petrobras 3W dataset. 25

6.1 Radar chart comparing all five models across multiple evaluation metrics, visualizing the trade-offs between accuracy, precision, recall, and F1-scores. 32
6.2 Per-class F1-score comparison across all supervised models, revealing the complementary strengths that motivate the stacking ensemble approach. . 32
6.3 Confusion matrix for the Advanced Stacking Ensemble, showing the distribution of predictions across all nine classes. 35
6.4 Confusion matrix for the Advanced Stacking Ensemble(Normalized). 35
6.5 Confusion matrix for the RF (Normalized). 36
6.6 Confusion matrix difference (Stacking Ensemble and RF). 36
6.7 Diversity analysis of the three base models, confirming complementary error patterns that the meta-learner exploits. 38
6.8 Transformer training and validation loss/accuracy curves, showing smooth convergence with early stopping and learning rate reduction. 39
6.9 SHAP summary plot showing how each feature's value influences model predictions. 40
6.10 Attention heatmap for anomaly samples from the first Transformer encoder layer. 41
6.11 Precision-Recall curves for all supervised models. 41

List of Tables

2.1 Comparison of related works on petroleum anomaly detection 15

3.1 Sensor features in the 3W dataset 16
3.2 Anomaly class definitions in the 3W dataset 17

4.1 Meta-features engineered for the stacking ensemble 27

5.1 Training hyperparameters for deep learning models 29

6.1 Model comparison on the held-out test set. The LSTM Autoencoder is evaluated in binary mode; all other models are evaluated in multi-class mode. 31
6.2 Comparison with published results on petroleum anomaly detection 33

1 Introduction

1.1 Background of the Project

The worldwide petroleum industry operates in an environment of remarkable technical complexity. Offshore production platforms are equipped with many components, such as sensors, valves, and pipelines, that operate under extreme pressures and temperatures. These systems produce a continuous flow of multivariate time-series data. It includes pressure readings at various points along the production string, temperature measurements at critical junctions, and flow-rate indicators. These collectively indicate the well's performance efficiency during gas lift operations and when abnormal situations or anomalous conditions develop. These include scaling buildup within production columns or abrupt events, such as the false closure of a downhole safety valve. The consequences can range from minor production losses to catastrophic equipment failures, environmental contamination, and threats to human life [1].

Anomaly detection in petroleum systems has historically relied on rule-based alarm systems designed by domain engineers [2]. An operator might configure a threshold so that if the pressure at the permanent downhole gauge falls below a specific value, an alarm is triggered. These systems suffer from fundamental limitations. They cannot capture the complex multivariate temporal patterns that often lead to failure, and they produce excessive false alarms, leading to alarm fatigue among operators, and require manual tuning to the unique operating characteristics of each well. As the industry matures, the volume of sensor data has grown exponentially, creating both a challenge and an opportunity. The challenge lies in the sheer scale of the data that human operators cannot manually inspect in real time. The opportunity lies in applying machine learning and deep learning techniques to automatically distinguish normal operating patterns from the early-stage signatures of developing faults.

To address this gap, the Petrobras 3W dataset [1], released by the Brazilian state oil company Petrobras, provides a well-structured, standardized benchmark built from real-world time-series recordings across various offshore production wells. It offers eight sensor measurements and nine well-defined operational states, one normal and eight anomaly types, making it a suitable foundation for a rigorous multi-model comparative study.

1.2 Problem Statement

Following on these foundations, this report aims to address the four most important limitations that face anomaly detection in existing approaches. The prior works adopt a single-model focus, evaluating either Random Forest (RF) [3], a traditional tree-based machine learning model, or Long Short-Term Memory (LSTM) [4], a recurrent deep learning model. But rarely compare both under the same experimental conditions. This makes it hard to determine whether the additional complexity of deep learning is justified for this specific field. Many prior studies create overlapping sliding windows from the time-series

data before splitting into training and test sets. This results in temporally neighboring windows from the same recording. This temporal information leak has exaggerated reported performance metrics, often giving a misleading picture of how models would perform on genuinely unseen data. Few studies combine feature-level understandability methods, such as SHapley Additive exPlanations (SHAP) [5][6], with temporal interpretability methods for attention-weight analysis, and most offer only one form of explanation or none at all. Very few studies have explored stacking ensembles that combine classical, recurrent, and attention-based models on the 3W dataset.

1.3 Research Objectives

The primary objectives of this research are :

1. To design and implement an accurate preprocessing pipeline that includes file-level data splitting to prevent temporal leakage, sliding-window segmentation, statistical feature engineering, and Synthetic Minority Over-sampling Technique (SMOTE) [7][3] based class balancing.

2. To implement and compare five distinct anomaly detection models comprising RF, LSTM, LSTM Autoencoder, Transformer, and a Stacking Ensemble on the Petrobras 3W dataset under the same evaluation conditions.

3. To propose a novel multi-model stacking ensemble that combines a classical tree-based model, a recurrent deep learning model, and an attention-based deep learning model, augmented with engineered meta-features and a gradient-boosted meta-learner.

4. To offer complete model interpretability across dual-level explainability, specifically SHapley Additive exPlanations (SHAP), for identifying where sensor features drive predictions [5], and Transformer attention visualization for revealing if the temporal sequence of the model is distinct from its decision [8].

1.4 Contributions

This report offers four key contributions to the detection of petroleum anomalies.

- The first contribution is the proposed Advanced Stacking Ensemble, a multi-paradigm framework that combines Random Forest, LSTM, and Transformer classifiers for anomaly detection on the Petrobras 3W dataset. Using cross-validated out-of-fold (OOF) stacking with 55 engineered meta-features, prediction entropy, inter-model agreement, and confidence statistics, and an eXtreme Gradient Boosting (XGBoost) meta-learner.

- The second key contribution is the application of Focal Loss (focuses on the examples that the model gets wrong rather than the ones that it can confidently predict) [9]. It is used to train the LSTM and Transformer classifiers, down-weighting easy-to-classify normal samples and sharpening the training signal for hard-to-detect minority anomaly classes.

- The third key contribution is a dual-level understandability framework that combines SHAP analysis for feature-level explanations with Transformer attention to

visualize temporal-level explanations. This framework aims to support operator understanding of model decisions at multiple details, which may assist trust and transparency.

- The fourth key contribution is an evaluation protocol designed to prevent temporal data leakage, achieved through file-level stratified splitting that ensures no temporally overlapping windows cross the train-test boundary.

1.5 Report Organization

The report is organized into nine chapters, each addressing a specific aspect of the anomaly detection study. Below is a summary of the content covered in each chapter:

1. Chapter 2 reviews the initial work on anomaly detection in petroleum systems, covering traditional machine learning approaches and deep learning architectures for time-series classification, ensemble techniques, and model explainability methods.

2. Following this chapter, Chapter 3 presents the Petrobras 3W dataset in detail and the preprocessing pipeline, including missing-value handling, file-level data splitting, sliding-window segmentation, statistical feature engineering, normalization, and class-imbalance handling.

3. Chapter 4 proposed a detailed methodology to describe the architecture of each of the five models. This includes the novel stacking ensemble-up and the loss functions employed.

4. Chapter 5 covers the experimental design, including implementation details, training configuration, and evaluation metrics.

5. Chapter 6 shows results and analysis, while chapter 7 presents the discussion of key findings.

6. Chapter 8 and Chapter 9 show the limitations and conclusions, and what the future implications of the work should be.

2 Related Works

This chapter presents a structured review of the existing research on anomaly detection, with specific highlights on machine learning and deep learning in the oil and gas industry. The approaches implemented for time-series sensor data. The review is structured sectionally, beginning with general anomaly detection in petroleum systems, then narrowing to studies on the Petrobras 3W dataset, followed by specific architectural families: recurrent networks, Transformers, and autoencoders. This report is motivated by research gaps and addresses ensemble methods and model explainability.

2.1 Anomaly Detection in Petroleum Systems

The petroleum industry has experienced rapid growth, which has helped shift from purely physics-based monitoring to data-driven anomaly detection. Shaik and Rao et al. [2] conducted a systematic review using machine learning approaches to predict oil and gas pipeline failures from 2010 to 2023. Their analysis showed that Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are the most commonly implemented techniques. But hybrid models that combine multiple algorithms frequently outperform single-model approaches. They identified deep learning and ensemble methods as promising yet underexplored directions. Their study addresses this gap directly.

Asghar and Khan [10] proposed an anomaly detection approach using five traditional ML algorithms and found that the SVM model with optimized parameters achieved the best performance, with 97.43% accuracy and an ROC-AUC of 0.97 for oil and gas pipelines. Their work did not address temporal dependencies in sensor data. Sarker et al. [11] introduced a deep-learning method, Deeppipe, for detecting multi-product pipeline anomalies that combines simulation data with a deep classifier and achieves 99.5% accuracy. This was great in terms of the results obtained on a proprietary simulated dataset. However, it limits their generalizability to real-world mass-production environments. Uddin et al. [12] present a Convolutional Neural Network (CNN) based approach that achieves 94.2% accuracy for anomaly detection in smart industrial plants using IoT sensor data, reducing false positives from 12% to 3%.

2.2 The Petrobras 3W Dataset

The 3W dataset released by Petrobras [1] has become a standard benchmark for petroleum anomaly detection research since 2019. It offers real-world time-series recordings from various offshore production wells, based on eight sensor measurements, spanning nine well-defined operational states: one normal and eight anomaly types. This offers a level of realistic complexity that simulated datasets cannot match.

LSTM and Gated Recurrent Unit (GRU) architectures optimized with genetic algorithms on the 3W dataset were implemented by Santos and Filho et al. [13], achieving

an F1-score of 0.97. Their work shows the effectiveness and efficiency of hyperparameter optimization in deep learning for petroleum time series. They did not explore ensemble approaches or attention mechanisms, nor did they provide an interpretability analysis. A comparative study of one-class classifiers [14] that includes Isolation Forest (IF), one-class SVM, Local Outlier Factor (LOF), Elliptical, envelope, and Autoencoders, where LOF achieved the best F1-score of 0.87 in real data and 0.915 in simulated data. Their study pointed out that when labeled anomaly data are scarce, unsupervised methods have advantages but are limited to one-class approaches for multi-class fault classification.

LOF was particularly applied to subsurface safety valve anomalies by Santos and Silva [15], achieving an exceptional F1-score of 0.9969, although this was limited to a single, distinct anomaly type with peculiar feature signatures. RF was combined with SHAP and Local Interpretable Model-agnostic Explanations (LIME) by Silva et al. [5] for an understandable anomaly detection, achieving a precision of 99.84% and a recall of 99.91%. Their key contribution was to merge AI into the petroleum anomaly detection workflow, but they had not compared it with deep learning options or explored how different models might work well together through ensemble learning. Real-time event detection using RF and Temporal Convolutional Networks (TCN) was proposed by Lima-Junior and Filho et al. [16], combining traditional and deep approaches without a formal stacking framework to learn to optimally weight their respective strengths.

2.3 Deep Learning for Time-Series Anomaly Detection

2.3.1 Recurrent Neural Networks

LSTM networks have become the preferred architecture for temporal anomaly detection due to their blocking mechanism. This allows them to selectively retain or discard information across time steps, enabling them to capture long-range dependencies that simpler recurrent networks cannot. LSTM Autoencoders for stuck pipe detection in drilling operations were implemented by Al-Rubaii and Khan et al. [17] in the petroleum field, achieving 99.06% accuracy using unsupervised learning on normal operational data. Gradient Boosting for accident detection at directional drilling was implemented by Perevalov et al. [18], attaining an ROC-AUC of 0.908. ROC-AUC is a metric that measures how well a model distinguishes between classes, where 1.0 is perfect, and 0.5 is random guessing[19]. On the other hand, a combined semi-supervised approach was proposed by Papadopoulos et al. [20] that pairs an LSTM Autoencoder with a 2D CNN operating on spectrograms, achieving nearly the best accuracy in noisy industrial environments.

2.3.2 Transformer-Based Models

Transformer architectures were initially proposed by Vaswani et al. [21] for natural language processing and have since been increasingly adopted for time-series tasks. This is because their self-attention mechanism can process all time steps in parallel and express relationships between any pair of positions without narrowing the information inherent to recurrent architectures in the sequence. Transformer networks to the 3W dataset using the TranAD architecture were implemented by Lima-Junior et al. [22], showing their potential for real-time petroleum anomaly detection. An EfficientTransformer that integrates a 1D-CNN for local feature extraction with linear multi-head self-attention was proposed by Zhang and Wang et al. [23], achieving an accuracy improvement of 1 to

2% over standard Transformers. On the other side, reducing computational cost. Lin et al. [24] achieved 98.87% accuracy for insulation oil monitoring using Transformer-based multi-source fusion, and further validated these architectures on industrial sensor data.

2.3.3 Autoencoder-Based Methods

Autoencoders engage with a distinctive position as they do not require labeled anomaly data during training in the anomaly detection landscape. The model completely defines a boundary by learning to rebuild only normal operational patterns. Samples that cannot be accurately rebuilt are likely anomalous. This unsupervised model is particularly significant in petroleum operations, where labeled fault data is scarce and expensive to obtain. Variational Autoencoders with Federated Learning for privacy-preserving anomaly detection in industrial control systems were combined by Friha et al. [25], achieving an F1-score of 0.9992, which shows that autoencoder-based methods can be both highly accurate and compatible with decentralized architectures.

2.4 Ensemble Methods

Ensemble learning combines diverse models and systematically produces better results than any single model by training models that make sufficiently different errors. Stacking, also known as stacked generalization, takes this further by training a meta-learner to optimally combine base-model predictions rather than using a fixed combination rule such as majority voting. This combination of Stacked Denoising Autoencoders and Extreme Learning Machines was demonstrated by Liu and Chen et al. [26] for gas turbine anomaly detection, achieving 99% sensitivity. Though the theoretical appeal and practical success of ensemble methods in other domains are well documented, their application to petroleum anomaly detection remains relatively unexplored in the published literature on the 3W dataset. With a novel multi-model stacking ensemble, this report fills this gap.

2.5 Model Explainability

In the petroleum production field, safety is very critical. A models predictive accuracy is very important, though its not sufficient on its own for deployment. The operators and engineers must also understand why a model makes a specific prediction. SHAP is a theory-driven method from cooperative game theory that connects each prediction to the contribution of individual input features. SHAP and LIME to RF predictions on the 3W dataset implemented by Silva et al. [5] to demonstrate a benchmark for explainable petroleum anomaly detection. Transformer attention weights provide an additional form of interpretability by showing which time steps are attended to. The model focuses on making each prediction.The study combines SHAP-based feature explanations and Transformer attention-based temporal analysis to improve model interpretability on the Petrobras 3W dataset. Although attention weights can provide useful temporal insight, prior work has shown that attention should not always be treated as a faithful explanation of model decisions [27].

