

**MACHINE LEARNING BASED WATER DISTRIBUTION
SYSTEM LEAKAGE TYPE DETERMINATION AND SEVERITY
ASSESSMENT**

BY

AHMAD FARES AL-NOUTI

A Thesis Presented to the
COLLEGE OF GRADUATE AND INTERDISCIPLINARY STUDIES
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

Civil and Environmental Engineering

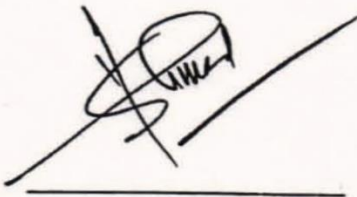
May 2025

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN- 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **Ahmad Fares Al-Nouti** under the direction of his thesis advisor and approved by his thesis committee, has been presented and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN CIVIL ENGINEERING**.



Dr. Shamsad Ahmad
Department Chairman



Zahr Mundher Yaseen
21/5/25

Dr. Zaher Mundher Yaseen
(Advisor)

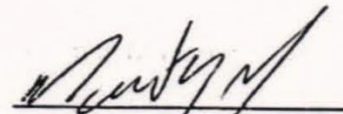


Dr. Suliman Al-Homidan
Dean of Graduate Studies



Dr. Muhammad Alsuwaiyan
(Member)

20/5/2025
Date



Dr. Moataz Ahmed
(Member)



© Ahmad Fares Al-Nouti

2025

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

{ يَا أَيُّهَا الْإِنْسَانُ مَا غَرَّكَ بِرَبِّكَ الْكَرِيمِ ○ الَّذِي خَلَقَكَ فَسَوَّاكَ فَعَدَلَكَ ○ فِي أَيِّ صُورَةٍ مَا شَاءَ رَكَّبَكَ ○ }

(Surah Al-Infitar, 82:6-8)

“O humanity! What has emboldened you against your Lord, the Most Generous, who created you, fashioned you, and perfected your design, moulding you in whatever form He willed?”

To the One who created me, shaped me, and guided me through every moment of this journey, to Allah, Al-Karim, the Most Generous. This work is sincerely dedicated to Your Noble Countenance, blessed and exalted are You.

ACKNOWLEDGMENTS

First and foremost, I thank my Lord, Allah, for His countless blessings and for making every step of this journey smooth and manageable. If He willed to leave me on my own, none of this would have been possible.

My deepest gratitude goes to my beloved parents. Your endless support, prayers, love, and belief in me have been my anchor. Without you, I would have been lost, overwhelmed, or deeply discouraged. Thank you for being my constant source of strength and light.

I would also like to express my sincere appreciation to my advisor, Dr. Zaher Mundher Yaseen, for his invaluable support and guidance, not only during my thesis but throughout the entire course of my master's program. His mentorship, patience, and encouragement have made a lasting impact on my academic and personal growth.

TABLE OF CONTENTS

ACKNOWLEDGMENTS.....	v
TABLE OF CONTENTS.....	vi
LIST OF TABLES.....	viii
LIST OF FIGURES.....	xii
LIST OF ABBREVIATIONS.....	xviii
ABSTRACT.....	xx
ملخص الرسالة.....	xxii
CHAPTER 1 INTRODUCTION.....	1
1.1 Background.....	1
1.2 Research Scope.....	2
1.3 Research Significance	3
1.4 Research Objectives	5
1.5 Need of Research.....	6
1.6 Thesis Layout	8
CHAPTER 2 LITERATURE REVIEW.....	11
2.1 Leakage concept.....	11
2.2 Sensor frameworks.....	12
2.3 Distribution network topologies.....	15
2.4 Machine Learning employment.....	17
2.5 Gap Recognition and Motivation.....	23
CHAPTER 3 MATERIALS AND METHODOLOGY.....	32
3.1 Data Description.....	32
3.2 Deep Learning Algorithms	39
3.2.1 Recurrent Neural Networks.....	39
3.2.2 Long Short-Term Memory.....	41
3.2.3 Gated Recurrent Unit	43
3.3 Ensemble Algorithms	45
3.3.1 Extreme Gradient Boosting.....	45
3.3.2 Light Gradient Boosting Machine	49
3.3.3 Random Forest.....	50
3.4 Models Development	52
3.4.1 Leakage Type Classification.....	54
3.4.2 Leakage Severity Assessment	56

3.4.3 <i>Performance Analysis</i>	58
CHAPTER 4 RESULTS AND DISCUSSION	64
4.1 Data Preprocessing	64
4.2 Branched Topology	76
4.2.1 <i>Without outliers</i>	77
4.2.2 <i>With outliers</i>	119
4.3 Looped Topology	159
4.3.1 <i>Without outliers</i>	159
4.3.2 <i>With outliers</i>	206
CHAPTER 5 CONCLUSION	250
References	255
Vitae	268

LIST OF TABLES

- Table 2.1:** ML-based literature overview of leakage detection, localization and severity assessment of WDS using different types of sensors.
- Table 3.1:** all possible values of naming convention characters followed to name CSV and RAW files in the dataset.
- Table 3.2:** LSTM time-step formulations tabulated for three gates (Forget Gate, Input Gate, and Output Gate) in addition to Cell State update.
- Table 3.3:** Values used in the training process of the internal parameters of XGB model in both severity assessment and leakage detection tasks.
- Table 3.4:** Formulations used to calculate overall performance metrics in classification ML models evaluation process based on confusion matrices.
- Table 3.5:** Abbreviations, and brief description of the terms used in the formulas of calculating By-class performance metrics.
- Table 3.6:** Formulations used to calculate overall performance metrics in classification ML models evaluation process based on confusion matrices.
- Table 4.1:** Measurements of Leak flowrate based on demand and supply flowrates for both looped and branched WDS.
- Table 4.2:** Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over branched network after removing the outliers.
- Table 4.3:** By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.
- Table 4.4:** By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.
- Table 4.5:** By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.
- Table 4.6:** By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.
- Table 4.7:** By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.
- Table 4.8:** By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.
- Table 4.9:** Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over branched network after removing the outliers.
- Table 4.10:** By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.
- Table 4.11:** By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

Table 4.12:By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

Table 4.13:By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

Table 4.14:By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

Table 4.15:By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

Table 4.16:Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over branched network before removing the outliers.

Table 4.17:By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

Table 4.18:By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

Table 4.19:By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

Table 4.20:By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

Table 4.21:By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

Table 4.22:By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

Table 4.23:Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over branched network before removing the outliers.

Table 4.24:By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

Table 4.25:By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

Table 4.26:By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

Table 4.27:By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

Table 4.28:By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

Table 4.29:By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

Table 4.30:Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over looped network after removing the outliers.

Table 4.31:By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

Table 4.32:By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

Table 4.33:By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

Table 4.34:By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

Table 4.35:By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

Table 4.36:By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

Table 4.37:Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over looped network after removing the outliers.

Table 4.38:By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

Table 4.39:By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

Table 4.40:By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

Table 4.41:By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

Table 4.42:By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

Table 4.43:By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

Table 4.44:Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over looped network before removing the outliers.

Table 4.45:By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

Table 4.46:By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

Table 4.47:By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

Table 4.48:By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

Table 4.49:By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

Table 4.50:By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

Table 4.51:Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over looped network before removing the outliers.

