HANOI UNIVERSITY
FACULTY OF INFORMATION TECHNOLOGY

Project 2 Report:

GRU-Based Predictive Maintenance for Industrial Pump Systems

Group: 29

Members:

2201140028	**Nguyen Trung Hieu**
2201140043	**Tran Nguyen Khai**
2201140002	**Dao Viet Anh**

Lecturer: Nguyen Xuan Thang

Course: 62FIT4ATI

Semester: Fall 2025

Contents

1 Introduction **2**

2 Model and Setup **2**
2.1 Problem Formulation and Data Characteristics 2
2.2 Choice of Recurrent Architecture 2
2.3 Dual-Model Strategy: Why Two Models? 3
2.4 Network Architecture and Training Configuration 3

3 Results **4**
3.1 Quantitative Performance Comparison 4
3.2 Per-Class Performance 4
3.3 ROC Curve Analysis 5
3.4 Confusion Matrix Analysis 5

4 Discussion **6**
4.1 Impact of the Dual-Model Approach 6
4.2 Threshold Selection for Deployment 6
4.3 Limitations 6

5 Conclusion **7**

1 Introduction

This project addresses predictive maintenance for industrial pump systems using sensor data. The objective is to classify machine state as NORMAL, BROKEN, or RECOVERING based on readings from 52 sensors (220,320 samples).

The primary challenge is **extreme class imbalance**: only 7 BROKEN samples (0.003%). We employ a dual-model strategy using GRU: a 3-class classifier and a binary classifier (NORMAL vs ANOMALY). This report analyzes and compares both approaches.

2 Model and Setup

2.1 Problem Formulation and Data Characteristics

The task is multivariate time-series classification using sliding windows of 20 timesteps, with labels corresponding to the machine state at the final timestep.

Table 1: Dataset Class Distribution

Status	**Count**	**Percentage**
NORMAL	219,420	99.59%
RECOVERING	893	0.41%
BROKEN	7	0.003%

The dataset exhibits extreme class imbalance characteristic of real-world predictive maintenance. During preprocessing, `sensor_04` was removed due to data leakage (-0.916 correlation with target). Analysis of BROKEN events revealed that sensors deviate 10-20 minutes before failure, confirming early warning potential.

2.2 Choice of Recurrent Architecture

A GRU-based architecture is selected for this task. Compared to LSTM, GRU offers several advantages:

- **Fewer parameters**: GRU has 2 gates versus LSTM's 3 gates, reducing model complexity
- **Faster training**: Simpler architecture leads to reduced computational cost
- **Less prone to overfitting**: Critical when training data for minority classes is extremely limited (only 7 BROKEN samples)
- **Comparable performance**: For many sequence modeling tasks, GRU achieves similar results to LSTM

Given the severe data limitation for failure classes, a simpler architecture that generalizes well is preferred over a more complex model that may overfit.

2.3 Dual-Model Strategy: Why Two Models?

We developed two complementary models to address the classification task:

Model 1: 3-Class Classifier (NORMAL / RECOVERING / BROKEN)

- Follows the original problem formulation directly
- Attempts to distinguish between all three machine states
- Challenge: Only 7 BROKEN samples in entire dataset (4 in training)

Model 2: Binary Classifier (NORMAL vs ANOMALY)

- Combines BROKEN and RECOVERING into single ANOMALY class
- Increases minority class samples from 7 to 900
- Rationale: In practice, detecting *any* anomaly is more actionable than distinguishing failure types
- Aligns with real-world maintenance needs: operators need to know "is something wrong?" rather than "what exactly is wrong?"

This dual approach allows us to evaluate both the theoretical (3-class) and practical (binary) solutions, demonstrating the trade-offs involved in handling extreme class imbalance.