2.6 Outlier Detection and Systematic Reviews

The many recent systematic reviews show a wide context for this work. For production time series in a meta-analysis, Syed et al. [28] evaluated 17 outlier detection techniques. The resulting finding is that eight unsupervised algorithms perform far better than the others, with the K-Nearest Neighbors (KNN) and Fulford methods ranking highest. Deep reinforcement learning (DRL) for anomaly detection was reviewed by Nassif et al. [29]. He found DRL models perform better than traditional approaches but are considerably more complex to implement. A hybrid approach combining Density-Based Spatial Clustering of Applications with Noise (DBSCAN) for unsupervised outlier detection with Gradient was proposed by BoostingZhu et al. [30] for well-log reconstruction. This achieves high accuracy under challenging open-hole conditions. These reviews highlighted a move toward using multi-method approaches and the need for systematic benchmarking, both of which are integral to the design of the current study.

2.7 Research Gaps and Motivation

Based on this complete review, this report is motivated by four specific research gaps. The foremost gap is the lack of a systematic comparison of traditional ML, recurrent DL, attention-based DL, and unsupervised approaches on the 3W dataset under the same evaluation conditions. While most studies evaluate one or two models in isolation. Secondly, there is limited work on multi-model stacking ensembles combining these diverse model types with meta-feature engineering has yet been proposed for this specific field. Third, overlapping sliding windows that cause data leakage and need to be prevented have received barely any attention, leaving doubt on the generalizability of some published results [31]. Fourth, for petroleum anomaly detection on the 3W dataset, there is no combined feature-level and temporal-level interpretability framework. Table 2.1 shows a summarization of the positioning of this study relative to key related works.

Table 2.1: Comparison of related works on petroleum anomaly detection

Study	**Dataset**	**Multi-Model**	**Ensemble**	**XAI**	**Best F1**
Silva et al. [5] RF+SHAP	3W	No	No	SHAP/LIME	0.9984
Santos et al. [15] LOF	3W (DHSV)	No	No	No	0.9969
Deeppipe [11]	Pipeline sim.	No	No	No	0.9940
Santos & Filho [13] LSTM/GRU	3W	Yes	No	No	0.9700
Al-Rubaii & Khan [17] LSTM-AE	Volve	No	No	No	0.9415
Oliveira et al. [14] One-Class	3W	Yes	No	No	0.8700
This Study	3W	Yes (5)	Yes	SHAP+Attn	0.9622

3 Dataset Materials and Preprocessing

3.1 The Petrobras 3W Dataset

The 3W dataset [1] is an open-source research dataset created by a Brazilian state-controlled petroleum corporation, Petrobras. The dataset contains time-series data with multiple variables collected from offshore oil production wells using various downhole and surface sensors. It was specifically designed to support the development of anomaly detection and event classification algorithms, which have since become standard benchmarks in the petroleum anomaly detection literature. The dataset contains three types of instances: real well data (prefixed WELL), simulated data (SIMULATED), and hand-drawn instances (DRAWN). This study uses only the real well instances to ensure that the evaluation reflects the characteristics and noise patterns of actual production environments.

Each sample in the dataset contains eight sensor continuous measurements captured at regular time steps. These sensors monitor pressure at various points along with the production string (P-PDG at the permanent downhole gauge, P-TPT at the temperature/pressure transducer, P-MON-CKP at the choke monitoring point, and P-JUS-CKGL downstream of the gas lift valve), temperatures at critical junctions (T-TPT, T-JUS-CKP, and T-JUS-CKGL), and the gas lift flow rate (QGL). Table 3.1 shows the features of the sensors that are used in this dataset. These eight measurements capture the thermodynamics and fluid dynamics in the well. This provides information about the diverse operational conditions and fault signatures present in the data.

In the Table 3.2, it shows the anomaly class definitions in the dataset. Abrupt anomalies hit fast, gradual ones sneak up slowly, and oscillatory ones keep going back and forth repeatedly. Class 0 represents normal operation, while classes 1 through 8 represent different types of anomalies. Abrupt increase of basic sediment and water (BSW), spurious closure of the downhole safety valve (DHSV), severe slugging, flow instability, rapid pro-

Table 3.1: Sensor features in the 3W dataset

Abbreviation	**Full Name**	**Unit**
P-PDG	Pressure at Permanent Downhole Gauge	kPa
P-TPT	Pressure at Temperature/Pressure Transducer	kPa
T-TPT	Temperature at Temp./Pressure Transducer	°C
P-MON-CKP	Monitoring Pressure at Choke	kPa
T-JUS-CKP	Temperature Downstream of Choke	°C
P-JUS-CKGL	Pressure Downstream of Gas Lift Valve	kPa
T-JUS-CKGL	Temperature Downstream of Gas Lift Valve	°C
QGL	Gas Lift Flow Rate	m^3/d

Table 3.2: Anomaly class definitions in the 3W dataset

Class ID	Description	Pattern Type
0	Normal Operation	
1	Abrupt Increase of BSW	Abrupt
2	Spurious Closure of DHSV	Abrupt
3	Severe Slugging	Oscillatory
4	Flow Instability	Oscillatory
5	Rapid Productivity Loss	Gradual
6	Quick Restriction in PCK	Abrupt
7	Scaling in PCK	Gradual
8	Hydrate in Production Line	Gradual

ductivity loss, quick restriction in the production column (PCK), scaling in the PCK, and hydrate formation in the production line. These anomalies show a wide range of physical situations, from mechanical failures to chemical processes.

3.2 Exploratory Data Analysis

Initially, a thorough investigatory data analysis was done to understand the characteristics of the 3W dataset. An extreme class imbalance was identified across the full dataset. Figure 3.1 shows that normal operation (Class 0) dominates the dataset, whereas some anomaly classes, such as Class 1 (BSW increase) and Class 7 (scaling), contain very few samples. This imbalance creates a fundamental challenge for supervised learning. They prioritize the majority class and mostly neglect the rare anomalies.

Figure 3.2 shows moderate-to-strong connections among certain sensor groups. Particularly between pressure and temperature measurements. These patterns suggest that feature reduction or selection could be beneficial. They also motivate the statistical feature engineering steps described later.

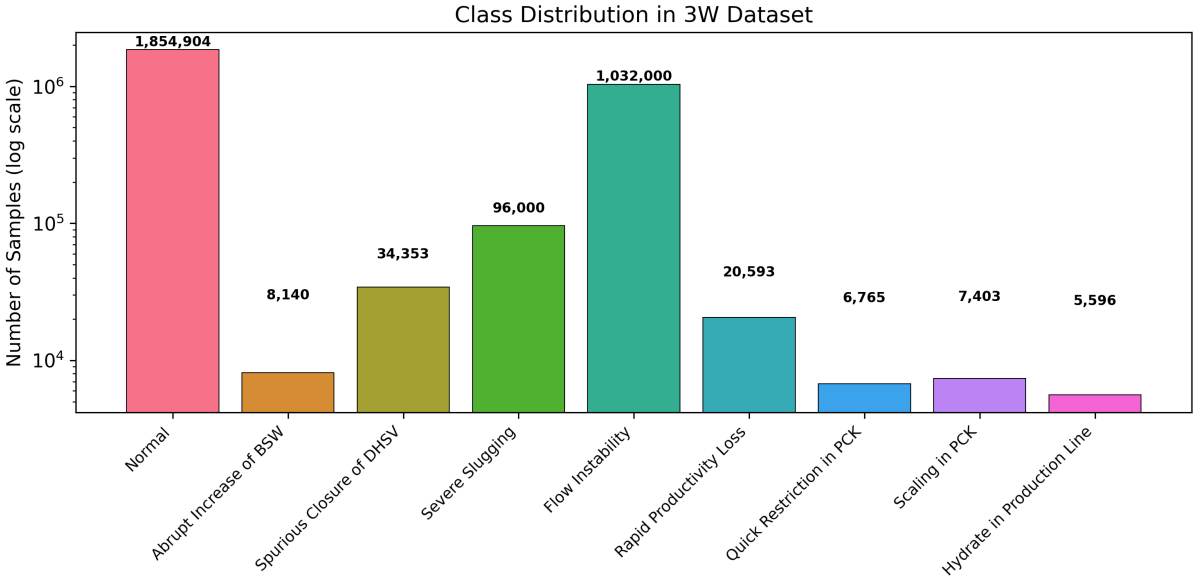


Figure 3.1: Distribution of anomaly classes in the 3W dataset, showing class imbalance.

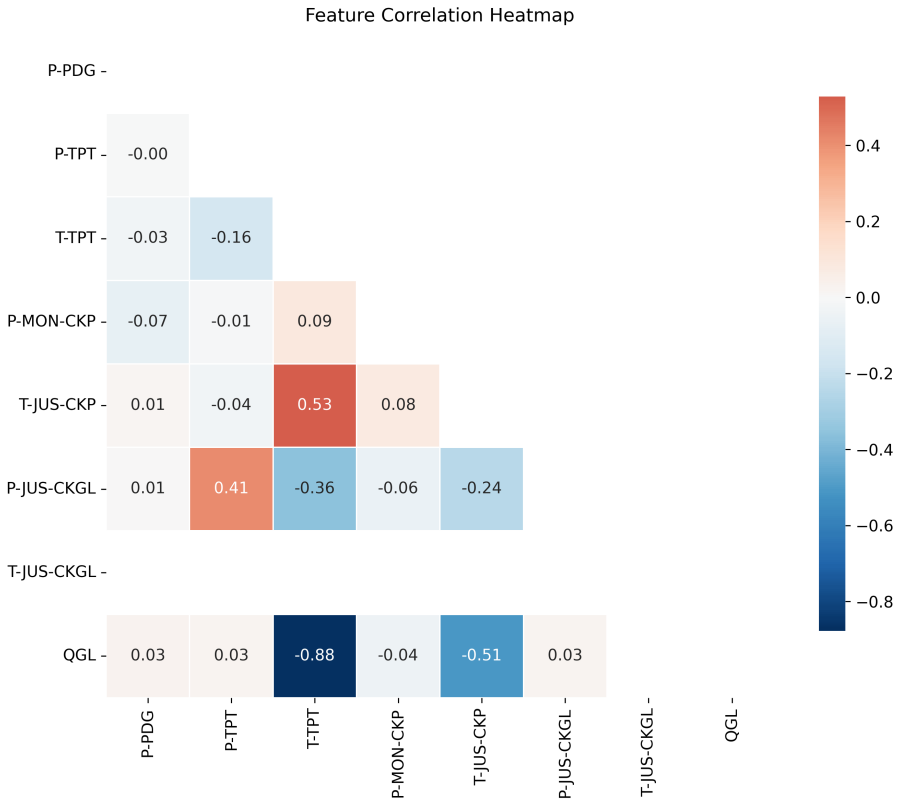


Figure 3.2: Correlation heatmap of the eight sensor features, showing inter-feature connections that report feature engineering decisions.

3.3 Preprocessing Pipeline

3.3.1 Missing Value Handling

Raw industrial sensor data, as expected, contains missing values, infinite readings from sensor malfunctions, and communication failures. To address these issues, all infinite values are replaced with NaN, and extreme values exceeding the float32-supported range are similarly treated as missing. Features with more than 60% of values missing are dropped entirely, as they provide insufficient signal for learning. The remaining gaps are then filled using forward-fill, followed by backward-fill, within each source file group, keeping the time continuity.

3.3.2 File-Level Data Splitting

In time-series classification, a critical and often overlooked methodological decision is how to partition the data into training, validation, and test sets. The basic approach is to create sliding windows from all files, then randomly split the windows into partitions, which creates serious data leakage. Because adjacent windows overlap by design (the stride is 25 with a window size of 50, giving 50% overlap). Random splitting may put consecutive windows from the same well recording into both the training and test sets.

This work uses file-level splitting to prevent these leakages. All the CSV files represent a complete recording session from a specific well. They are assigned to exactly one partition before any sliding windows are created. Stratified splitting makes sure that all nine classes are represented in each partition. The rare-class reservation strategy ensures that minority classes with very few source files have at least one file in every split. The final proportions

are 69.7% training, 15.1% validation, 15.2% test. Figure 3.3, 3.4, 3.5 shows the graphical distribution of classes across train, validation, and test splits. Windows are then created within each partition separately. So, no window extends the partition boundary.

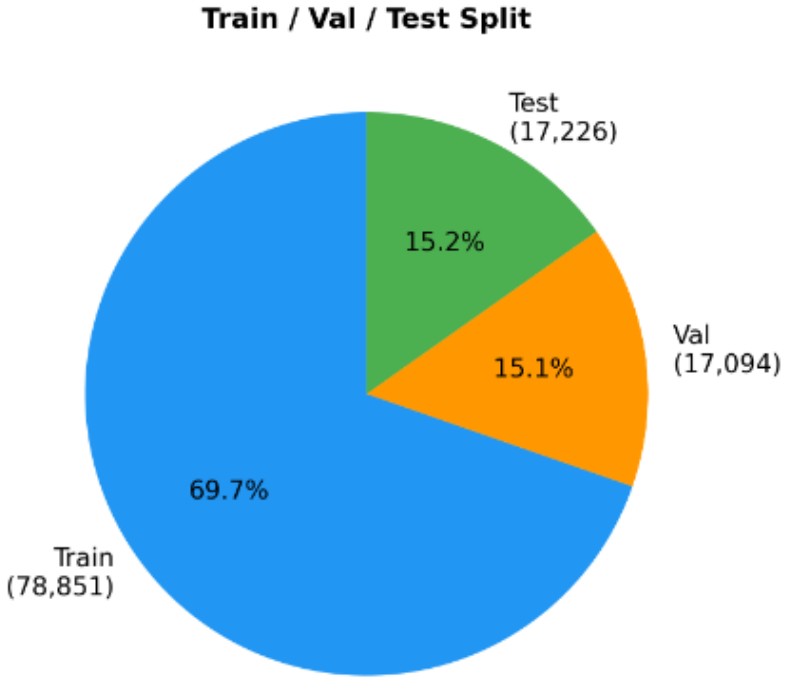


Figure 3.3: Distribution of classes across train, validation, and test splits.