Table 4.52:By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

Table 4.53:By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

Table 4.54:By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

Table 4.55:By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

Table 4.56:By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

Table 4.57:By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

LIST OF FIGURES

- Figure 1.1:** Overview of Proposed ML Model Architecture which Illustrates the categorization and flow of 120 models for leak type classification and severity assessment in WDS.
- Figure 3.1:** Experimental layout of looped WDS topology in the laboratory where (a) real-time perspective view, (b) schematic view.
- Figure 3.2:** Experimental layout of branched WDS topology in the laboratory where (a) real-time perspective view, (b) schematic view.
- Figure 3.3:** systematic presenting all possible pathways of the available WDS dataset, where each possible pathway represents two CSV/RAW files that include corresponding sensors readings (Sensor1, Sensor2).
- Figure 3.4:** schematic diagram showing the overall learning mechanism of RNN model which consists of Input layer, Hidden SimpleRNN layers, followed by Output layer.
- Figure 3.5:** visualization of the calculations occurring inside of each LSTM layer in a multilayer LSTM ML model.
- Figure 3.6:** visualization of the calculations occurring inside of each GRU layer in a multilayer GRU ML model.
- Figure 3.7:** visual presentation of the classification learning mechanism of XGB classifier model starting from the input of training set and ending with the final predictions of training set.
- Figure 3.8:** Random Forest classifier prediction process visualization.
- Figure 3.9:** Structured workflow for processing and analyzing data from CSV and RAW files to develop predictive models for leak type classification and severity assessment in a water distribution system.
- Figure 3.10:** Systematic illustration of the leakage detection dataset structure for both cases (after and before removing the outliers) used in models development.
- Figure 3.11:** Visual representation of the labeling process, highlighting the systematic methodology and the relationships between sensor data and severity assessment.
- Figure 4.1:** Visualization of the impact of transient flow demand on ML model classification accuracy, demonstrating fluctuations across different leak types.
- Figure 4.2:** Severity of leakage observed to increase as demand decreases, visualization of leakage flow rates for each leak type in both systems (a) branched (b) looped.
- Figure 4.3:** Impact of outlier removal on data scarcity in the looped WDS. The bar charts illustrate the number of observations before and after outlier removal for different leakage types (a) Effect on the validation and test datasets (b) Effect on the training dataset.
- Figure 4.4:** Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over branched network after removing the outliers.
- Figure 4.5:** Confusion matrices for four Branched network RF models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.

- Figure 4.6:** Bar charts presenting classification performance of RF model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.7:** Radar charts illustrating RF model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.8:** Confusion matrices for four Branched network LightGBM models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.
- Figure 4.9:** Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.10:** Radar charts illustrating LightGBM model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.11:** Confusion matrices for four Branched network XGB models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.
- Figure 4.12:** Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.13:** Radar charts illustrating XGB model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.14:** Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over branched network after removing the outliers.
- Figure 4.15:** Confusion matrices for four Branched network LSTM models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.
- Figure 4.16:** Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.17:** Radar charts illustrating LSTM model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.18:** Confusion matrices for four Branched network RNN models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.
- Figure 4.19:** Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

- Figure 4.20:** Radar charts illustrating RNN model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.21:** Confusion matrices for four Branched network GRU models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.
- Figure 4.22:** Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.23:** Radar charts illustrating GRU model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.24:** Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over branched network before removing the outliers.
- Figure 4.25:** Bar charts presenting classification performance of RF model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.26:** Radar charts illustrating RF model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.27:** Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.28:** Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.29:** Radar charts illustrating XGB model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.30:** Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over branched network before removing the outliers.
- Figure 4.31:** Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.32:** Radar charts illustrating LSTM model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

- Figure 4.33:** Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.34:** Radar charts illustrating RNN model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.35:** Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.36:** Radar charts illustrating GRU model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.37:** Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over looped network after removing the outliers.
- Figure 4.38:** Confusion matrices for four Looped network RF models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.
- Figure 4.39:** Bar charts presenting classification performance of RF model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.40:** Radar charts illustrating RF model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.41:** Confusion matrices for four Looped network LightGBM models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.
- Figure 4.42:** Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.43:** Radar charts illustrating LightGBM model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.44:** Confusion matrices for four Looped network XGB models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.
- Figure 4.45:** Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.46:** Radar charts illustrating XGB model performance after removing outliers for Leakage type determination task of Looped network across different leakage

classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

Figure 4.47: Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over looped network after removing the outliers.

Figure 4.48: Confusion matrices for four Looped network LSTM models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

Figure 4.49: Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

Figure 4.50: Radar charts illustrating LSTM model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

Figure 4.51: Confusion matrices for four Looped network RNN models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

Figure 4.52: Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

Figure 4.53: Radar charts illustrating RNN model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

Figure 4.54: Confusion matrices for four Looped network GRU models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

Figure 4.55: Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

Figure 4.56: Radar charts illustrating GRU model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

Figure 4.57: Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over looped network before removing the outliers.

Figure 4.58: Bar charts presenting classification performance of RF model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

Figure 4.59: Radar charts illustrating RF model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

- Figure 4.60:** Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.61:** Radar charts illustrating LightGBM model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.62:** Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.63:** Radar charts illustrating XGB model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.64:** Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over looped network before removing the outliers.
- Figure 4.65:** Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.66:** Radar charts illustrating LSTM model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.67:** Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.68:** Radar charts illustrating RNN model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 4.69:** Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.
- Figure 4.70:** Radar charts illustrating GRU model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.
- Figure 5.1:** visual presentation of the expected impact and outcomes of ML models of water leakage type classification and severity assessment.

LIST OF ABBREVIATIONS

WDS	:	Water Distribution System
ML	:	Machine Learning
AI	:	Artificial Intelligence
ANN	:	Artificial Neural Network
SVM	:	Support Vector Machine
k-NN	:	k-Nearest Neighbor
ET	:	Extra Trees
NRW	:	Non-Revenue Water
RF	:	Random Forest
LightGBM	:	Light Gradient Boosting Machine
XGB	:	Extreme Gradient Boosting
CNN	:	Convolutional neural networks
LSTM	:	Long Short-Term Memory
RNN	:	Recurrent Neural Network
GRU	:	Gated Recurrent Unit
BR	:	Branched
LO	:	Looped
LC	:	Longitudinal Crack

CC	:	Circumferential Crack
GL	:	Gasket Leakage
OL	:	Orifice Leakage
NL	:	Non-Leak
MEMS	:	Micro-electromechanical Systems
IoT	:	Internet of Things
IQR	:	interquartile range
A	:	Accelerometer
P	:	Pressure Transducer
H	:	Hydrophone
S	:	Sound Present
NS	:	Sound Absent
ND	:	No Demand
LPS	:	Liter Per Second
TP	:	True Positives
FP	:	False Positives
TN	:	True Negatives
FN	:	False Negatives
NIR	:	No Information Rate

ABSTRACT

Full Name : Ahmad Fares Al-Nouti

Thesis Title : Machine learning based water distribution system leakage type determination and severity assessment

Major Field : CIVIL ENGINEERING – WATER AND ENVIRONMENTAL

Date of Degree : May 2025

Pipeline leakages in water distribution systems (WDS) cause substantial water losses, elevated operational costs, and environmental damage, exacerbating water scarcity challenges. This thesis addresses these issues by applying advanced machine learning (ML) techniques to automatically classify leak types and assess leak severity in WDS pipelines. Recent studies have demonstrated the potential of ML-based leak detection, with models like Random Forest and Convolutional Neural Networks achieving high accuracy (often above 90%) in identifying leaks. However, prior work typically treats leakage as a binary event or focuses on localization, leaving a gap in distinguishing specific leak failure modes and quantifying leak severity, which this research aims to fill.