2.4 Network Architecture and Training Configuration

Both models share the same base architecture:

Table 2: GRU Network Architecture

Layer	**Configuration**	**Parameters**
Input	(20 timesteps, 50 features)	–
GRU	32 units	8,160
Dropout	Rate = 0.3	–
Dense	16 units, ReLU	528
Dropout	Rate = 0.2	–
Output	Softmax (3 or 2 classes)	51 / 34
Total		∼8,700

Training configuration includes:

- **Optimizer**: Adam with learning rate 0.0005
- **Loss**: Sparse categorical cross-entropy with class weights
- **Gradient clipping**: Max norm = 1.0 to prevent exploding gradients
- **Early stopping**: Patience = 5 epochs monitoring validation loss
- **Learning rate reduction**: Factor = 0.5 on plateau

To address class imbalance, we apply **undersampling** (reducing NORMAL samples) combined with **class-weighted loss** (max 10x weight for minority classes). SMOTE was deliberately avoided as it creates synthetic samples through interpolation, which breaks temporal dependencies in time-series data.

3 Results

3.1 Quantitative Performance Comparison

Table 3: Model Performance Metrics on Test Set (24,981 samples)

Metric	**3-Class Model**	**Binary Model**
Accuracy	98.01%	98.03%
Balanced Accuracy	62.46%	**93.11%**
F1 Score (Macro)	40.08%	21.44%
ROC-AUC	N/A	**96.11%**

While both models achieve similar standard accuracy (∼98%), this metric is misleading due to class imbalance—a trivial classifier predicting all samples as NORMAL would achieve 99.6% accuracy. The **balanced accuracy** reveals the true difference: the binary model (93.11%) dramatically outperforms the 3-class model (62.46%). This 30+ percentage point gap demonstrates that combining minority classes into a single ANOMALY category provides sufficient training data for effective learning. The binary model's ROC-AUC of 96.11% further confirms excellent discrimination ability.

3.2 Per-Class Performance

Table 4: Per-Class Performance Comparison

	3-Class			**Binary**		
Class	True	Pred	Recall	True	Pred	Recall
NORMAL	24,905	24,425	98.0%	24,905	24,432	98.1%
RECOVERING	75	556	89.3%	–	–	–
BROKEN	1	0	**0.0%**	–	–	–
ANOMALY	–	–	–	76	549	**88.2%**

The per-class breakdown reveals why the 3-class model struggles: it achieves 0% recall on BROKEN because with only 4 training samples, the model cannot learn meaningful patterns for this class. Instead, it classifies all anomalies as RECOVERING (89.3% recall), effectively ignoring the BROKEN state entirely.

The binary model solves this by merging BROKEN and RECOVERING into ANOMALY, increasing minority class samples from 7 to 900. This enables the model to learn a robust decision boundary, achieving 88.2% anomaly recall (67 of 76 anomalies detected). Both models show high NORMAL recall (∼98%), indicating that class-weighted loss successfully prevents the majority class from dominating predictions.

3.3 ROC Curve Analysis

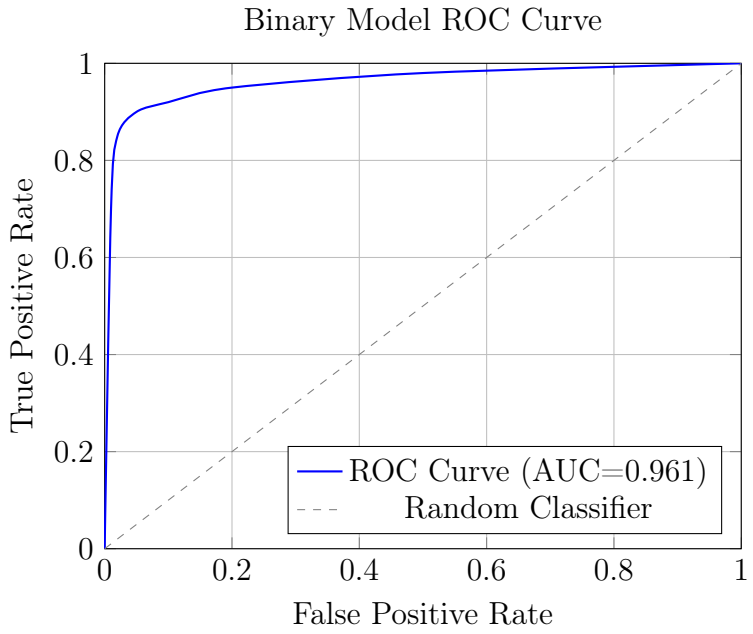

Figure 1: ROC Curve for Binary Classification Model

The ROC-AUC of **0.961** indicates excellent discrimination ability between normal and anomalous states. The model achieves 85% true positive rate at only 2% false positive rate, demonstrating strong separation between class distributions.