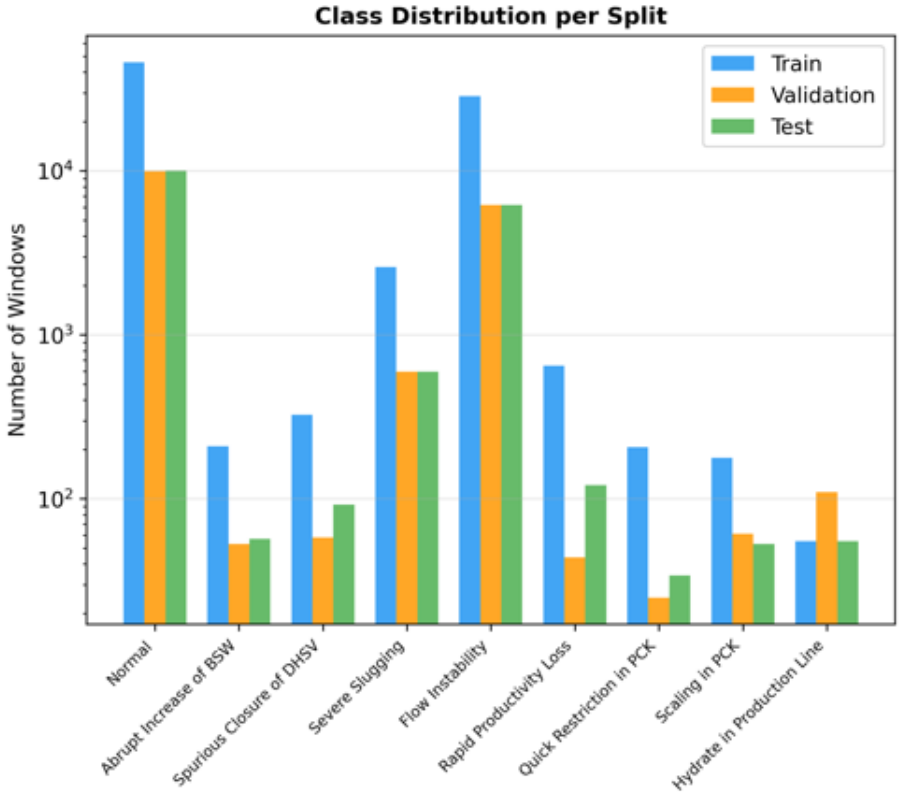


Figure 3.4: Bar chart with log scale showing class sample counts across splits

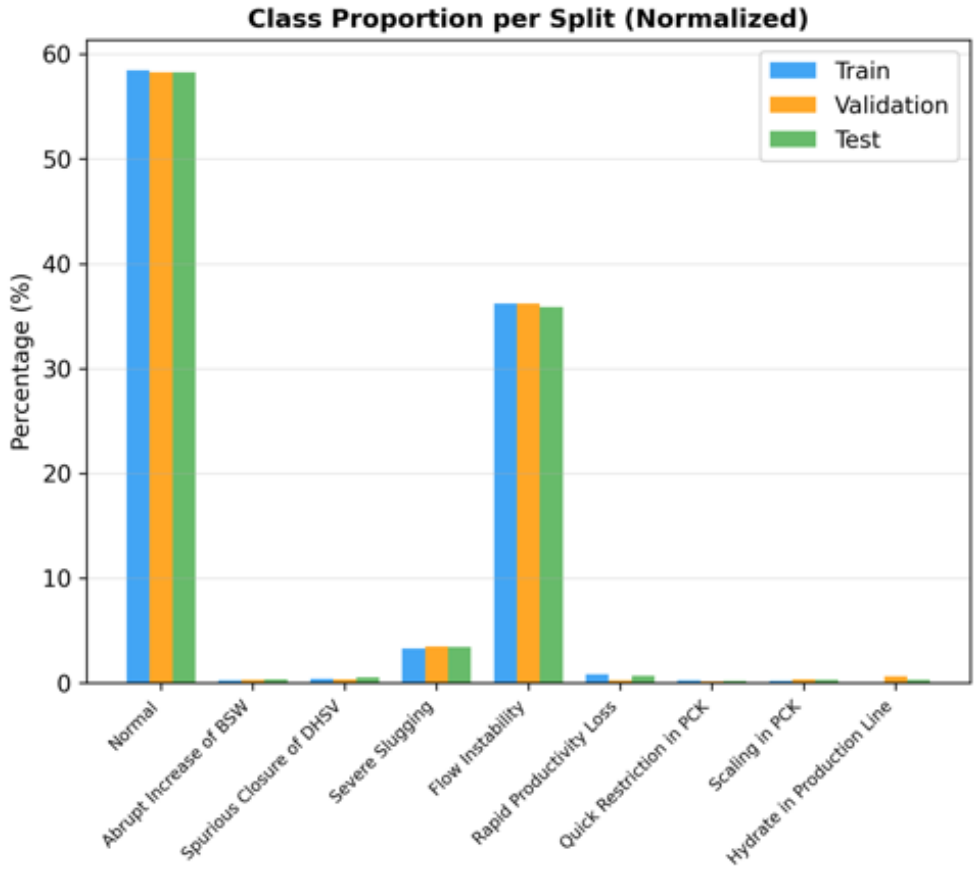


Figure 3.5: Bar chart of normalized percentage distribution by class

3.3.3 Sliding Window Segmentation

The continuous time-series data is divided into fixed-size windows of 50 time steps with a step size of 25 (50% overlap). Each window represents around a 50-second sample of the wells operational state. The most frequent class label across the 50 time steps becomes the window label by majority vote. Windows never cross file boundaries, ensuring each window contains a well-structured portion from a single session. This produces 3D tensors of shape (N, 50, F) where N is the number of windows and F is the number of features.

3.3.4 Statistical Feature Engineering

Along with the eight raw sensor features, statistical features are computed over each window to improve the representation and provide the models with clear information about the temporal dynamics. For each of the eight base features, five summary statistics are calculated. The rolling mean (capturing the central trend), rolling standard deviation (capturing volatility), rolling minimum, rolling maximum (together capturing the range of variation), and the rate of change computed as the difference between the last and first time step divided by the window length (capturing the direction and speed of change). These statistics are expanded across all time steps within the window. This increases the feature dimension from 8 raw features to 48 enriched features per time step. The final windowed tensors have shape (N, 50, 48).

3.3.5 Normalization

StandardScaler [32] normalization is applied to transform each feature to zero mean, and unit variance. The scaler is trained only on the training data and then applied to transform the validation and test data. Preventing any information from future data leaking into the training process. The trained scaler is saved to disk so the result can be produced again and reused later.

3.3.6 Class Imbalance Handling

The extreme class imbalance in the 3W dataset is handled using three strategies. For the RF Classifier, SMOTE is applied to the flattened training data to generate artificial samples for minority classes. Because SMOTE operates in the flattened feature space, it may generate samples that do not fully preserve temporal and physical consistency. Therefore, SMOTE was applied only to the RF baseline , while deep learning models used class weighting and Focal Loss. SMOTE [3] operates in the feature space by filling the values between existing minority samples and their nearest neighbors, producing synthetic examples that expand the minority class representation. Figure 3.6 and Figure 3.7 shows the difference before and after using SMOTE. Two strategies are used for the deep learning models. The first one is inverse-frequency class weights, which are computed from the training label distribution and added to the loss function. So that misclassifying a rare anomaly incurs a relatively higher penalty. For the second one, Focal Loss is applied instead of standard cross-entropy. It helps to focus on the rare cases rather than wasting time on artificially made samples.

3.4 Dimensionality Reduction Visualization

To visualize the high-dimensional preprocessed feature space, Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE) projections Figure 3.8 were computed. PCA is a technique that reduces high-dimensional data to fewer dimensions by finding the directions of maximum variance, as used by Marti et al. [33]. Tao et al. used t-SNE, a technique that reduces high-dimensional data to 2D or 3D for visualization while preserving local neighborhood structure [34]. Figure 3.8 show that some anomaly classes form mostly separated clusters in the reduced space, while others overlap significantly with the normal class. This confirms the difficulty of the classification problem and the need for high-level modeling approaches.

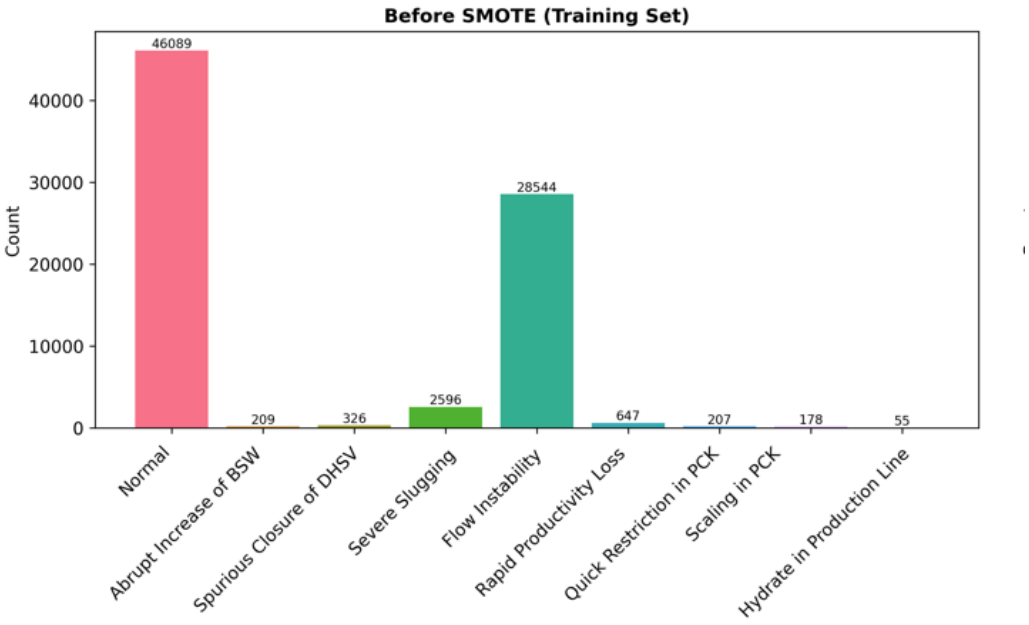


Figure 3.6: Class distribution before using SMOTE.

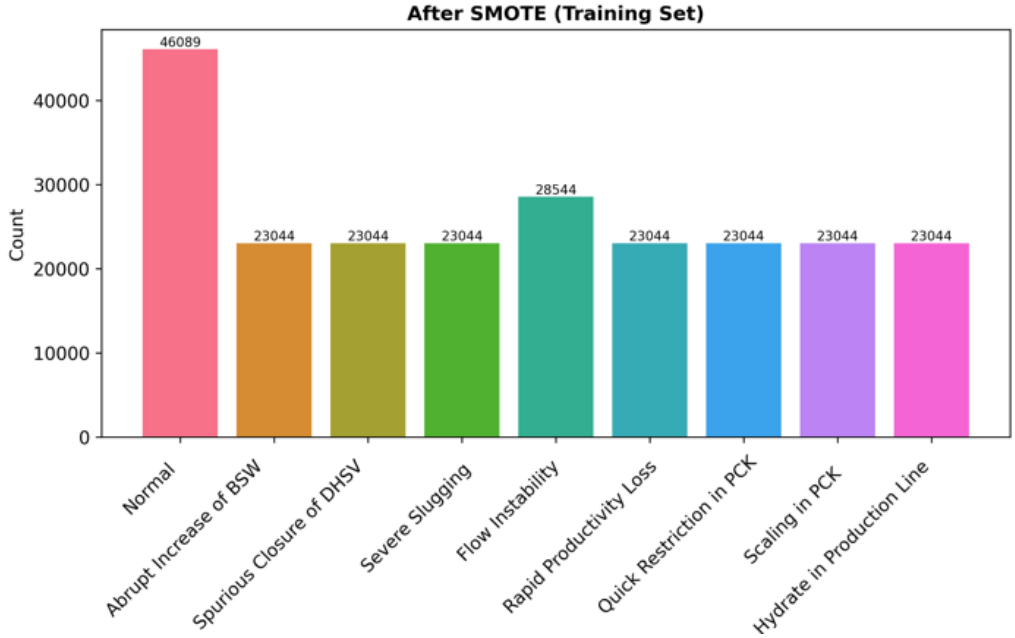


Figure 3.7: Class distribution SMOTE oversampling to achieve balanced class representation with fabricated data.

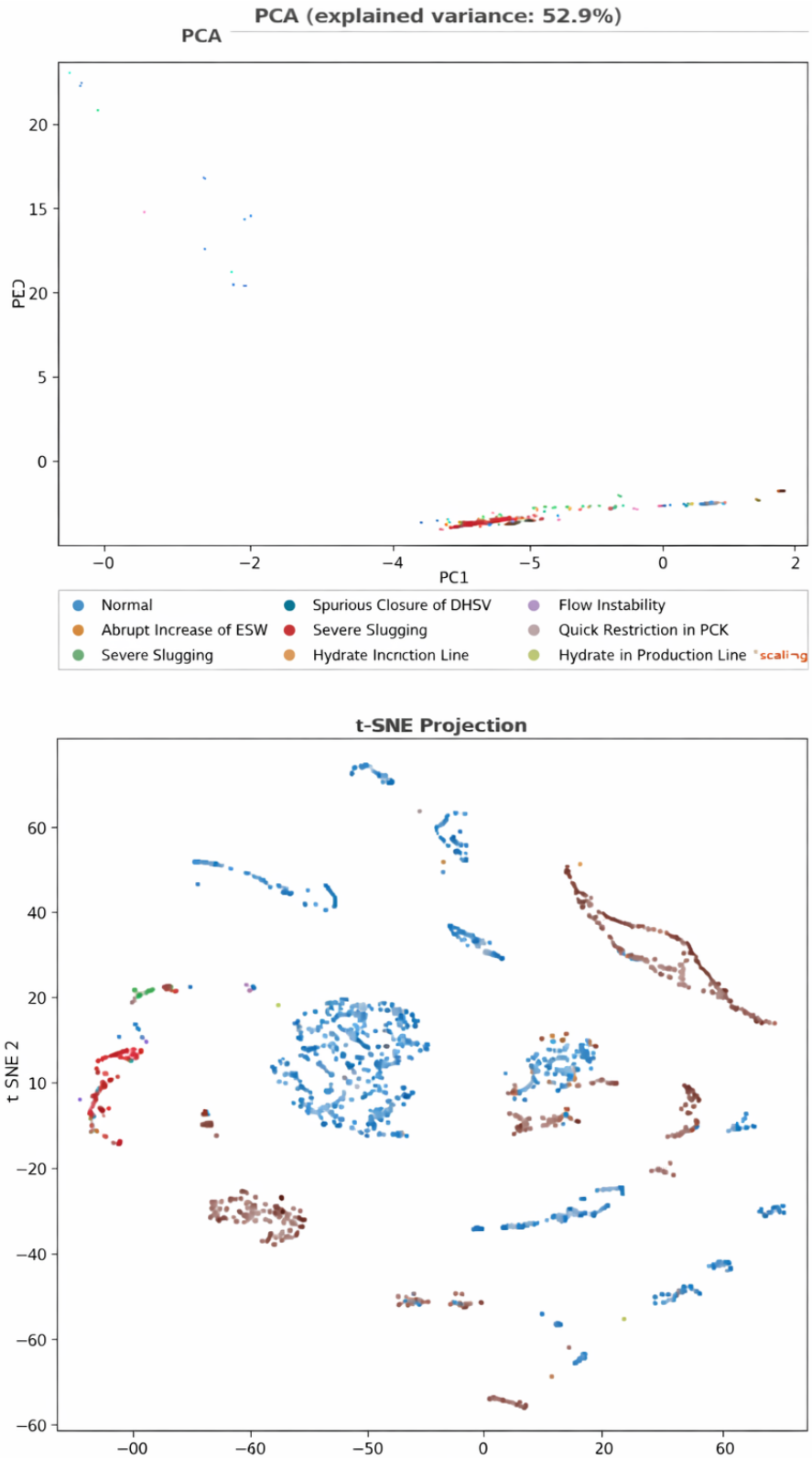


Figure 3.8: PCA and t-SNE projections of the preprocessed feature space, showing the class separability landscape that the models must navigate.