This study develops a multi-sensor data-driven framework using accelerometers (vibration), hydrophones (acoustic), and pressure transducers to capture leak signatures in both branched and looped pipeline network configurations. Experiments were conducted on a controlled pipeline testbed under diverse conditions, including varying flow rates (steady and transient flows) and simulated background noise, to emulate real-world scenarios. The acquired dataset encompasses scenarios with no leak (normal operation) and four distinct leak types – longitudinal crack, circumferential crack, gasket leak, and orifice leak – along with leak severity labels of minor, moderate, and severe determined by leak flow rate. Supervised ML classifiers including Extreme Gradient Boosting (XGB), Random Forest, and LightGBM were trained alongside deep

learning models (recurrent neural networks, LSTM and GRU) to evaluate their performance in leak type determination and severity classification. Rigorous preprocessing (interquartile range outlier removal and feature normalization) was applied to improve model generalization. The results show that leak type classification can be achieved with very high accuracy: the best model (XGB) attained nearly 99.9% balanced accuracy and an F1-score of ≈ 0.99 in distinguishing the five leak classes (including the non-leak case). For leak severity prediction, the top-performing model (LightGBM) reached about 93% overall accuracy ($\kappa \approx 0.85$), correctly identifying severe leaks with $\approx 97\%$ recall, although its precision for minor leaks was lower.

These findings demonstrate that a multi-sensor ML approach can reliably identify not only the presence of a leak but also the specific leak type and its severity. The trained models maintained robust performance across both branched and looped network layouts. Furthermore, validation and test results were closely aligned, indicating minimal overfitting and strong generalization under transient flow conditions and sensor noise. Notably, minor leaks remained the most challenging to classify: these small leaks often produced sensor signals that overlap with normal background variations, leading to occasional misclassification and indicating the need for further feature refinement to improve sensitivity. Overall, the proposed ML-based leakage diagnosis system enables water utilities to prioritize repairs for critical leaks, optimize maintenance scheduling, and reduce non-revenue water losses. This research contributes a scalable, cost-effective pipeline monitoring solution that enhances the sustainability and resilience of water distribution networks, laying the groundwork for future advancements in intelligent leak detection and predictive maintenance.

ملخص الرسالة

الاسم الكامل: أحمد فارس محمود النوتي

عنوان الرسالة: تحديد نوع التسريب وتقييم شدة التسريب في أنظمة توزيع المياه بواسطة تقنيات التعلم الآلي

التخصص: الهندسة المدنية | المياه والبيئة

تاريخ الدرجة العلمية: أيار - 2025

تتسبب تسربات الأنابيب في أنظمة توزيع المياه (WDS) - في خسائر مائية كبيرة، وارتفاع في التكاليف التشغيلية، وأضرار بيئية، مما يؤدي إلى تفاقم التحديات المتعلقة بندرة المياه. تتناول هذه الأطروحة هذه المشكلات من خلال تطبيق تقنيات التعلم الآلي المتقدم (الذكاء الاصطناعي المعتمد على البيانات) بهدف تصنيف أنواع التسربات تلقائيًا وتقييم مدى حدتها في أنابيب شبكات توزيع المياه. أظهرت دراسات حديثة الإمكانيات العالية للتقنيات الذكية القائمة على التعلم الآلي في اكتشاف التسربات، حيث حققت نماذج مثل "غابة القرارات العشوائية" و"الشبكات العصبية الالتفافية" دقة مرتفعة غالبًا ما تتجاوز 90٪ في التعرف على التسربات. ومع ذلك، فإن معظم الأبحاث السابقة تعاملت مع التسرب باعتباره حدثًا ثنائيًا (يوجد أو لا يوجد) أو ركزت فقط على تحديد موقعه، مما يترك فجوة في القدرة على التمييز بين أوضاع فشل التسرب المختلفة وتقدير شدته. وتهدف هذه الدراسة إلى سد هذه الفجوة.

طورت الدراسة إطارًا منهجيًا يعتمد على بيانات متعددة المستشعرات باستخدام مقاييس التسارع (لقياس الاهتزازات)، والهيدروفونات (لالتقاط الصوتيات)، ومحولات الضغط لالتقاط بصمات التسرب في شبكات أنابيب ذات تكوينات متفرعة وحلقية. وقد أجريت التجارب على منصة اختبار خاضعة للرقابة تحاكي سيناريوهات حقيقية، وشملت ظروفًا متنوعة من معدلات تدفق مختلفة (تدفق مستقر وتدفق متغير) وضوضاء خلفية تم تمثيلها بشكل مصطنع. وتضم قاعدة البيانات الناتجة حالات تشغيل طبيعية (دون تسرب) وأربعة أنواع مختلفة من التسربات: شق طولي، شق دائري، تسرب من الجوان (الكاوتشوك)، وتسرب عبر فتحة، بالإضافة إلى مستويات شدة التسرب المصنفة إلى: طفيف، متوسط، وشديد، اعتمادًا على معدل تدفق التسرب.

تم تدريب نماذج تعلم آلي تحت إشراف بشري شملت: نموذج التعزيز المتدرج الشديد (Extreme Gradient Boosting - XGB)، ونموذج غابة القرارات العشوائية، ونموذج التعلم الخفيف المعزز (LightGBM)، إلى جانب نماذج التعلم العميق مثل الشبكات العصبية المتكررة (RNN)، و شبكات الذاكرة طويلة

وقصيرة الأمد (LSTM) ، والوحدات التكرارية البوابية (GRU) ، وذلك لتقييم أدائها في تصنيف نوع التسرب وشدته. وقد تم تطبيق معالجة مسبقة دقيقة للبيانات شملت إزالة القيم المتطرفة باستخدام نطاق الربيعيات وتطبيق الخصائص بهدف تحسين قدرة النماذج على التعميم.

أظهرت النتائج أن تصنيف أنواع التسربات يمكن تحقيقه بدقة عالية جداً؛ حيث حقق النموذج الأفضل (XGB) دقة موزونة بلغت حوالي 99.9% ودرجة F1 بلغت ≈ 0.99 عند التمييز بين الأنواع الخمسة (بما في ذلك حالة عدم التسرب). أما بالنسبة لتوقع شدة التسرب، فقد تمكن النموذج الأعلى أداءً (LightGBM) من تحقيق دقة إجمالية تقدر بحوالي 93% (معامل كبا ≈ 0.85)، وتمكن من التعرف على التسربات الشديدة بنسبة استدعاء $\approx 97\%$ ، رغم انخفاض دقته في تحديد التسربات الطفيفة.

تُظهر هذه النتائج أن النهج متعدد المستشعرات المعتمد على التعلم الآلي قادر على تحديد ليس فقط وجود التسرب، بل أيضاً نوعه وشدته بدقة. كما أظهرت النماذج المدربة أداءً قوياً في كل من تكوينات الشبكات المتفرعة والحلقية. وفضلاً عن ذلك، فإن نتائج التحقق والاختبار كانت مقاربة، مما يشير إلى أن مشكلة الإفراط في التكيف (Overfitting) كانت طفيفة، كما أن النماذج كانت قادرة على التعميم بشكل جيد في ظل ظروف تدفق غير مستقرة وضجيج أجهزة الاستشعار. ومع ذلك، بقيت التسربات الطفيفة هي الأصعب في التصنيف، حيث أنتجت إشارات استشعار تتداخل مع الخلفية الطبيعية، مما أدى إلى بعض حالات التصنيف الخاطئ، ما يبرز الحاجة إلى تحسين خصائص البيانات لزيادة الحساسية.