3.4 Confusion Matrix Analysis

Table 5: Confusion Matrix: Binary Model

	Pred: NORMAL	**Pred: ANOMALY**
True: NORMAL	24,432	473
True: ANOMALY	9	67

The confusion matrix reveals the precision-recall trade-off inherent in imbalanced classification. The model predicts 549 anomalies when only 76 exist, yielding low precision (12.2%) but high recall (88.2%). Only 9 true anomalies were misclassified as NORMAL.

This trade-off is **acceptable for predictive maintenance**: the cost of 473 unnecessary inspections is far lower than missing 9 potential failures. A missed pump failure can cause production downtime, equipment damage, and safety hazards—costs that typically exceed inspection costs by orders of magnitude. The model successfully prioritizes recall over precision, aligning with the asymmetric cost structure of industrial maintenance.

4 Discussion

4.1 Impact of the Dual-Model Approach

The comparison between 3-class and binary models reveals fundamental insights about handling extreme class imbalance:

1. **Sample size matters**: The 3-class model fails on BROKEN (7 samples) but succeeds on RECOVERING (893 samples). The binary model succeeds by combining them (900 samples).

2. **Problem reformulation is valid**: When the original problem formulation is infeasible due to data constraints, reformulating to a more practical task (anomaly detection) yields better results.

3. **Practical vs theoretical**: The 3-class approach is theoretically correct but practically useless for BROKEN detection. The binary approach sacrifices granularity for reliability.

4.2 Threshold Selection for Deployment

The probability threshold can be adjusted based on operational requirements:

Table 6: Threshold Impact on Performance

Threshold	Anomaly Recall	False Positive Rate
0.3	∼95%	High
0.5 (default)	88%	Medium
0.7	∼75%	Low

From an operational perspective, a lower threshold (more false alarms, fewer missed failures) may be preferable given the asymmetric costs: unnecessary inspection is far less costly than unexpected equipment failure.

4.3 Limitations

1. **Extremely limited failure data**: Only 7 BROKEN events constrain model learning

2. **High false positive rate**: 473 false alarms may cause alert fatigue in production

3. **Single failure mode**: Results may not generalize to different pump failure types

4. **Class ambiguity**: RECOVERING overlaps with early BROKEN stages, limiting distinguishability

5 Conclusion

This project demonstrates the effectiveness of GRU-based architectures for multivariate industrial time-series classification under extreme class imbalance. Our dual-model approach reveals important insights:

Table 7: Summary: 3-Class vs Binary Model

Metric	**3-Class**	**Binary**
Balanced Accuracy	62.46%	**93.11%**
Anomaly Recall	44.7%	**88.2%**
BROKEN Detection	0%	N/A
ROC-AUC	N/A	**96.11%**

Key findings:

1. The 3-class model fails completely on BROKEN class (0% recall) due to insufficient training samples
2. The binary model achieves 93.11% balanced accuracy and 96.11% ROC-AUC by reformulating the problem
3. High false positive rate is acceptable given cost asymmetry in maintenance contexts
4. Problem reformulation (3-class $\rightarrow$ binary) is a valid strategy when original formulation is data-constrained

Recommendation: Deploy the binary model as an early warning system, with threshold adjusted based on maintenance team capacity and risk tolerance. The model can provide alerts 10-20 minutes before failures based on observed sensor patterns.

Future work should focus on collecting more failure data and exploring unsupervised anomaly detection approaches that learn normal patterns rather than relying on labeled failure samples.