4 Proposed Methodology

4.1 System Overview

The methodology includes five anomaly detection models implemented in a unified pipeline that ensures identical data preprocessing, training conditions, and evaluation protocols across all models as shown in Figure 4.1. These models cover three fundamental learning paradigms. The RF classifier is a classical machine learning method that provides a strong baseline, using engineered features without temporal modeling. The LSTM and Transformer classifiers are supervised deep learning models, each managing time series data directly but through fundamentally different mechanisms. Recurrent state propagation versus parallel self-attention. The LSTM Autoencoder is an unsupervised deep learning model trained exclusively on normal data that detects anomalies through reconstruction error. A recent survey demonstrates that LSTM and Transformer models perform well for time-series anomaly detection in identifying faults in industrial systems [35], although they are not generally anomaly detection models. The Advanced Stacking Ensemble represents the novel contribution, combining the three supervised models through a meta-learning architecture.

4.2 Random Forest Classifier (Baseline)

The RF is a classical machine-learning baseline. As tree-based models require tabular (2D) data, the windowed tensors of shape (N, 50, 48) are flattened to (N, 2400) by linking all time steps [19]. This flattening loses precise temporal ordering but provides the trees with access to the full feature-time landscape. Hyperparameter optimization is performed using RandomizedSearchCV (a powerful, efficient tool in Python's scikit-learn library used for hyperparameter tuning) with 20 iterations and 2-fold cross-validation. It searches over the number of estimators (100 to 500 trees) and the maximum tree depth (10 to unlimited). Also, minimum samples for splitting and leaf nodes, feature subsampling strategy, and class weighting scheme. The use of balanced class weights ensures that the RF charges misclassification of minority classes in proportion to their rarity.

4.3 LSTM Classifier

The LSTM classifier is a two-layer LSTM network with a hidden dimension of 128 units per layer. The LSTM processes the temporal sequence in its natural sequential order, maintaining a hidden state vector that is updated at each of the 50 time steps according to the LSTM gating equations.

The input, forget, and output gates control the flow of information into, within, and out of the cell state. This enables the network to selectively remember relevant patterns across time. After processing all 50 time steps, the final hidden state vector (a

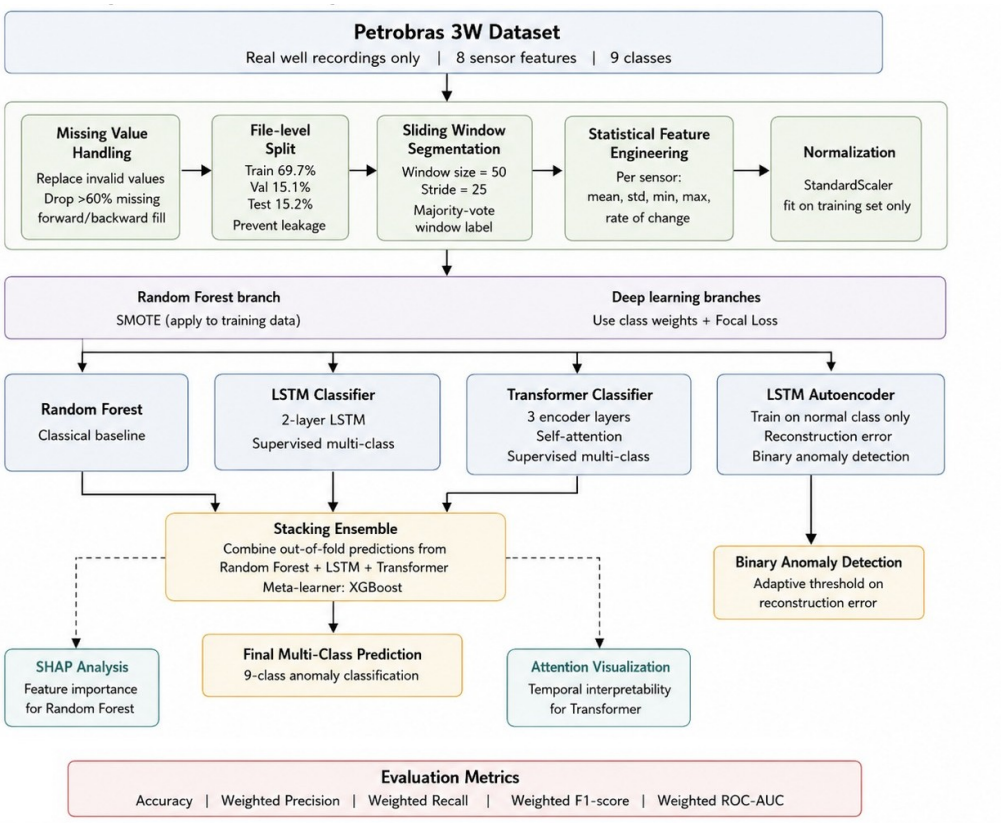


Figure 4.1: Pipeline for anomaly detection in petroleum production wells using the Petrobras 3W dataset.

128-dimensional summary of the entire sequence) is passed through a fully connected layer that maps to the nine output classes [4].

The model contains approximately 188,000 trainable parameters and is trained with Focal Loss ($\gamma = 2.0$) to focus the training signal on hard-to-classify minority-anomaly samples.

4.4 LSTM Autoencoder (Unsupervised)

The LSTM Autoencoder is an unsupervised machine learning method. Rather than learning to discriminate between classes, it learns to reconstruct normal operational patterns, and any sample it cannot accurately reconstruct is flagged as abnormal. The encoder is a two-layer LSTM network with 64 hidden units that reduces each 50-time-step input sequence to a compact hidden representation, obtained from the final hidden and cell states [36].

The decoder is also a two-layer LSTM with 64 hidden units. It attempts to reconstruct the original sequence from this hidden representation. A fully connected output layer maps the decoders hidden states back to the original 48-dimensional feature space. The model is trained exclusively on normal data (Class 0) using a mean squared error (MSE) reconstruction loss.

The reconstruction error for each test sample is computed as the MSE between the input and its reconstruction, averaged across all time steps and features. Rather than using a fixed statistical threshold (such as the mean $+3\sigma$), this work employs an adaptive threshold-optimization strategy. The threshold is searched over percentiles of the reconstruction error distribution on the validation set, and the value that maximizes the

weighted F1 Score is selected.

This data-driven approach ensures that the decision boundary is adjusted to the actual separation between normal and abnormal reconstruction errors.

4.5 Transformer Classifier

The Transformer classifier uses self-attention to learn relationships among all time steps simultaneously, rather than processing them sequentially as in recurrent models. The input features at each time step are first projected from the 48-dimensional enhanced feature space to a 64-dimensional model space through a learned linear transformation. Sinusoidal positional encoding is added to incorporate time-order information, since self-attention does not inherently encode sequence order.

Three Transformer encoder layers are employed. Each layer consists of multi-head self-attention with 4 heads (each attending to 16-dimensional subspaces), a position-wise feed-forward network that expands to 256 dimensions with GELU activation, residual connections, and layer normalization.

Global average pooling aggregates the 50 per-time-step representations into a single 64-dimensional vector. This vector is then passed through a dropout layer ($p = 0.2$) followed by a fully connected layer that maps to the 9 output classes. The model contains approximately 220,000 trainable parameters and, like the LSTM model, is trained with Focal Loss ($\gamma = 2.0$).

4.6 Advanced Stacking Ensemble

The Advanced Stacking Ensemble is the primary novel contribution of this study. It is based on the observation that different model families commit different errors. The RF does very well at feature-based pattern discrimination with strong penalties, but cannot model temporal dynamics. The LSTM captures sequential dependencies and temporal evolution but struggles with long-range data samples. The Transformer attends to global temporal patterns and provides understandable attention weights, but it requires large amounts of data. A stacking ensemble that learns to combine these different perspectives most effectively should outperform any individual model, particularly on unclear cases where models disagree.

4.6.1 Cross-Validated Out-of-Fold Stacking

A simple stacking approach, training the meta-model on predictions from the same training data used to train the base models. It can lead to serious overfitting because the base models have already memorized the training data. The training set was divided into five folds. For each fold, base models were trained on four folds and used to generate OOF predictions on the held-out fold. These predictions were then used to train the meta-learner. These OOF predictions are connected across all five folds to form the complete meta-training set. This ensures that the meta-learner never sees a samples prediction that was generated by a model that also trained on that sample.

4.6.2 Meta-Feature Engineering

Rather than feeding the meta-learner only the raw stacked probabilities (3 models $\times$ 9 classes = 27 features), this report introduces an additional 28 engineered meta-features

that capture higher-order patterns in the base models behavior.

To capture the informativeness and discriminative power of meta-features, we compute the Shannon Entropy for each meta-feature across the output group predictions. Shannon Entropy was also used by Weiying et al. in their fault detection report on the gas turbine [37]. For each of the three base models, the prediction entropy is computed as:

$$H = -\sum p \log p$$

which provides a measure of the models uncertainty. The maximum predicted probability captures each models confidence, while the hard prediction (arg max) records its discrete class decision.

An inter-model agreement feature measures the instances in which all three models predict the same class. Additionally, the per-class mean and standard deviation of the predicted probabilities across models capture the degree of agreement and disagreement for each anomaly type.

These 55 meta-features shown in Table 4.1 provide the meta-learner with a rich representation. It captures not only what each model predicted, but also how confident it was. Whether the models agreed, and where their predictions disagreed.

Table 4.1: Meta-features engineered for the stacking ensemble

Meta-Feature	**Count**	**Description**
Raw stacked probabilities	27	P(class | model) for each model–class pair
Prediction entropy	3	H = -Σ p log p per model
Max confidence	3	max P(c|model) per model
Hard predictions	3	argmax P(c|model) per model
Inter-model agreement	1	Fraction of models agreeing
Mean probability per class	9	$\bar{P}(c)$ averaged across models
Std of probability per class	9	σ(P(c)) across models
Total	55	

4.6.3 XGBoost Meta-Learner

The meta-learner is an XGBoost classifier configured with 300 estimators, a maximum depth of 6, a learning rate of 0.05, a subsample ratio of 0.8, a column subsampling ratio of 0.8, and L1/L2 regularization ($\alpha = 0.1$, $\lambda = 1.0$).

XGBoost is selected for its robustness to overfitting and its ability to efficiently handle the 55-dimensional meta-feature space. It is particularly effective at capturing non-linear interactions among the base models predictions.

The meta-learner is trained on OOF meta-feature vectors and evaluated on the test set. For evaluation, the base models predictions on the test data are transformed into the same 55-dimensional meta-feature representation, which is then provided as input to the XGBoost model for final prediction [38].

4.7 Loss Functions

All supervised deep learning models in this report are trained using Focal Loss [39], a modification of the standard cross-entropy loss designed to address class imbalance.

The Focal Loss is defined as:

$$\mathrm{FL}(p_t) = -(1 - p_t)^{\gamma} \log(p_t)$$

where p_t is the predicted probability for the ground-truth class. The key innovation is the modulating factor $(1-p_t)^\gamma$, which down-weights the loss contribution from easy-to-classify samples (those with high p_t) and amplifies the contribution from hard-to-classify samples (those with low p_t).

With $\gamma = 2.0$ as used in this work, a sample predicted with 90% confidence contributes only about 1% of the loss it would incur under standard cross-entropy, whereas a sample predicted with only 10% confidence contributes nearly the full loss.

The per-class weight α_t further addresses class imbalance by assigning higher importance to rare anomaly classes. This combination of focal modulation and class weighting is particularly well-suited to the 3W dataset, where normal operational data significantly outnumber anomaly instances.

placeins

5 System Implementation

5.1 Implementation Details

The pipeline is implemented in Python. PyTorch is used for deep learning models (LSTM, LSTM Autoencoder, Transformer). Scikit-learn handles RF and preprocessing. XGBoost is used as the stacking meta-learner. CUDA with mixed-precision (float16) speeds up training. SMOTE balances the data, and SHAP is used for interpretation. A fixed random seed of 42 was applied throughout to ensure reproducibility.

5.2 Training Configuration

The deep learning models share a common training configuration to ensure a fair comparison (Table 5.1). The Adam optimizer [40] is used with a learning rate of 1×10^{-3} and weight decay of 1×10^{-4}. It gives adaptive per-parameter learning rates with separated weight regularization. The learning rate is reduced by 0.5. When the validation loss stabilizes in steady regions for 3 continuous epochs, implemented through PyTorchs ReduceLROnPlateau scheduler [41]. Gradient clipping with a maximum norm of 1.0 helps prevent exploding gradients, a common concern in recurrent architectures. Early stopping with patience of 7 epochs monitors the validation loss and halts training when no improvement is observed, preventing overfitting. The model checkpoint with the highest validation F1-score is restored as the final model. Training uses a batch size of 256 and a maximum of 50 epochs, though early stopping typically triggers well before this limit.

Table 5.1: Training hyperparameters for deep learning models

Parameter	**Value**
Batch size	256
Learning rate	1×10^{-3}
Optimizer	AdamW (weight decay = 10^{-4})
Maximum epochs	50
Early stopping patience	7 epochs
LR scheduler	ReduceLROnPlateau (factor=0.5, patience=3)
Gradient clipping	max_norm = 1.0
Mixed precision	Enabled (CUDA AMP)
Loss function	Focal Loss ($\gamma = 2.0$)

5.3 Evaluation Metrics

Model performance is evaluated using six metrics. Accuracy measures the overall fraction of correct predictions. It can be misleading when classes are imbalanced. Weighted precision, recall, and F1-score each weigh a class's contribution by the number of samples it has, giving a summary that reflects the actual class distribution. The weighted ROC-AUC measures the area under the ROC curve using one-vs-rest binarization, reflecting how well each model separates anomalous samples from normal ones across different benchmarks. placeins

6 Results and Analysis

6.1 Model Comparison

Table 6.1 presents the comprehensive results of all five models on the held-out test set. The results reveal several important findings. The Stacking Ensemble achieves the highest weighted F1-score of 0.9622 and the highest ROC-AUC of 0.9962, confirming that the multi-paradigm meta-learning approach successfully exploits the complementary strengths of the three base models. From Figure 6.1, it can be seen graphically that the RF achieves the raw accuracy at 95.98%, demonstrating that classical machine learning with proper feature engineering remains remarkably competitive even against deep learning alternatives. The Transformer classifier achieves a weighted F1-score of 0.9295, outperforming the LSTM (0.9237) by approximately 0.6 percentage points, as shown in Figure 6.2. This suggests that the self-attention mechanisms ability to model global temporal relationships confers a modest advantage over the LSTMs sequential processing for this dataset. The LSTM Autoencoder, operating in the binary detection regime without any labeled anomaly data, achieves 86.07% accuracy and a weighted F1-score of 0.8588. Because the LSTM Autoencoder performs binary detection rather than multi-class fault diagnosis, its metrics are not directly comparable to the supervised multi-class models. The Stacking Ensembles superior weighted F1 among multi-class models indicates that combining diverse base models helps improve detection across the full spectrum of anomaly types.