بصورة عامة، يقدم نظام تشخيص التسربات المقترح والقائم على تقنيات الذكاء الاصطناعي أداة فعالة لمؤسسات المياه لتمكينها من إعطاء الأولوية لإصلاح التسربات الحرجة، وتحسين جدول الصيانة، وتقليل فاقد المياه غير المفوترة. وتسهم هذه الدراسة في تقديم حل قابل للتوسع وذو كفاءة من حيث التكلفة لمراقبة الأنابيب، مما يعزز من استدامة ومرونة شبكات توزيع المياه، ويمهد الطريق لتطويرات مستقبلية في الكشف الذكي عن الأعطال والصيانة التنبؤية.

CHAPTER 1

INTRODUCTION

1.1 Background

Pipeline leakages are a major concern across various industries, including gas, oil, and water distribution systems, significantly affecting the environment, economy, and public health. Leaks in oil and gas pipelines can lead to environmental contamination, including soil and water pollution, which endangers ecosystems and human health. Additionally, such incidents can cause significant economic losses due to product wastage and cleanup costs [1], [2]. For instance, methane leaks from natural gas pipelines not only contribute to global warming but also pose risks of explosions, making early detection crucial for safety and economic efficiency.

Wastewater pipelines can also suffer from leaks, leading to untreated waste contaminating surrounding soils and water bodies. This poses severe public health risks, as it can lead to waterborne diseases and degrade water quality. In water distribution systems, leakages are a critical issue due to their direct impact on water availability. Water loss from leaks exacerbates issues like water scarcity, increasing stress on already limited water resources. This is particularly problematic in areas facing climate change-induced shifts in water availability [3].

Leaks in water distribution systems are categorized based on their nature, such as visible surface leaks and non-visible underground leaks [4]. Underground leaks are particularly challenging to detect due to the buried nature of pipelines and the complexity of their networks [5]. Focusing on leak detection in water systems is essential because it directly affects the efficiency of water supply systems and the operational costs of water utilities [6]. The application of machine learning (ML) for predicting and detecting leaks has emerged as a promising approach due to its ability to analyze large datasets, identify patterns, and provide predictive maintenance strategies that help in early leak detection and localization [7].

1.2 Research Scope

Recent literature has extensively studied the detection of pipeline leakage using both hardware-based and software-based approaches. Hardware-based methods include the use of sensors such as acoustic devices, infrared sensors, and ground-penetrating radar, which physically interact with the pipeline to detect anomalies [1], [3]. Software-based approaches, on the other hand, leverage data from sensors and employ numerical modeling techniques like inverse transient analysis or time domain analysis. These methods aim to analyze pressure or flow data to identify potential leaks [8], [9], [10].

Several studies have explored the integration of artificial intelligence (AI) and ML models, such as artificial neural network (ANN), support vector machine (SVM), and deep learning models, for leak detection [1], [11], [12]. These models can automate the analysis process, improving the accuracy of detection and enabling real-time monitoring. However, some limitations have been identified, such as the need for large datasets for model training

and the challenge of achieving high accuracy across different pipeline environments [2]. While laboratory simulations can mimic leak scenarios, their reliability is often constrained by their ability to replicate real-world conditions accurately. The effectiveness of such simulations depends on how closely they can emulate the complexities of actual pipeline networks, including variables like pressure changes and flow dynamics [3].

For the benchmarked dataset used in this research, the available studies up to date are split into two. One is about leak detection (Leak, nonLeak) using hydrophone sensors data, where the ML models namely, Random Forest (RF) and SVM, achieved outstanding accuracy of 0.95 and 0.9 respectively [13]. The data was also used to create combined dataset of Dynamic Pressure transducers and Accelerometers measurements. To locate and detect leakage using ANN with accuracy of 0.87 [14]. Then, a classification approach was adopted to determine the category of leak for the same benchmarked data. Many ML models were used such as Extra Trees (ET) and RF, for the Leak type classification of both Branched and Looped networks respectively. It was found that Accelerometer dataset gave the best accuracy of 0.984 for ET and 0.934 for RF in terms of classifying with Feature selection [15].

1.3 Research Significance

The significance of this research lies in its potential to advance leak detection methods, especially in regions where water losses due to leaks are a major issue. In developing countries, NRW, which is water that is produced but not billed due to leaks or theft, can account for up to 30% of the total water supplied [3]. Addressing leaks in these areas is crucial for improving water security and reducing operational costs for utilities. By

developing more accurate and efficient leak detection methods, this research can help reduce physical losses in pipeline networks, ensuring that water resources are better managed and preserved.

In developed countries, the emphasis is often on reducing the costs associated with maintaining large water distribution systems. Improved leak detection can significantly cut down the resources spent on manual inspections and repairs, leading to lower maintenance costs and more sustainable water management practices [1]. The findings of this study can, therefore, contribute to both improving the operational efficiency of water distribution networks in developing regions and optimizing the maintenance strategies of advanced systems in developed countries.

The selection of ML models, namely: XGB, RF, LightGBM, LSTM, RNN, and GRU, is grounded in their proven effectiveness in real-world water distribution systems for leakage detection and severity assessment. For instance, XGB has been utilized to develop leak detection models with high accuracy, effectively reducing water waste and economic losses [16]. RF models have demonstrated robustness in handling complex datasets, making them suitable for identifying leak patterns amidst environmental noise [17]. LSTM networks, adept at processing sequential data, have been employed to detect leaks by analyzing time-series sensor data, capturing temporal patterns indicative of leaks [18]. The practical application of these models in various studies underscores their suitability for enhancing leak detection and severity assessment in water distribution networks.

The significance of this research extends to include leakage severity assessment which lies in its potential to provide utilities with a structured framework for prioritizing

repair efforts [19]. In many regions, particularly those with aging infrastructure or limited maintenance resources, accurately assessing the severity of leaks is critical for optimizing resource allocation [20], [21]. By employing ML algorithms, this research aims to categorize leaks into severity levels (e.g., severe, moderate, or minor), enabling utilities to focus their efforts on high-priority leaks that pose the greatest risk to water loss and system performance [21], [22]. This stratified approach not only improves the overall efficiency of maintenance workflows but also ensures that critical leaks are addressed promptly, reducing water losses and preserving system reliability [19], [23].

This focus on severity assessment not only minimizes water and energy wastage but also contributes to more sustainable and economically viable water management practices in both developing and developed regions [24], [25].

1.4 Research Objectives

The research was motivated by the necessity to establish a reliable computer aid technology for water distribution system monitoring and assessment. Data were collected from an open sources literature about water distribution system leakage records for different types of networks “branched and looped”. The leakage was recorded using different types of sensors (i.e., *accelerometers, hydrophones, and dynamic pressure transducers*). Hence, the main research objectives were designed as follows:

- A newly developed integrated ML models was developed for water distribution system network leakage,
- For better understanding the behavior of the leakage in different network, two types of networks were investigated including branched and looped,

- As the type of leakage plays essential role in each water distribution system network, different types of leakage (i.e., *Longitudinal Crack*, *Circumferential Crack*, *Gasket Leak*, *Orifice Leak*, *Nonleak*) were tested.
- Knowing leakage severity in terms of losses is essential to help maintain the WDS economically. Therefore, the severity of leakage is assessed using different sensors readings to classify the leakage into severe, moderate, or minor.
- Due to the existence of the uncertainty in developing ML models, different types of associated uncertainties were investigated including input parameters, models, and data.

The ultimate objective of this research is to employ a ML model to accurately classify leakage data in water distribution systems, predicting the type of leakage and its severity when applicable. This will involve utilizing historical and real-time data to train the ML models for effective detection and prediction, improving the response time and accuracy of leakage identification.

1.5 Need of Research

Leakage in water distribution systems is a critical issue that affects both developed and developing countries, with significant impacts on water availability, economic losses, and infrastructure maintenance. Several factors contribute to leakage in these systems, including internal pressures, pipeline material, age, and external factors like soil movements and traffic loads. Understanding these factors is crucial for designing effective strategies to reduce water loss and maintain the integrity of the distribution network.

Factors influencing leakage include Pressure and Flow Dynamics, Material quality, Pipe Aging, External Loads, and Environmental Conditions.

High pressures in pipelines can increase the rate of leakage, especially through small cracks or weak joints, as pressure fluctuations directly impact the volume of water lost. Adjusting pressure levels can be a means of controlling leaks, but it requires careful balance to maintain service quality without exacerbating leaks [3], [26].

Over time, pipeline materials like PVC and iron experience wear and tear, making them more susceptible to cracks and bursts. This aging process is influenced by external conditions such as temperature changes, which can cause expansion or contraction in the pipes, leading to potential failure points [3].

Factors like soil movement, vibrations from nearby traffic, and even seasonal temperature variations can contribute to the deterioration of pipeline integrity. These conditions can introduce additional stresses, particularly to pipes near roads or those subjected to significant external loads [26].

Several solutions have been proposed to mitigate water leakage, including both hardware-based and software-based approaches. Hardware methods like acoustic sensors, ground-penetrating radar, and pressure sensors are commonly used to detect leaks. However, these methods often come with high costs and require extensive maintenance [1], [3].

Software-based methods, especially those utilizing ML, have emerged as powerful tools for identifying leaks. ML models can process large datasets from pressure and flow sensors to detect patterns indicative of leaks. This approach is more scalable and can

provide real-time insights into the network's health [3], [26]. Furthermore, the integration of these methods with hydraulic models can enhance the accuracy of leak localization by comparing simulated and observed data to pinpoint leak locations [26].

Determining the type, location, and size of a leak is critical for efficient water management. Knowing the type of leakage helps in selecting the appropriate repair strategy, while accurate location identification reduces the time and costs associated with pinpointing and fixing leaks. Furthermore, understanding the size of a leak allows for prioritizing repairs based on the severity of water loss [26].

These insights highlight the extensive need for research in developing advanced ML models that can accurately classify and localize leakages in water distribution systems. Such models would enable more proactive management, reduce water loss, and support the sustainability of water resources, especially in regions where infrastructure is aging or where maintenance budgets are limited. By focusing on these aspects, the research aims to fill existing gaps in predictive capabilities, providing a framework that enhances water utility management and reduces non-revenue water losses.

1.6 Thesis Layout

This thesis is structured into five comprehensive chapters. Chapter 1 introduces the research, setting the stage with an overview of the growing challenges in managing water distribution system (WDS) leakages. It presents a holistic review of current literature, highlighting gaps and emphasizing the importance of advanced solutions. Sections 1.3, 1.4, and 1.5 delve into the significance of this research within the context of modern leakage management practices, advocating the need for machine learning techniques to classify

both the type and severity of leaks. This chapter also outlines the clearly defined objectives designed to fill those literature gaps.

Chapter 2 expands the discussion by reviewing in detail the foundational concepts surrounding WDS leakages. It distinguishes between leakage detection and the more nuanced task of leakage type identification. Further, it explores various system configurations—branched vs. looped networks—and discusses existing methods for leak localization, size estimation, and type classification. Special attention is given to machine learning applications, particularly those utilizing single or multi-sensor frameworks for improved accuracy.

Chapter 3 is divided into four key sections. Section 3.1 introduces the dataset used in the research, explaining the experimental setup, data acquisition process, and the relevant features extracted. Section 3.2 describes the training methodology for tree-based machine learning models, while Section 3.3 outlines the architecture and training strategies for the proposed deep learning models. Section 3.4 consolidates the adopted methodology: beginning with model development for leak type classification, advancing through severity assessment (categorizing leaks as minor, moderate, or severe), and concluding with performance evaluation and robustness analysis of the models.

Chapter 4 constitutes the core of the thesis, presenting and interpreting the study's results. It begins with data preprocessing steps—including outlier removal and labeling of sensor signals by severity—and explains their necessity. The chapter then details the performance of machine learning models across different scenarios: branched systems with and without outliers, followed by looped systems under similar conditions. Four models were developed for severity assessment, and one model was dedicated to leak type

classification. Figure 1.1 visually summarizes this architecture, tracing the flow from initial model design to final application.

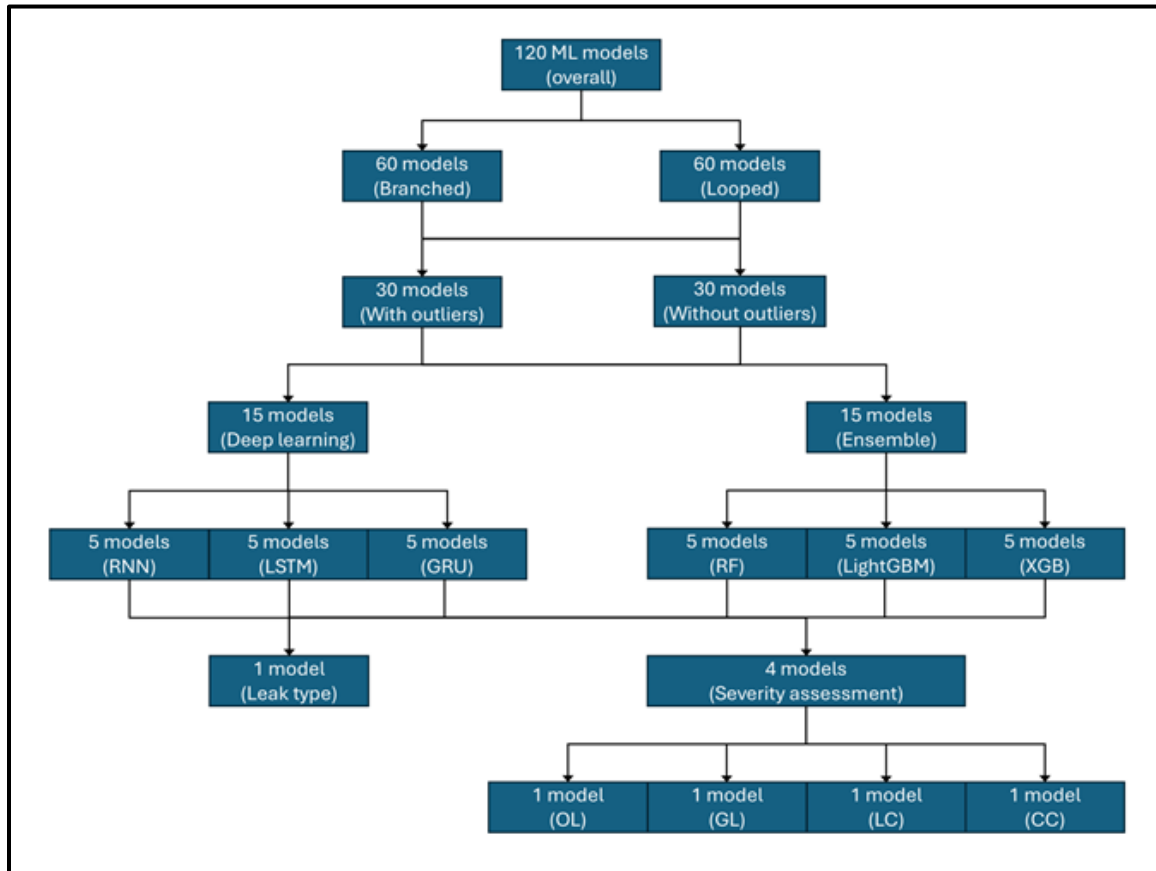


Figure 1.1: Overview of Proposed ML Model Architecture which Illustrates the categorization and flow of 120 models for leak type classification and severity assessment in WDS.