Table 6.1: Model comparison on the held-out test set. The LSTM Autoencoder is evaluated in binary mode; all other models are evaluated in multi-class mode.

Model	**Accuracy**	**Prec. (W)**	**F1 (W)**	**ROC-AUC (W)**
Stacking Ensemble	0.9625	0.9670	0.9622	0.9962
Random Forest	0.9598	0.9531	0.9558	0.9949
Transformer	0.9302	0.9329	0.9295	0.9820
LSTM	0.9244	0.9274	0.9237	0.9712
LSTM Autoencoder	0.8607	0.8629	0.8588	—

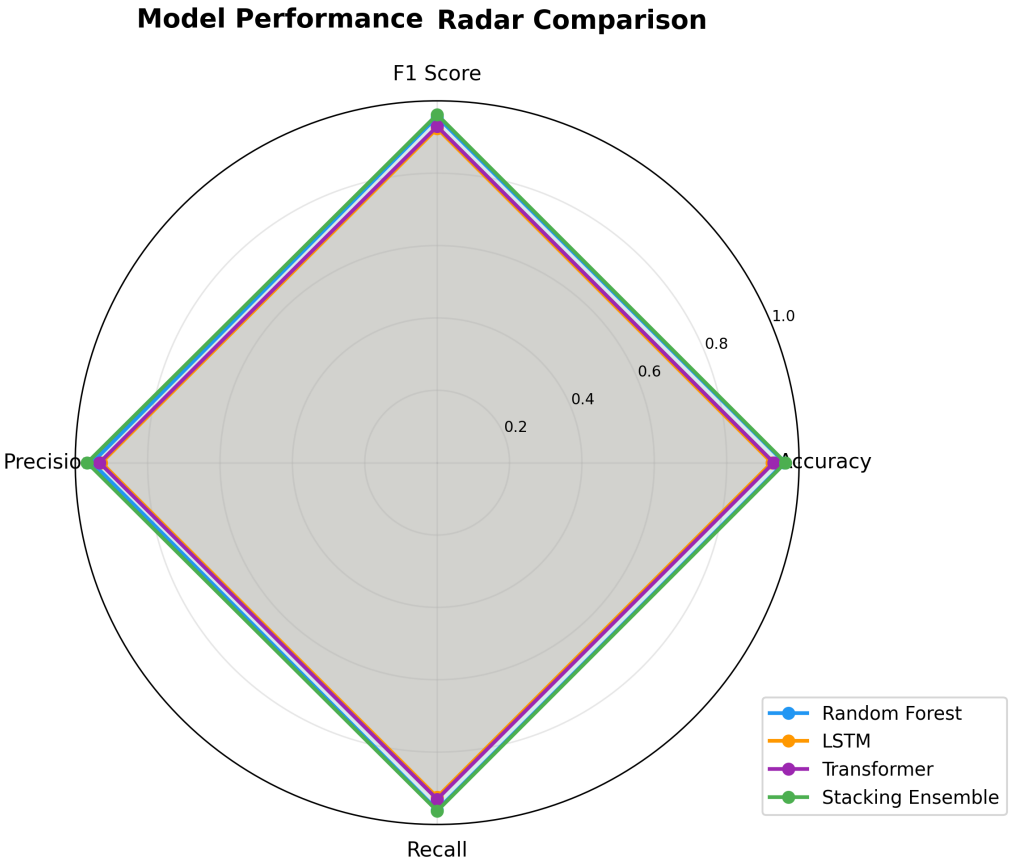


Figure 6.1: Radar chart comparing all five models across multiple evaluation metrics, visualizing the trade-offs between accuracy, precision, recall, and F1-scores.

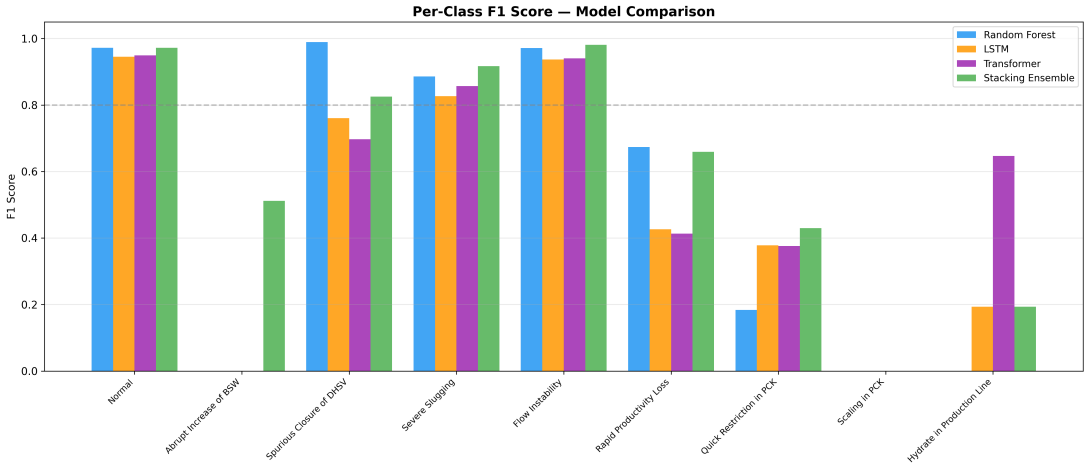


Figure 6.2: Per-class F1-score comparison across all supervised models, revealing the complementary strengths that motivate the stacking ensemble approach.

6.2 Literature Comparison

Table 6.2: Comparison with published results on petroleum anomaly detection

Method	**Weighted F1-score**	**Reference**
RF + SHAP/LIME	0.9984	[5]
LOF Unsupervised	0.9969	[15]
DeepPipe	0.9940	[11]
LSTM/GRU + Genetic Algorithm	0.9700	[13]
Stacking Ensemble	0.9622	This Study
LSTM Autoencoder for Stuck Pipe Detection	0.9415	[17]
One-Class Classifiers on Simulated 3W Data	0.9150	[14]
One-Class Classifiers on Real 3W Data	0.8700	[14]

Table 6.2 shows our results against published benchmarks on petroleum anomaly detection. Our models are competitively positioned. The Stacking Ensembles F1-score of 0.9622 places it above the LSTM Autoencoder drilling benchmark (0.9415) and Oliveira et al.s one-class classifiers (0.87). While falling below the top-performing published results from Silva et al. (0.9984) and Santos et al. (0.9969). These comparisons must be interpreted carefully. The studies reporting the highest F1-scores often evaluate on subsets of the 3W dataset, target specific anomaly types, or use splitting methodologies that may permit temporal leakage. Our file-level splitting strategy provides a more conservative and realistic estimate of generalization performance, which accounts for the modest gap relative to the most optimistic published numbers.

6.3 Per-Model Analysis

6.3.1 Random Forest

The RF classifier achieves 95.98% accuracy and a weighted F1-score of 0.9558. Its per-class analysis reveals a clear pattern: it performs exceptionally well on majority classes (Class 0: F1 = 0.9720, Class 4: F1 = 0.9711) and certain distinct minority classes (Class 2: F1 = 0.9891, Class 3: F1 = 0.8860), but completely fails on very rare classes (Class 1: F1 = 0.0, Class 7: F1 = 0.0, Class 8: F1 = 0.0). This pattern suggests that the Random Forest can learn robust decision boundaries when sufficient training examples exist, but cannot generalize to classes with extremely limited representation, even with SMOTE oversampling.

6.3.2 LSTM Classifier

The LSTM classifier achieves 92.44% accuracy and a weighted F1-score of 0.9237. While its overall metrics are lower than the RF, the per-class analysis reveals important complementary strengths. The LSTM detects Class 2 (spurious DHSV closure) with F1 = 0.7603 through temporal pattern recognition, and achieves F1 = 0.8265 on Class 3 (severe slugging) an oscillatory anomaly whose temporal signature is naturally suited to recurrent modeling. Notably, the LSTM detects Class 6 (quick restriction in PCK) with F1 = 0.3778 and Class 8 (hydrate) with F1 = 0.1935, classes that the RF completely misses. These complementary detection capabilities are precisely what the stacking ensemble exploits

6.3.3 Transformer Classifier

The Transformer achieves 93.02% accuracy and a weighted F1 Score of 0.9295, outperforming the LSTM by approximately 0.5 percentage points. More significantly, it detects the broadest range of anomaly classes among the individual supervised models. It achieves F1 = 0.6970 on Class 2, F1 = 0.8567 on Class 3, F1 = 0.4134 on Class 5 (rapid productivity loss), F1 = 0.3757 on Class 6, and most notably F1 = 0.6471 on Class 8 (hydrate in production line) far surpassing all other individual models on this critical fault type. The Transformers global attention mechanism appears particularly effective at recognizing the gradual, diffuse patterns characteristic of hydrate formation and productivity decline.

6.3.4 LSTM Autoencoder

The LSTM Autoencoder performs binary classification (Normal vs. Anomaly) and achieves 86.07% accuracy. For the Normal class, it achieves a precision of 0.8466, a recall of 0.9293, and an F1 = 0.8860. For the Anomaly class: precision = 0.8857, recall = 0.7650, and F1 = 0.8209. The adaptive threshold optimization, which searches percentiles of the validation reconstruction error distribution rather than using a fixed mean+3 cutoff, calibrates the decision boundary to maximize the weighted F1-score. The key strength of this approach is that it requires no labeled anomaly data during training, making it suitable for deployment in wells with incomplete historical fault records.

6.3.5 Stacking Ensemble

The Stacking Ensemble achieves 96.25% accuracy, a weighted F1-score of 0.9622, and the highest ROC-AUC of 0.9962. Its per-class analysis confirms that the meta-learner successfully combines the complementary strengths of the three base models. On the majority classes, it performs on par with RF (Class 0: F1 = 0.9717, Class 4: F1 = 0.9812). On moderate minority classes, it matches or exceeds the best individual model: Class 3 (severe slugging), F1 = 0.9170, Class 5 (productivity loss), F1 = 0.6595. Most importantly, the Ensemble detects Class 1 (BSW increase) with F1 = 0.5114 and Class 6 (quick restriction) with F1 = 0.4295, classes where individual models either fail or achieve only marginal detection. The ensemble also detects Class 8 (hydrate) with F1 = 0.1935, leveraging the Transformers ability to recognize this gradual fault pattern. From the Figure 6.3, the confusion matrix of Stacking Ensemble Figure 6.4, RF Figure 6.5, and the difference between these two is shown in Figure 6.6.

6.4 Ensemble Diversity Analysis

The effectiveness of a stacking ensemble depends critically on the diversity of its base models. If all models make the same errors, combining them provides no benefit. Figures 6.7a, 6.7b, and 6.7c quantify this diversity using pairwise disagreement rates, Yules Q statistics [42], and prediction correlations among the three base models. The analysis confirms substantial diversity. The RF and LSTM exhibit different error patterns from the Transformer, particularly on minority classes, where the Transformer's attention mechanism detects patterns that tree-based and recurrent models miss. This principled diversity, resulting from fundamentally different inductive biases rather than random perturbations, enables the XGBoost meta-learner to achieve performance that exceeds that of any individual model.

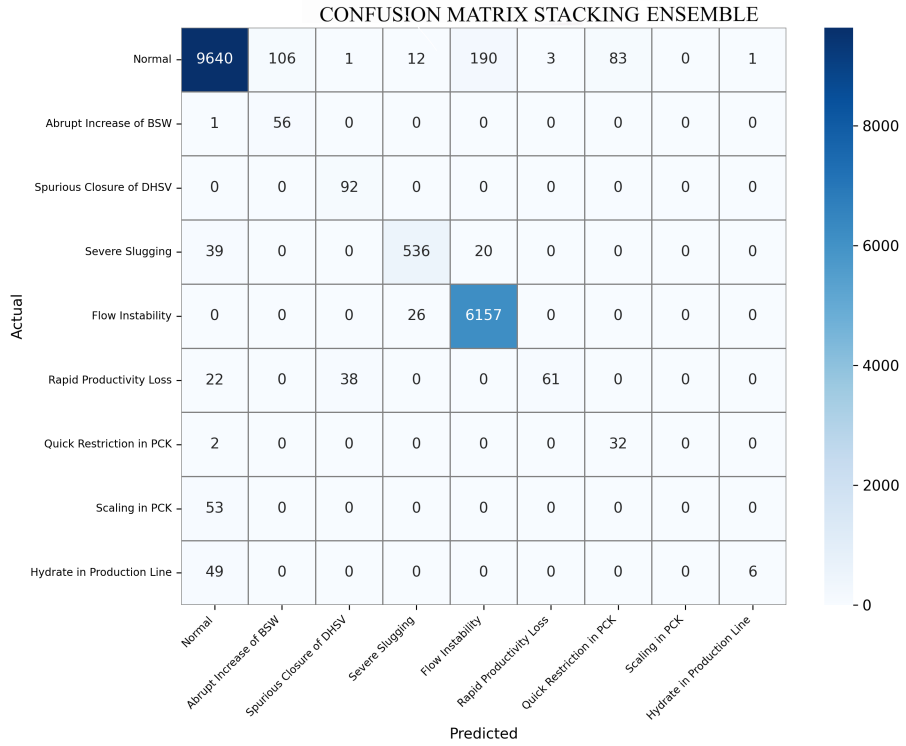


Figure 6.3: Confusion matrix for the Advanced Stacking Ensemble, showing the distribution of predictions across all nine classes.

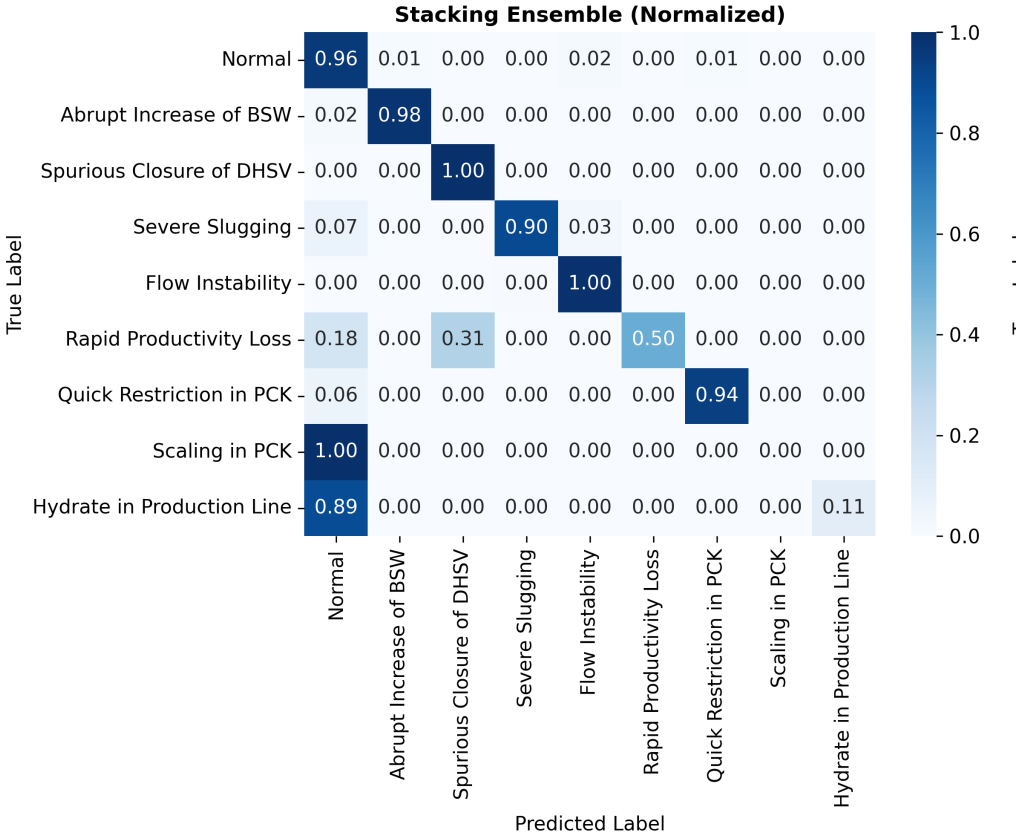


Figure 6.4: Confusion matrix for the Advanced Stacking Ensemble(Normalized).