Finally, Chapter 5 wraps up the thesis by spotlighting the top-performing model, supported by its key performance metrics. It also discusses the potential engineering applications of the findings, offering a visual summary for ease of reference. The chapter concludes with future research directions, providing inspiration for those interested in advancing this vital area of study.

CHAPTER 2

LITERATURE REVIEW

2.1 Leakage concept

Leakage in water distribution networks (WDNs) is a pervasive and costly problem, leading to significant water losses and infrastructure damage. On average, urban water transmission pipelines lose roughly 20–30% of their water throughput due to leaks and related issues, and this figure can exceed 50% in aging systems with poor maintenance. Such leaks not only waste a precious resource but also cause financial losses and risks to public health and safety. For instance, pipeline leaks can undermine roadways, introduce contaminants, and interrupt supply, incurring billions in repair and social costs annually. Timely detection and control of leaks is therefore crucial to minimize environmental impact, ensure system reliability, and reduce economic losses [1], [16], [27].

Traditional leak detection methods often rely on manual surveys using listening devices or visual inspections, which are labor-intensive and prone to human error [28], [29], [30]. In contrast, modern approaches increasingly leverage sensor technologies and data analytics to automatically identify leak signatures in water networks [31], [32], [33]. Physical sensors deployed on pipelines can capture pressure transients, acoustic vibrations, or other signals caused by leaks, while machine learning (ML) algorithms analyze this data to recognize patterns indicative of leaks [31], [34], [35]. The overall goal is not only to detect the presence of a leak but also to characterize it – determining the leak type (e.g.

crack or joint failure) and assessing its severity or size so that utility operators can prioritize repairs and mitigate losses [36], [37], [38]. Achieving accurate detection, classification, and severity estimation of leaks in real time, however, remains a challenging task due to complex network dynamics and noisy sensor data [39]. Researchers are therefore actively exploring advanced sensing and ML techniques to improve leak diagnosis in water distribution systems [27], [40].

2.2 Sensor frameworks

The detection and assessment of WDS leakages have garnered significant attention due to their environmental and economic implications. Recent literature highlights the integration of multi-sensor frameworks, comprising pressure, acoustic, and vibration sensors, coupled with ML techniques to enhance the accuracy and timeliness of leakage type classification and severity assessment. Islam et al. emphasize that the growing adoption of vibration and acoustic sensors is driven by their cost-effectiveness and ease of installation, with ML algorithms increasingly being employed to automate detection processes and mitigate reliance on manual expertise [41]. Similarly, Rathi underscores the role of optimal sensor placement, especially pressure sensors, in accurately localizing leaks while addressing the challenges posed by variable operational conditions and infrastructural complexities in WDS [42]. Additionally, Wan et al. provide a comprehensive review of data-driven and model-based methods for leakage detection, advocating for the integration of smart sensors and real-time analytics to handle data uncertainties and enhance detection accuracy [43]. The confluence of these studies suggests a promising direction for advancing WDS management through the synergistic use of multi-sensor data and ML-based analytical frameworks.

Many studies suggest that water distribution networks face significant water losses due to pipeline leaks, making early leak detection a critical priority [31], [44], [45]. Recent research shows that continuous monitoring with pressure, acoustic, and vibration sensors is indispensable for timely leak identification [46], [47], [48]. These sensor types correspond to the primary indicators of a leak: pressure sensors capture drops or anomalies in water pressure, acoustic sensors detect sound signals (leak noise) in the water or pipe, and vibration sensors identify pipe wall vibrations caused by escaping water. Some researchers claim that these sensors provide a comprehensive understanding of the methodologies required for effective leak detection, making them essential components in water distribution systems [42], [48], [49].

Many researchers emphasize that most data-driven approaches for leak detection utilize continuous pressure readings, flow data, or acoustic signals from the network. Instead of relying on specialized survey tools, studies highlight the effectiveness of widely deployed sensors like pressure gauges and acoustic accelerometers, which enable long-term monitoring across extensive areas [50]. Evidence suggests that leaks cause sudden pressure drops and unusually strong acoustic signal amplitudes compared to normal conditions, making these sensors vital for identifying leak patterns [51]. Research also indicates that while methods based on acoustic or vibration signals depend on pipe material and environmental noise, pressure and flow-based methods rely on network topology and demand patterns [52]. These findings underscore that without data from pressure and acoustic sensors, leak detection models cannot be effectively trained or deployed.

Reliable leak detection in WDNs depends on effective sensing of the physical phenomena caused by leaks [53], [54]. A variety of sensors have been employed to capture leak-related signals, with a trend toward microelectromechanical systems (MEMS)-based devices for their low cost, small size, and wireless capabilities [39], [55], [56]. Acoustic/vibrational sensors are among the most widely used in this domain: for example, accelerometers, hydrophones, and piezoelectric microphones are attached to pipes to detect the sound or vibration produced by leaking water [57]. These acoustic methods (including portable noise loggers and “listening” devices) have proven effective at leak detection due to the distinctive noise profile leaks generate, especially in pressurized pipes [58], [59], [60]. Recent studies have demonstrated that MEMS accelerometer sensors can successfully pick up leak-induced pipe vibrations in both metal and plastic pipes, enabling leak detection with high accuracy. For instance, Tariq et al. trained ML classifiers (KNN, decision tree, random forest, AdaBoost) on MEMS accelerometer data from water pipes and found that a random forest achieved the best leak detection accuracy [40]. Similarly, Yu et al. reported that classifying pipe vibration signals with machine learning can reliably distinguish leaking from normal conditions in a distribution system [61].

In addition to vibration sensors, pressure sensors are commonly deployed at hydrants or pipe nodes to monitor pressure drops or transients associated with leaks [62], [63], [64]. Sudden pressure fluctuations (e.g. negative pressure waves) can indicate the occurrence of a leak, and networks of pressure loggers or wireless pressure transducers have been used to pinpoint leak events. Some approaches also incorporate flow sensors (flow meters) to detect unexpected increases in flow that would suggest leakage, though pressure and acoustic methods tend to be more sensitive [33], [65], [66]. Emerging sensor

technologies include fiber-optic cables and distributed acoustic sensing along pipelines, as well as high-sensitivity MEMS hydrophones placed in the water to “listen” for leak noise [67]. Overall, the trend is toward integrating multiple MEMS-based sensors that can be deployed permanently in the water network for continuous monitoring [67]. By leveraging the complementary strengths of different sensors (e.g. an accelerometer for structural vibration and a hydrophone for acoustic noise), the system can capture a richer picture of leak signatures, potentially improving detection reliability. However, most past studies use only a single type of sensor at a time, and research is still evolving on the optimal multi-sensor configurations for leak detection [9], [68], [69].