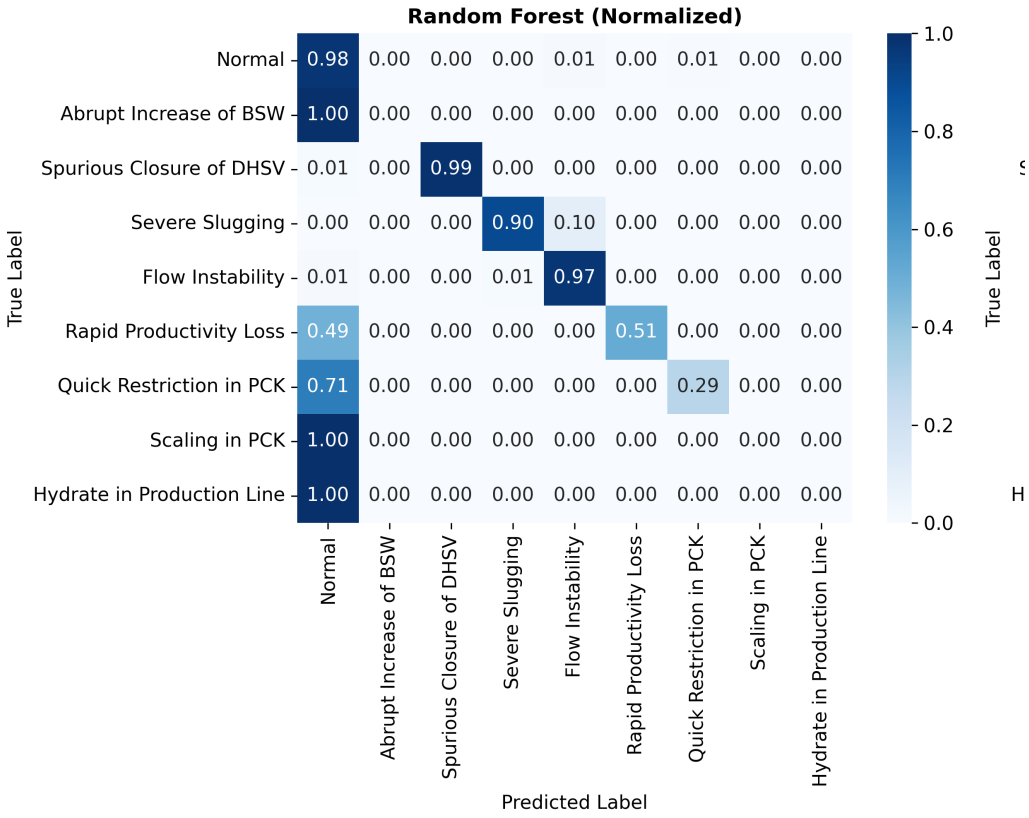


Figure 6.5: Confusion matrix for the RF (Normalized).

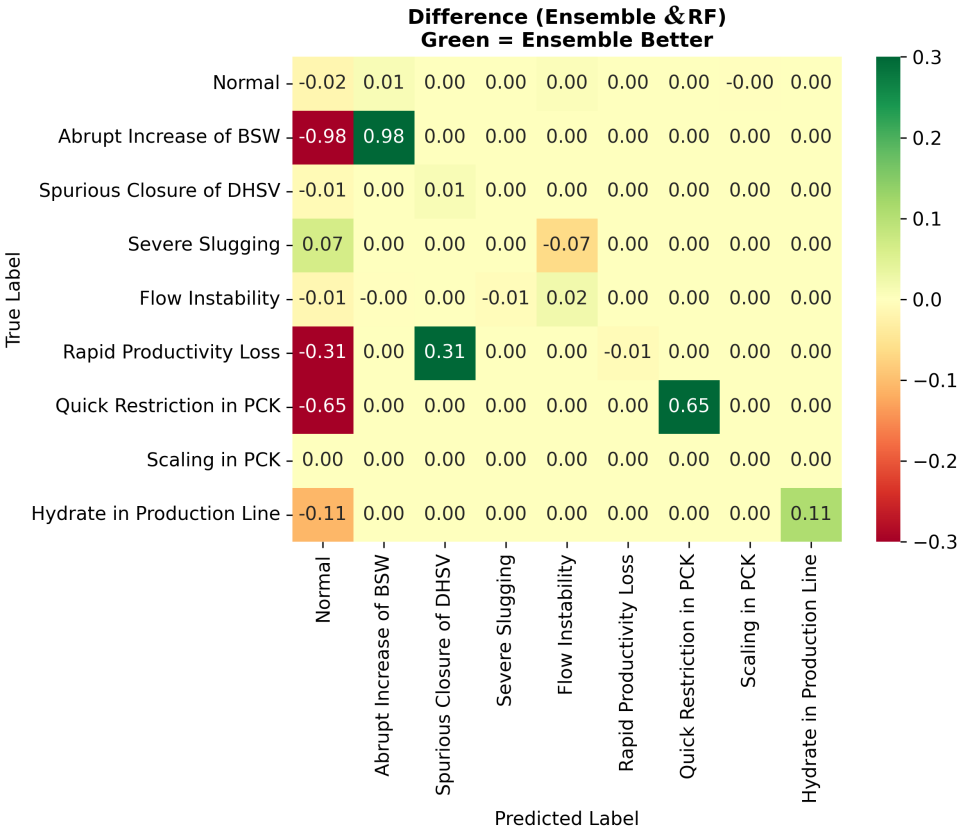
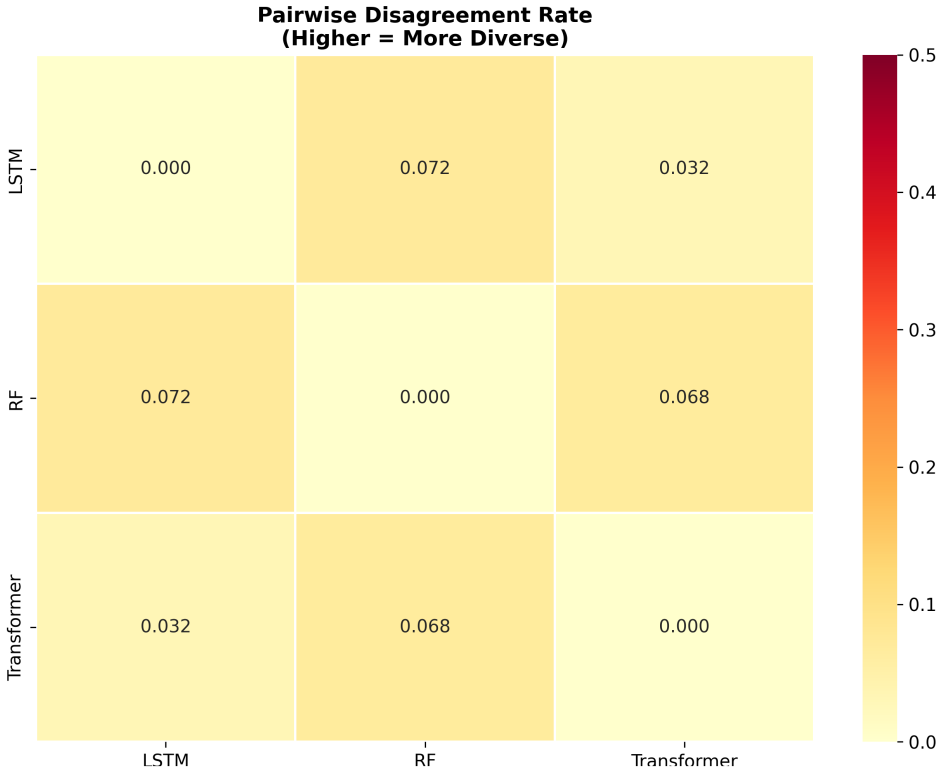
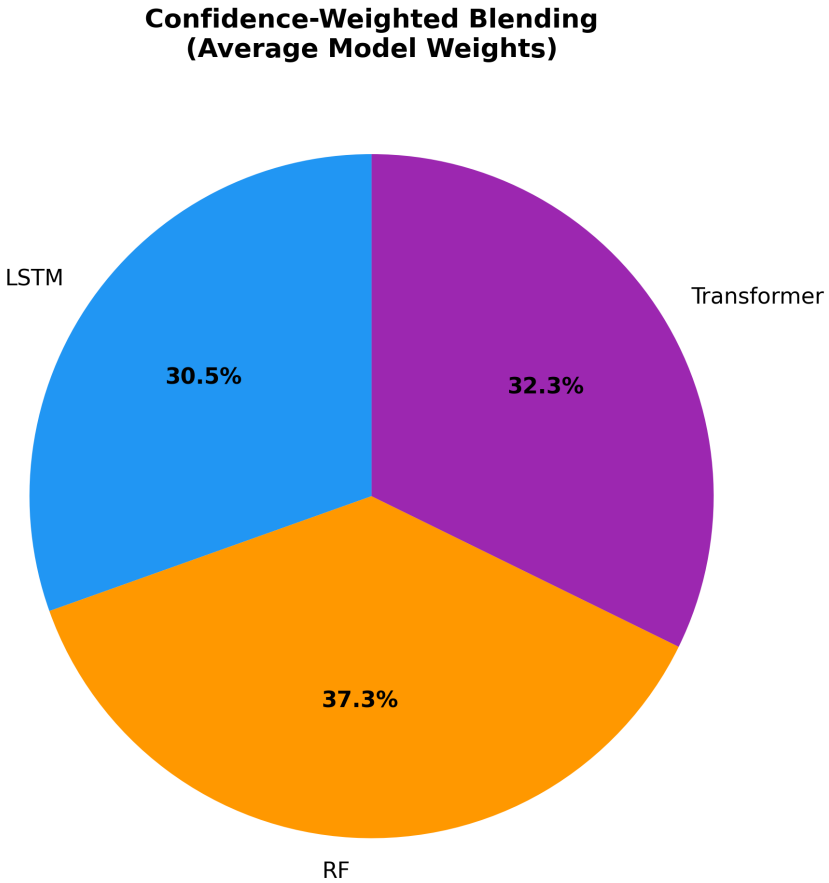


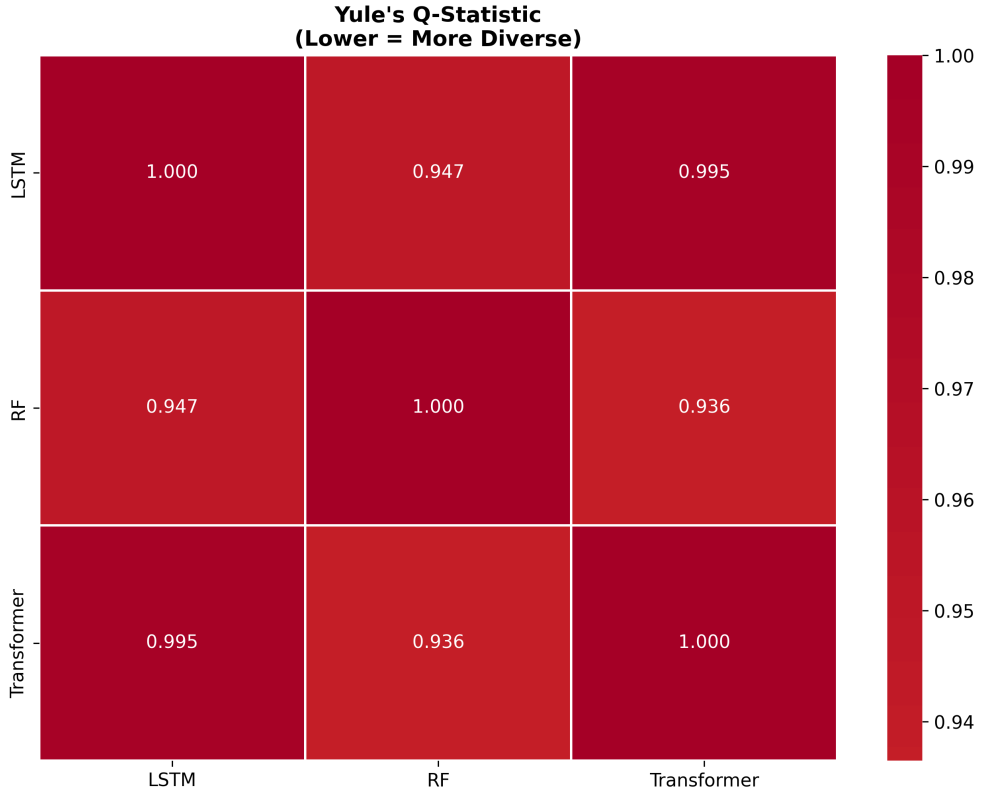
Figure 6.6: Confusion matrix difference (Stacking Ensemble and RF).



(a) Pairwise disagreement rate.



(b) Confidence weighted blending.



(c) Yule's Q statistic.

Figure 6.7: Diversity analysis of the three base models, confirming complementary error patterns that the meta-learner exploits.

6.5 Training Dynamics

The training curves for the deep learning models, exemplified by the Transformer in Figure 6.8, reveal smooth convergence for all models. Early stopping consistently triggers before the maximum epoch limit of 50, indicating effective regularization through the combination of dropout, weight decay, and the early stopping mechanism itself. The Transformer shows slightly more variance in its validation loss than the LSTM, which is consistent with the known sensitivity of attention mechanisms to small dataset sizesthe attention weights have more parameters to fit, and hence more potential for overfitting on limited data.

6.6 Explainability Analysis

6.6.1 SHAP Feature Importance

SHAP is applied to the RF model using the TreeExplainer to compute Shapley values that reveal which sensor features drive the models predictions.

Figure 6.9 shows the collected SHAP importance through all classes. SHAP values for each base feature across all time-step positions in the window are very important and calculated precisely. Pressure features (P-PDG, P-TPT, P-MON-CKP) are always the most important for all anomaly classes. Temperature features (T-TPT, T-JUS-CKP) are particularly important for detecting thermally related anomalies such as hydrate formation. The gas lift flow rate (QGL) is crucial for flow-related anomalies such as slugging

and flow instability. Importantly, Statistical features, including rolling mean and standard deviation, really play an important role in validating the feature engineering progress in Chapter 3.

6.6.2 Transformer Attention Visualization

The Attention weights are extracted from the Transformers self-attention layers to provide temporal interpretability, revealing which time steps the model focuses on when making each prediction. Figure 6.10 shows the difference in the attention patterns for normal operation against anomalous samples. In normal operation, without a strong focus on any particular region, attention is relatively evenly distributed across the temporal window. On the other hand, anomalous samples show strong attention peaks at specific time steps, which align with the distinctive signature of the fault condition. Different anomaly types produce well-defined attention profiles, abrupt events, such as spurious DHSV closure, tend to show concentrated attention on specific time steps. Multi-head diversity suggests different task specialization in recognizing different temporal patterns. This temporal interpretability, a combination of SHAP (which sensors matter) and attention analysis (when in the sequence they matter), offers a comprehensive interpretability framework that is stronger than either approach used alone.

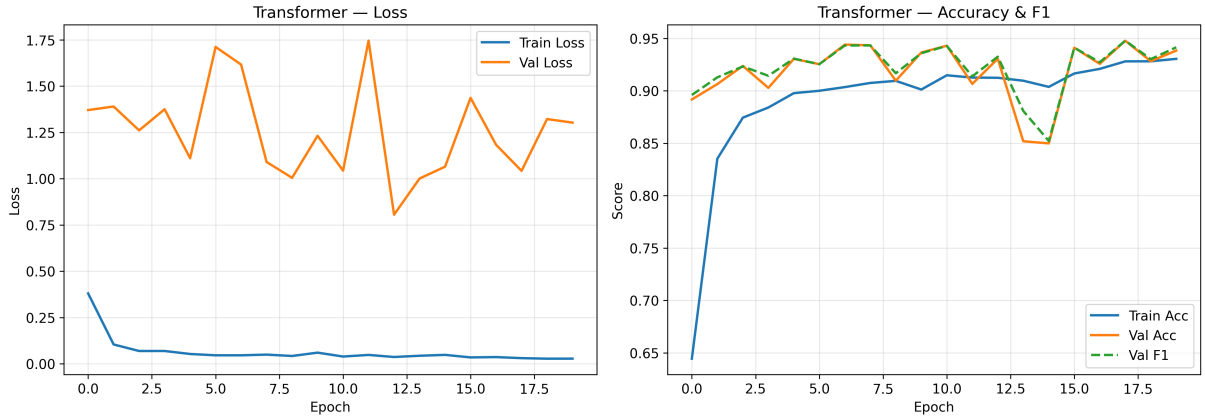


Figure 6.8: Transformer training and validation loss/accuracy curves, showing smooth convergence with early stopping and learning rate reduction.

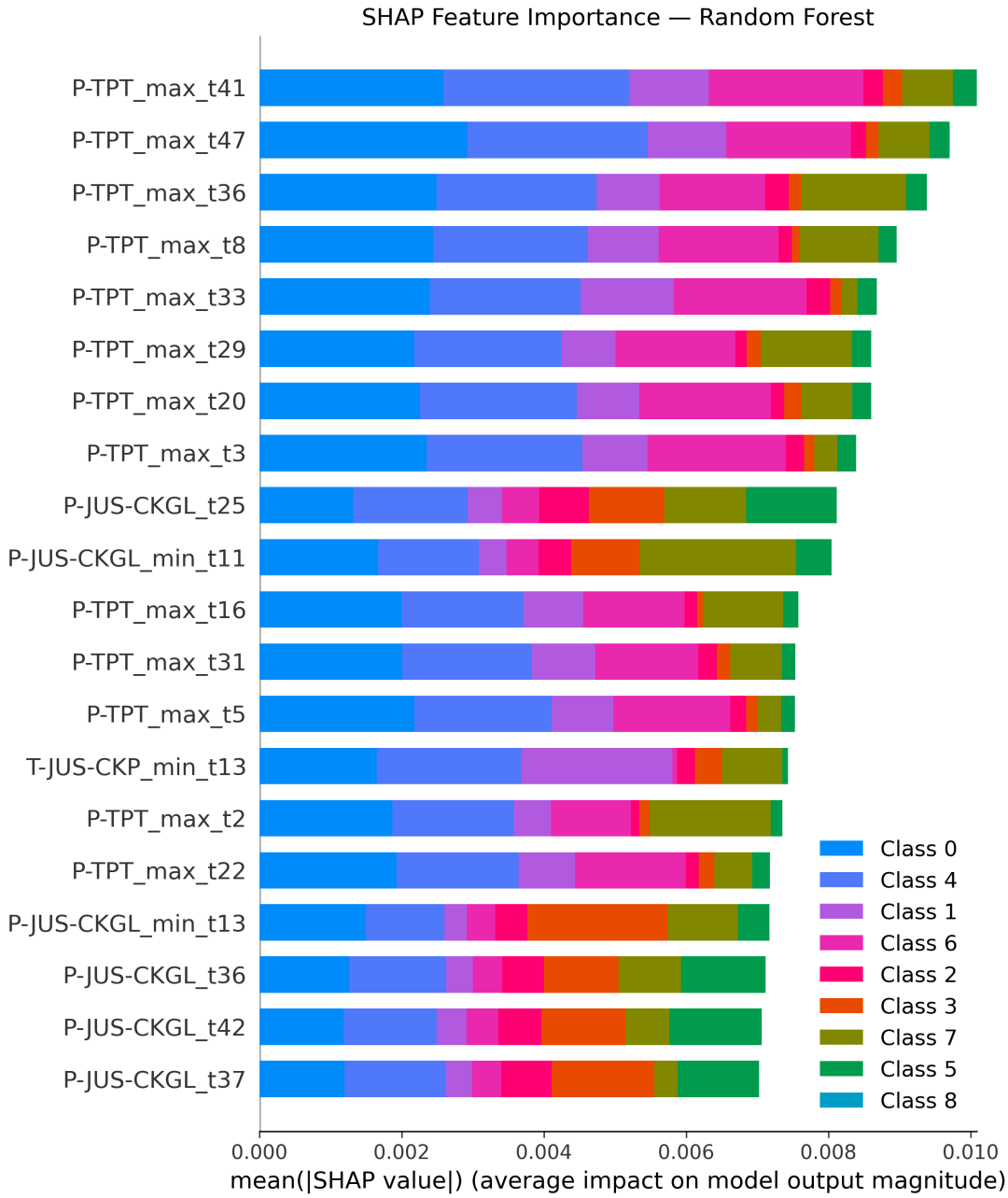


Figure 6.9: SHAP summary plot showing how each feature's value influences model predictions.

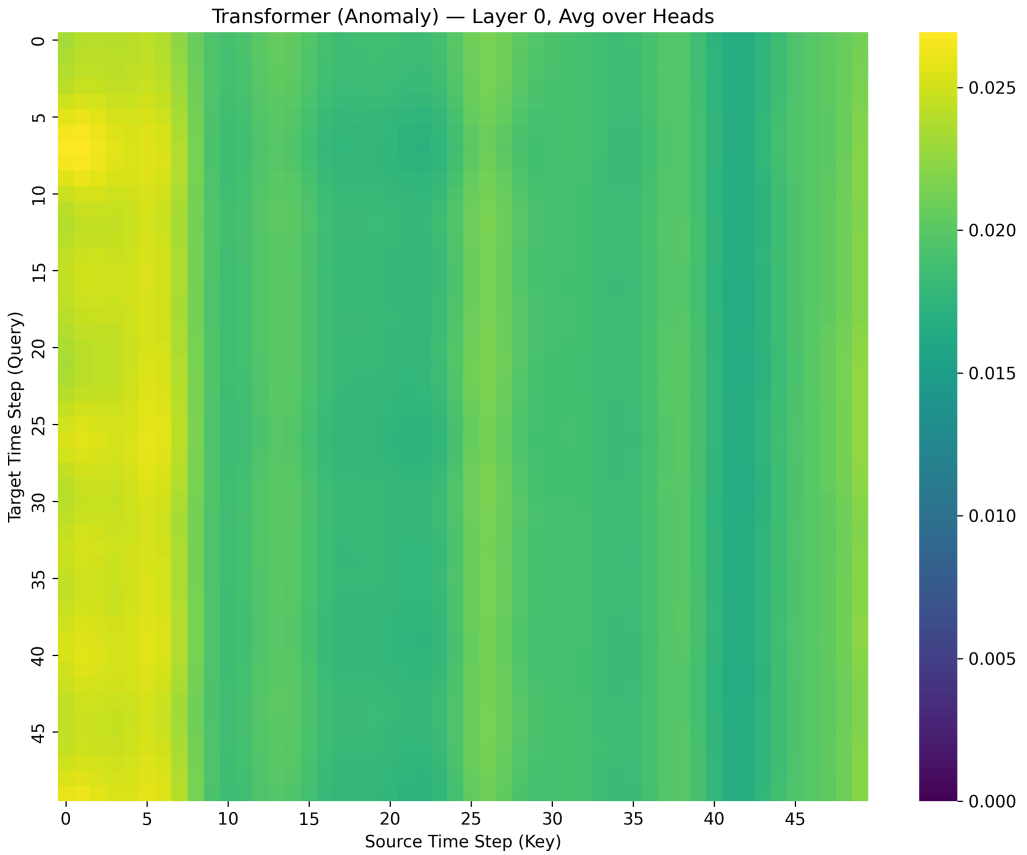


Figure 6.10: Attention heatmap for anomaly samples from the first Transformer encoder layer.

6.7 Precision-Recall Analysis

The precision-recall analysis shows a new understanding of the model performance under diverse classification margins in Figure 6.11. Thus, the majority classes in all supervised models continue to achieve high precision and recall by wide margins, confirming strong detection. On the other hand, the minority classes show curves that reveal the essential precision-recall trade-off, such as the Transformer attaining the best balance for rare classes (Class 8). At the same time, RF achieves higher precision but very low recall for

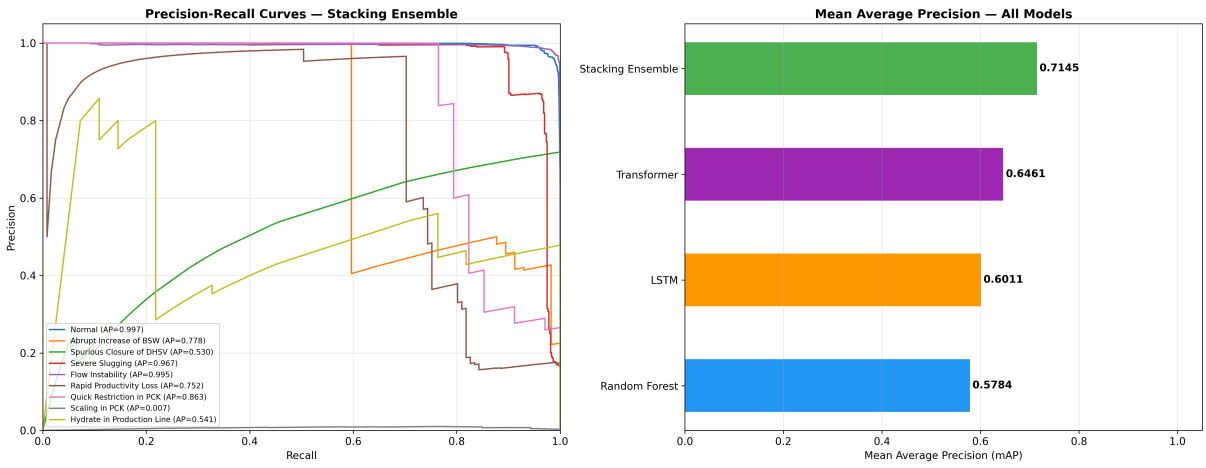


Figure 6.11: Precision-Recall curves for all supervised models.

these same classes. Besides, the Stacking Ensembles curves perform better than all these individual models, especially within the high-recall regime. This is most important for safety-critical anomaly detection, since missing a true fault is a more serious fault than generating a false alarm.

7 Discussion of Key Findings

7.1 Comparative Model Performance Analysis

The results show that the complex model does not always give the best result. RF is simple, but it has achieved the highly competitive accuracy of 95.98%. Also with F1-weighted=0.9558. It shows that classical feature engineering remains highly competitive. The Transformer classifier (F1-weighted: 0.9295) did better than the LSTM (F1-weighted: 0.9237). It shows that self-attention mechanisms are more effective than sequential processing. The Transformer's advantage is likely due to its ability to attend to all 50 timesteps simultaneously. This enables the model to detect LSTM events without the vanishing gradient problem that constrains bidirectional LSTMs. Both deep learning models underperformed the RF baseline. This indicates that the 8-sensor, 50-step feature space is not enough to fully use deep architectures. The LSTM Autoencoder achieved an accuracy of 86.07%. It's performing comparably to supervised models despite receiving no labeled anomaly data during training. This confirms the autoencoder's utility for real-world deployment, where new well types may present anomaly signatures not in the training distribution. The autoencoder is limited to binary classification (normal vs. any anomaly). This makes it unsuitable for fault diagnosis.

7.2 The Stacking Ensemble Bringing Models Together

The Advanced Stacking Ensemble achieved the highest performance (F1-weighted: 0.9622, AUC: 0.9962, Accuracy: 96.25%). The Ensemble outperforms all individual models in the reported test metrics. This supports the idea that combining tree-based, recurrent, and attention-based learners through principled meta-learning gives better predictions. The ensemble's meta-learner receives 55 engineered features that represent base-model predictions, confidence scores, and derived statistics. It allows XGBoost to learn second-order patterns in how the base models succeed and fail. The ensemble demonstrated exceptional strength on Class 1 (Abrupt Increase of BSW). It achieved an F1-score of 0.5114. This is important because no individual base model could reliably detect Class 1. But the ensemble's meta-learning mechanism successfully learned to recognize the combination of base-model predictions and disagreement. This finding suggests that robust anomaly detection in petroleum wells may be inherently a multi-model problem. Individual sensors or modeling paradigms are insufficient, but their combination gives rise to predictive power. The ensemble, like all models, completely failed on Class 7 (Scaling), with F1 = 0.0. This systematic failure across all five models, all hyperparameter configurations, and all training procedures points to a basic restriction. The 50-second observation window is insufficient to capture the temporal signature of scaling, a slow mineral precipitation process that evolves over hours to days.

7.3 Impact of Focal Loss and Adaptive Thresholding

The implementation of Focal Loss (gamma=2.0) in the LSTM and Transformer classifiers improved their ability to learn from hard samples. By down-weighting well-classified samples via the factor $(1-p)^{\gamma}$, Focal Loss forced the models to allocate training effort toward rare, ambiguous anomaly classes rather than optimizing for the majority class (Class 0: 90%+ of data). The adaptive threshold optimization for the LSTM Autoencoder similarly demonstrated the value of data-driven hyperparameter tuning. Rather than applying the traditional approach of threshold = (mean + 3*std), the pipeline searches percentiles on the validation set to find the threshold that maximizes F1-score.