2.3 Distribution network topologies

Water distribution networks are typically designed in either looped (meshed) layouts or branched (tree) layouts, and this topology significantly influences leak behavior and detection strategies [14], [70], [71]. Looped networks feature interconnected pipeline loops that provide multiple paths for water flow between the supply source and consumers. This redundancy improves reliability and hydraulics (maintaining pressure even if one pipe is down), but it also means that the impact of a leak can propagate in complex ways through the mesh. In a looped system, a leak’s effects on pressure and flow may be distributed across several paths; thus, the “wave” or pressure disturbance from a leak does not travel along a unique, straightforward path as it would in a simple pipe, complicating the analysis of sensor data [72], [73], [74]. Branched networks, on the other hand, consist of tree-like hierarchies where each downstream node has a single supply path from the source. In such systems, a leak typically causes a more localized pressure drop and a clear signature upstream (since there is no alternative route for the water) [1], [7], [38]. Because of these

differences, leak detection and localization techniques often need to be adapted to network topology [14], [69], [75]. For example, threshold-based methods using flow balance or pressure gradients can more easily bracket a leak in a zoned branched network, whereas looped networks may require more advanced pattern recognition and sensor placement optimization to interpret the multi-path effects of leaks.

Experimental research has highlighted the importance of testing leak detection models on both topology types. A recent study by Aghashahi et al. built a laboratory testbed that could be reconfigured into looped and branched layouts to compare leak scenarios in each configuration. The results showed that certain leak signals (e.g. the amplitude of pressure transients or the frequency content of acoustic emissions) can differ between looped vs. branched networks even for the same leak size and type [76]. This is because loops allow some flow diversion, dampening the immediate pressure drop at the leak point, whereas branched systems concentrate the effect [14], [70], [71]. Consequently, machine learning models for leak detection must be robust to network topology – an algorithm trained on data from a branched network might not directly transfer to a looped network, and vice versa [77], [78], [79]. To address this, researchers are beginning to develop topology-independent features or to include network topology as a parameter in the model training phase [80], [81]. The inclusion of both looped and branched network data in leak detection studies is increasingly seen as necessary for developing generalized ML solutions that can be deployed in diverse water systems [7], [14].

2.4 Machine Learning employment

A wide range of machine learning techniques have been explored for detecting leaks and classifying leak characteristics in WDNs [7], [82], [83]. Supervised learning models are commonly trained on sensor data (pressure, acoustic, etc.) labeled as “leak” or “no leak,” and in some cases with specific leak types or sizes, to learn patterns distinguishing normal and leaking conditions [84], [85], [86]. Simpler classifiers like k-Nearest Neighbors (k-NN) and support vector machines (SVM) have achieved good performance in leak detection tasks, especially when combined with features extracted from the time and frequency domain of sensor signals [87], [88]. For example, features such as signal energy, entropy, or spectral peaks can be fed into an SVM to classify whether a given sensor reading indicates a leak [60], [88], [89]. However, recent studies indicate that more complex models often outperform these baseline learners [14], [80], [81]. Tree-based ensemble models (e.g. RF, Gradient Boosting Machines like XGB) have shown particularly high accuracy in leak detection due to their ability to capture nonlinear relationships and interactions in the data [82], [90]. Lee et al. compared several ML algorithms (DT, RF, LightGBM, XGB, CatBoost, etc.) on water pipeline vibration data and found that an ensemble boosting model (XGB) achieved nearly 99.8% classification accuracy in distinguishing leak vs. no-leak cases. Similarly, Ullah et al. evaluated neural networks, decision trees, random forests, and k-NN on acoustic emission data and reported an overall accuracy around 99% for leak detection, with tree-based models performing strongly. These results suggest that ensemble methods, by aggregating many decision rules, are very effective for capturing the subtle signatures of leaks amid noise [16], [27].

In parallel, deep learning approaches have gained traction for leak detection in recent years, particularly for processing raw sensor signals or high-dimensional data [89], [91], [92]. Convolutional neural networks (CNNs) have been used to analyze pressure transient patterns and acoustic spectrograms, automatically learning discriminative features for leaks [11], [59], [93], [94], [95], [96]. Guo et al. developed a time–frequency CNN that operates on leak signal spectrograms, achieving robust leak detection performance on a dataset of pipeline acoustic signals [97]. Recurrent neural networks (RNNs) and variants like LSTM and GRU have also been applied to capture temporal sequences of pressure data in WDNs, allowing the model to detect leak-induced anomalies over time [86], [98]. In general, deep learning models can outperform traditional methods when large volumes of sensor data are available for training, as they can learn complex leak signatures without manual feature engineering. However, they require careful tuning and are often less interpretable [60], [99], [100]. To balance this, some researchers integrate deep learning with ensemble methods or use hybrid schemes [30], [91], [99]. For instance, one study extracted features via wavelet transform of pressure signals and then applied a deep CNN classifier, while another used an LSTM to preprocess the data fed into a gradient boosting detector [92], [99]. Ensemble learning vs. deep learning is an active comparison in the literature: tree ensembles are fast to train and interpret, whereas deep nets may capture more nuanced patterns [101]. Both approaches have achieved high detection rates (>95–99%), so current research evaluates them side by side to identify the optimal model for a given leak detection application.

In addition to detection (leak vs no-leak), some ML models aim to classify the type of leak or estimate leak severity [15], [101], [102]. For example, researchers have treated

different leak mechanisms as separate classes – distinguishing longitudinal cracks from circular orifice leaks or joint gasket failures based on the sensor signature [58], [101]. Data-driven models including CNNs and ensemble classifiers have successfully identified leak types under controlled experiments with multiple leak scenarios [83], [87]. Likewise, leak severity (often correlated with leak hole size or flow rate) has been approached as a classification or regression task [85], [103]. Ullah et al. considered various pinhole leak sizes and demonstrated that ML classifiers could detect even minor leaks (small holes) with high confidence, effectively distinguishing different severity levels by analyzing acoustic emission features [27]. Another study applied an artificial neural network to estimate leak size from pressure sensor data in a simple pipeline, illustrating the potential for ML to quantify leak severity in addition to detecting its occurrence. These efforts underscore the versatility of ML techniques in not only finding leaks but also characterizing their nature, which is valuable for prioritizing repairs (e.g. a large pipe break vs a pinhole leak) and deploying the right maintenance response [104].

Table 2.1 presents a comprehensive overview of recent literature studies focusing on WDS leakage detection, localization, and severity assessment using ML-based methodologies. The studies summarized in the table highlight a diverse range of ML models such as RF, CNN, ANN, and Adaboost, among others, demonstrating significant advancements in the accuracy and efficiency of leak detection. For instance, Random Forest achieved an impressive 100% accuracy for metal pipes and 94.93% for non-metal pipes using acceleration data, while Adaboost attained a 95.8% validation accuracy with acoustic emission signals. The integration of hybrid techniques is also notable, as seen in

the use of Genetic Algorithms (GA) with ANN and wavelet decomposition with Random Forest, which substantially enhanced the models' performance.

However, a critical analysis of the table suggests a significant research gap in accurately determining the exact type of leakage (e.g., pinhole, gasket, or burst leaks) rather than merely classifying leaks as present or absent. None of the studies explicitly addressed the classification of specific leakage types, which is essential for optimizing repair strategies and minimizing water loss effectively. Furthermore, while the use of simulated data is prevalent, there is a noticeable lack of real-time data from actual case studies or controlled laboratory experiments. Addressing these gaps by incorporating real-time, sensor-based data and focusing on classifying specific leakage types could significantly advance the practical implementation of ML-based leakage detection systems in WDS.

Table 2.1: ML-based literature overview of leakage detection, localization and severity assessment of WDS using different types of sensors.