7.4 Data Leakage Prevention via File-Level Splitting

Many published works on the 3W dataset split the data after creating overlapping windows. This causes temporal contamination, in which the train and test sets contain overlapping windows from the same well file. This increases the reported scores. This report uses file-level splitting (69.7% train, 15.1% validation, 15.2% test). This ensures that all windows from a given well recording belong to a single partition. The resulting accuracy and F1 scores are lower than those reported by simple post-split methods. But they represent honest performance estimates for deployment on unseen well recordings.

7.5 Statistical Feature Engineering Effectiveness

The enrichment from 8 raw sensor readings to 48 derived features (mean, std, min, max, rate of change per sensor) significantly improved RF performance. Also provided the feature basis for the stacking ensemble's meta-learner. This result shows that domain-based feature engineering (such as time-based averages, variability, and trends within each window) is as effective as, or even better than, larger, deeper models. If anomalies show as changes in mean, variance, or trends observable within 50 seconds, then engineered statistics suffice. If anomalies show differently (as ultra-high-frequency vibration signatures only resolvable at 1 kHz sampling), then the current 1 Hz sensor data is fundamentally limited.

7.6 Implications for Operational Deployment

The RF and ensemble models are suitable for real-time monitoring on production hardware. RF inference is typically computationally lighter than LSTM or Transformer inference, although deployment-time benchmarking is needed. They are well-suited as decision-support tools that alert human operators to potential faults and highlight which sensors are driving the alarm. Explainability methods enable operators to assess a model's validity before acting on its recommendations. The complete failure in Class 7 and moderate performance in Classes 1, 3 to 6 suggest that real-world production environments will encounter anomaly types not well represented in historical data. The LSTM Autoencoder provides a fallback mechanism. When the autoencoder flags an anomaly that the supervised classifier refuses to categorize, this disagreement can itself serve as an alert to a genuinely novel fault type requiring expert investigation.

7.7 Generalization and Robustness

The study is limited to eight well recordings from the Petrobras 3W dataset, all from the same geographic region and well-architected. Sensor failures, drift, miscalibration, and environmental variations specific to other wells risk degrading model performance. Within the 3W dataset, the models show stability across cross-validation folds, indicating low performance variance. The final models consistently fail on Class 7, suggesting that the current setup may be insufficient for this slow-evolving anomaly. Future validation using independent well data, along with exploration of transfer learning and domain adaptation techniques, is essential to enable robust cross-well generalization.

placeins

8 Limitations

This study is primarily limited by its reliance on a small dataset of eight well recordings from a single offshore platform in Brazil. This creates risks for how well the results apply to other settings and how well the model generalizes. All wells share the same geology, design, and Petrobras operating procedures, so other oil fields may exhibit very different sensor behavior and fault types. The data also covers only limited time periods per well and uses 1 Hz sampling, which is insufficient to capture slow-changing processes (such as Class 7 scaling) or very fast events. With only 8 sensors instead of the 50+ used in modern wells, important fault signals may be missing. The dataset is also highly imbalanced, with normal operation comprising most of the data, making it harder to detect rare faults even with methods like SMOTE and Focal Loss. RF does not work with time order, LSTM struggles with small capacity, and Transformers need more data. The stacking model achieves high accuracy but is hard to interpret. Cross-validation was used, but no testing was done on completely new wells. So real-world performance is uncertain. No transfer learning or deployment testing was conducted, and differences in sensor behavior between wells were not studied. Explainability methods show relations but not true causes. The model performs well on the dataset but requires real-world validation, broader data coverage, and careful monitoring before deployment.

9 Conclusion and Future Work

9.1 Conclusion

This report introduced a comparative study of machine learning for anomaly detection in petroleum production wells using the Petrobras 3W dataset. Five model architectures were developed and evaluated under standardized conditions: RF, LSTM, LSTM Autoencoder, Transformer, and a novel Advanced Stacking Ensemble.

The RF classifier demonstrated the accuracy at 95.98%, indicating that traditional Machine learning, when paired with proper feature engineering, remains highly effective and competitive. The Stacking Ensemble, which combines RF, LSTM, and Transformer through cross-validated out-of-fold stacking, meta-feature engineering, and with an XGBoost meta-learner, achieved the highest ROC-AUC of 0.9962 and a weighted F1-score of 0.9622. The Transformer showed the strongest individual performance on Class 8 (Hydrate in Production Line), while the ensemble improved overall weighted performance. . Both the Transformer and LSTM models showed strong performance, especially with the advantage of temporal modeling. The LSTM Autoencoder delivered 86.07% accuracy for binary anomaly detection without needing any labeled anomaly data.

Key methodological contributions include:

1. File-level data splitting to prevent temporal leakage.
2. Multi-model stacking with meta-feature engineering.
3. Focal Loss for handling extreme class imbalance.
4. combined SHAP and attention-based interpretability.

These contributions demonstrate a precise benchmark framework for future research on the 3W dataset.

9.2 Future Work

There is promising future work to study and implement from this report. The first is Real-time deployment, which enables models to run quickly on production platforms, even with limited resources. Secondly, few-shot learning to develop methods for detecting novel anomaly types using limited labeled examples, addressing the long-tail distribution of rare faults. To ensure they remain reliable, tests should be conducted to assess how well trained models handle noisy sensor data and cyberattacks [43]. Multi-well transfer learning looks into ways to adapt models trained on one well to new wells that operate differently. Federated learning to explore privacy-preserving methods for collaborative anomaly detection across multiple operators [25, 44]. Also, advanced architectures to evaluate new approaches such as Temporal Convolutional Networks (TCN), TranAD, and state-space

models (Mamba) for improved temporal modeling. Finally, a hierarchical anomaly taxonomy model is used to first detect anomalies (binary) and then categorize them into specific fault types, potentially combining the LSTM Autoencoder with supervised classifiers.

placeins

Bibliography

[1] R. V. Vargas, B. M. F. Lucas, A. A. Coutinho, and R. M. S. Werneck, "A realistic and public dataset with rare undesirable real events in oil wells," *Journal of Petroleum Science and Engineering*, vol. 181, 2019.

[2] M. R. Shaik and S. S. Rao, "Prediction of oil and gas pipeline failures through machine learning approaches: A systematic review," *Energy Reports*, vol. 9, pp. 11504–11526, 2023.

[3] N. Aslam, I. U. Khan, A. Alansari, M. Alrammah, A. Alghwairy, R. Alqahtani, R. Alqahtani, M. Almushikes, and M. A. Hashim, "Anomaly detection using explainable random forest for the prediction of undesirable events in oil wells," *Applied Computational Intelligence and Soft Computing*, vol. 2022, no. 1, p. 1558381, 2022.

[4] W. Fernandes Jr, K. S. Komati, and K. Assis de Souza Gazolli, "Anomaly detection in oil-producing wells: a comparative study of one-class classifiers in a multivariate time series dataset," *Journal of Petroleum Exploration and Production Technology*, vol. 14, no. 1, pp. 343–363, 2024.

[5] F. A. Silva, R. M. Filho, and M. B. Costa, "Anomaly detection using explainable random forest for the prediction of undesirable events in oil wells," *ResearchGate*, 2022.

[6] W. Wang, Y. Zhao, and Y. Li, "Ensemble machine learning for predicting the homogenized elastic properties of unidirectional composites: A shap-based interpretability analysis," *Acta Mechanica Sinica*, vol. 40, no. 3, p. 423301, 2024.

[7] M. Altalhan, A. Algarni, and M. T.-H. Alouane, "Imbalanced data problem in machine learning: A review," *IEEE access*, vol. 13, pp. 13686–13699, 2025.

[8] D. Soydaner, "Attention mechanism in neural networks: where it comes and where it goes," *Neural Computing and Applications*, vol. 34, no. 16, pp. 13371–13385, 2022.

[9] S. Jain and B. C. Wallace, "Attention is not explanation," *CoRR*, vol. abs/1902.10186, 2019.

[10] F. Asghar and M. Khan, "An anomaly detection model for oil and gas pipelines using machine learning," *Computation*, vol. 10, no. 8, p. 138, 2022.

[11] S. Sarker *et al.*, "Deeppipe: A deep-learning method for anomaly detection of multi-product pipelines," *Energy*, vol. 259, 2022.

[12] M. Uddin *et al.*, "Anomaly detection in a smart industrial machinery plant using IoT and machine learning," *Sensors*, vol. 23, no. 19, p. 8286, 2023.

[13] A. M. Santos and R. M. Filho, "Oil and gas flow anomaly detection on offshore naturally flowing wells using deep neural networks," *Petroleum Research*, vol. 9, no. 4, 2024.

[14] L. Oliveira, R. Lopes, and M. Costa, "Anomaly detection in oil-producing wells: A comparative study of one-class classifiers in a multivariate time series dataset," *Journal of Petroleum Exploration and Production Technology*, vol. 13, 2023.

[15] T. Santos and F. Silva, "Unsupervised machine learning model for predicting anomalies in subsurface safety valves and application in offshore wells during oil production," *Journal of Petroleum Exploration and Production Technology*, vol. 13, 2023.

[16] E. Lima-Junior and R. M. Filho, "Real-time event detection with random forests and temporal convolutional networks for more sustainable petroleum industry," *arXiv preprint arXiv:2310.08737*, 2023.

[17] M. Al-Rubaii and R. Khan, "Stuck pipe detection in oil and gas drilling operations using deep learning autoencoder for anomaly diagnosis," *Applied Sciences*, vol. 15, no. 9, 2025.

[18] A. Perevalov *et al.*, "Application of machine learning to accidents detection at directional drilling," *Petroleum Science*, 2025.

[19] S. S. Aljameel, D. M. Alomari, S. Alismail, F. Khawaher, A. A. Alkhudhair, F. Aljubran, and R. M. Alzannan, "An anomaly detection model for oil and gas pipelines using machine learning," *Computation*, vol. 10, no. 8, p. 138, 2022.

[20] S. Papadopoulos *et al.*, "A combined semi-supervised deep learning method for oil leak detection in pipelines using IIoT at the edge," *Sensors*, vol. 22, no. 11, p. 4105, 2022.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[22] E. Lima-Junior, "Deep transformer networks applied to anomaly detection in oil and gas wells," in *Offshore Technology Conference*, 2025.

[23] Y. Zhang and L. Wang, "EfficientTransformer: A dynamic anomaly detection model for industrial control networks," *IEEE Transactions*, 2025.

[24] H. Lin *et al.*, "Transformer anomaly detection based on multi-source information fusion of insulation oil," *Journal of Computers*, vol. 36, no. 3, 2025.

[25] L. Friha *et al.*, "Federated learning-based explainable anomaly detection for industrial control systems," *IEEE Access*, vol. 10, 2022.

[26] Z. Liu and W. Chen, "Gas turbine combustor anomaly detection using stacked denoising autoencoder and extreme learning machine," *Applied Sciences*, vol. 12, 2022.

[27] S. Jain and B. C. Wallace, "Attention is not Explanation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (J. Burstein, C. Doran, and T. Solorio, eds.), (Minneapolis, Minnesota), pp. 3543–3556, Association for Computational Linguistics, June 2019.

[28] Z. Syed, M. Saputelli, and S. Amr, "Performance evaluation of outlier detection techniques in production timeseries: A systematic review and meta-analysis," *Expert Systems with Applications*, vol. 191, 2022.

[29] A. B. Nassif, M. A. Talib, Q. Nasir, and F. M. Dakalbab, "Deep reinforcement learning for anomaly detection: A systematic review," *IEEE Access*, vol. 10, 2022.

[30] P. Zhu *et al.*, "A novel machine learning approach for detecting outliers, rebuilding well logs, and enhancing reservoir characterization," *Natural Resources Research*, vol. 32, 2023.

[31] S. Kapoor and A. Narayanan, "Leakage and the reproducibility crisis in machine-learning-based science," *Patterns*, vol. 4, no. 9, p. 100804, 2023.

[32] M. M. Ahsan, M. P. Mahmud, P. K. Saha, K. D. Gupta, and Z. Siddique, "Effect of data scaling methods on machine learning algorithms and model performance," *Technologies*, vol. 9, no. 3, p. 52, 2021.

[33] L. Martí, N. Sanchez-Pi, J. M. Molina, and A. C. B. Garcia, "Anomaly detection based on sensor data in petroleum industry applications," *Sensors*, vol. 15, no. 2, pp. 2774–2797, 2015.

[34] K. Tao, J. Cao, Y. Wang, J. Mi, W. Ma, and C. Shi, "Chemometric classification of crude oils in complex petroleum systems using t-distributed stochastic neighbor embedding machine learning algorithm," *Energy & Fuels*, vol. 34, no. 5, pp. 5884–5899, 2020.

[35] Z. Zamanzadeh Darban, G. I. Webb, S. Pan, C. Aggarwal, and M. Salehi, "Deep learning for time series anomaly detection: A survey," *ACM Computing Surveys*, vol. 57, no. 1, pp. 1–42, 2024.

[36] P. Mobtahej, X. Zhang, M. Hamidi, and J. Zhang, "An lstm-autoencoder architecture for anomaly detection applied on compressors audio data," *Computational and Mathematical Methods*, vol. 2022, no. 1, p. 3622426, 2022.

[37] W. Wang, Z. Xu, R. Tang, S. Li, and W. Wu, "Fault detection and diagnosis for gas turbines based on a kernelized information entropy model," *The Scientific World Journal*, vol. 2014, no. 1, p. 617162, 2014.

[38] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, (New York, NY, USA), p. 785794, Association for Computing Machinery, 2016.

[39] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980–2988, 2017.

[40] M. R. Fachrezi, A. F. Ihsan, and W. Astuti, "Anomaly detection using lstm-based deep learning on natural gas pipeline operational data," in *2024 12th International Conference on Information and Communication Technology (ICoICT)*, pp. 500–506, IEEE, 2024.

[41] E. Luna-Villagómez and V. Mahalec, "Exploring kolmogorov–arnold networks for unsupervised anomaly detection in industrial processes," *Processes*, vol. 13, no. 11, p. 3672, 2025.

[42] B. P. Lloyd, C. H. Kennedy, and P. J. Yoder, "Quantifying contingent relations from direct observation data: Transitional probability comparisons versus yule's q," *Journal of Applied Behavior Analysis*, vol. 46, no. 2, pp. 479–497, 2013.

[43] D. Han, Z. Wang, Y. Zhong, W. Chen, J. Yang, S. Lu, X. Shi, and X. Yin, "Evaluating and improving adversarial robustness of machine learning-based network intrusion detectors," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2632–2647, 2021.

[44] A. El-Hoshoudy, "Application of artificial intelligence and federated learning in petroleum processing," in *AI in Petroleum Processing*, Taylor & Francis, 2024.