Ref.	Model	Hybridized	Input data	Output	Conclusion	Remarks
[56]	Random Forest	Yes	Acceleration data from MEMS-based accelerometers	Leak/No-Leak	Achieved 100% accuracy for metal pipes and 94.93% for non-metal pipes. Random Forest outperformed other models in detecting leaks effectively.	Used MEMS-based accelerometers; did not mention IQR for outliers removal; used radial sensing direction; data collected from 75 sites in Hong Kong.
[103]	Random Forest	Yes	Pressure data transformed via wavelet decomposition	Leak/No-Leak	Achieved 99% accuracy with precision of 0.99 and recall of 1.0. Wavelet decomposition significantly improved model performance.	Used pressure sensors; did not mention IQR for outliers removal; used wavelet decomposition for feature extraction; tested on L-Town WDS simulation.
[83]	Convolutional Neural Network (CNN)	No	Simulated pressure data from L-Town network	Leak location and size	Achieved precision up to 98.9% and recall up to 97.4% for no-noise scenario. CNN outperformed MLP in all study cases, especially for noisy data.	Used pressure sensors; did not mention IQR for outliers removal; accounted for measurement device imprecision and hydraulic model uncertainties.
[105]	Convolutional Neural Network (CNN)	No	Simulated pressure and flow data	Leak location and size	Achieved 99% precision and 98.9% recall in no-noise cases. CNN outperformed MLP in handling noise and multiple leak scenarios effectively.	Used pressure and flow sensors; did not mention IQR for outliers removal; accounted for hydraulic model uncertainties and measurement device imprecision.
[106]	Artificial Neural Network (ANN)	Yes	Pressure data from smart meters	Leakage severity and location	Achieved MAPE of 11.78% for leakage severity prediction with a Pearson's correlation coefficient of 0.9626. ANN integrated with GA outperformed other models.	Used pressure sensors; did not mention IQR for outliers removal; employed genetic algorithms (GA) for optimization; tested on Hanoi benchmark WDS.
[107]	Fuzzy-Based Methodology	No	Pressure and flow data	Leak location and severity	Achieved detection accuracy above 90% with effective location identification using ILP (Index of Leakage Propensity).	Used pressure and flow sensors; did not mention IQR for outliers removal; utilized fuzzy set theory to handle uncertainties in data; implemented using MATLAB and EPANET.

[108]	Feature-Based Clustering	No	Pressure data	Recurring pressure anomalies	Achieved F1-score of 93.5% for detecting recurring patterns with high precision and recall. Feature-based clustering outperformed instance-based methods.	Used pressure sensors; did not mention IQR for outliers removal; implemented unsupervised learning for proactive leakage control; applied to Dutch WDS data sets from 2013 and 2017.
[109]	Random ID3 Decision Forest Classifier	Yes	Pressure data	Leak/No-Leak	Achieved 99% classification accuracy and 99% leak detection accuracy with reduced error rate of 1%. Outperformed 1D-CNN-SVM and ACO models.	Used pressure sensors; did not mention IQR for outliers removal; used Shuffled Frog Leaping Optimization (SFLO) for sensor placement; tested on EPANET-simulated WDS.
[110]	TDR-MI Based Wireless Sensor Network	Yes	Pressure residuals and sensor data	Leak location and severity	Achieved 98.5% accuracy in leak detection and 97.8% accuracy in localization using K-means clustering and classifiers. Outperformed linear and greedy algorithms.	Used pressure sensors and hybrid TDR-MI system; did not mention IQR for outliers removal; integrated ad-hoc WSNs and K-means for clustering; tested on underground pipelines.
[17]	Adaboost	Yes	Acoustic emission signals (frequency and MFCC features)	Leak/No-Leak	Achieved 95.8% validation accuracy, 92.8% precision, and 99.52% recall. Adaboost outperformed Decision Tree and Random Forest with lowest false positive rate of 7.35%.	Used acoustic sensors; did not mention IQR for outliers removal; utilized MFCC, spectral flatness, and roll-off rate features; data collected from multiple Chinese cities.
[111]	Model-Based Method	No	Pressure data from multiple sensors	Leak location	Achieved 97% localization accuracy within candidate regions, with about 74.5% of leaks detected in the first candidate region. Improved accuracy by 50% using dominant sensors.	Used pressure sensors; did not mention IQR for outliers removal; developed dominant sensor sequences for accurate localization; tested on a WDS in Zhejiang, China.

2.5 Gap Recognition and Motivation

A recent study reveals that MEMS based technologies are transforming leak detection methods by enabling real-time monitoring using pressure, accelerometer, and hydrophone sensors [44]. MEMS accelerometers and hydrophones, in particular, are gaining popularity for their ability to detect even minor leaks by capturing subtle vibrations and sound signals along pipelines [112]. Researchers report that conventional acoustic leak loggers often suffer from high costs and false alarms, which has led to a shift toward MEMS-based sensors that can be integrated into wireless networks for continuous monitoring [50], [113]. Studies also show that MEMS-based pressure and vibration sensors can significantly reduce detection times and enhance localization accuracy by using advanced signal processing techniques [114]. The consensus in the literature is that MEMS-based pressure, acoustic, and vibration sensors are emerging as essential, cost-effective tools for improving leak detection efficiency [48].

Many studies report that vibration (accelerometers), acoustic sensors, and pressure sensors are the most commonly deployed for leak detection due to their affordability and ease of installation [41]. Some researchers argue that traditional pressure and flow sensors, while effective for detecting large pipeline bursts, may miss smaller leaks that do not significantly impact pressure levels [115]. In contrast, vibration and acoustic sensors are highly sensitive to minor leak signatures, making them indispensable for detecting early-stage leaks that could otherwise go unnoticed [116]. Evidence also suggests a growing adoption of accelerometer-based leak sensors due to their mass production, low power requirements, and cost-effectiveness [112]. The literature predicts that future leak detection

systems will heavily rely on a combination of these sensors paired with machine learning and IoT technologies for near real-time leak alerts [117], [118].

The consensus in the literature is that pressure, acoustic, and vibration sensors are essential for effective leak detection in water distribution systems [119], [120]. Pressure sensors provide broad coverage by detecting major leaks through pressure drops and unexpected flow increases, while acoustic and vibration sensors capture the physical phenomena of leaks – the sound and vibrations caused by water escaping from pipes [64], [121]. Research indicates that by leveraging both hydraulic and acoustic data, utilities can detect leaks more reliably and pinpoint their locations with greater accuracy [122], [123]. Most studies conclude that a multi-sensor approach, combining pressure, acoustic, and vibration data, is fundamental to minimizing water loss and enhancing the sustainability of water supply networks [124], [125], [126].

Despite substantial progress, several gaps and open challenges remain in current leak detection research. To date, few studies tackle leak type identification and severity assessment together in an integrated model. Most prior works focus on either detecting leaks and perhaps classifying the leak type or on estimating the leak size (severity), but not both simultaneously. For instance, experiments might classify among different leak types (longitudinal crack, hole, etc.) under one fixed leak size, or predict leak size in a simplified scenario without distinguishing the failure mode. The absence of a unified approach means that a field-deployed system might detect a leak but not know its nature or urgency, or vice versa. This is a notable research gap: an ideal leak diagnosis system should concurrently determine if a leak is present, what type of leak it is, and how severe it is, all in one framework. Addressing this gap would significantly enhance decision support for water

utilities, yet the literature so far offers very limited examples combining all these classification tasks.

Another limitation is the lack of multi-sensor frameworks that fuse different sources of information for leak detection (Multi-Sensor Data Fusion). The majority of studies use a single sensor type (either pressure-only or acoustic-only data) when developing ML models. While these single-sensor approaches have shown success, they may miss complementary information that another sensor could provide. Recent work has hinted at the benefits of combining sensors – for example, using both a hydrophone and an accelerometer together to capture both fluid-borne and structure-borne leak signals – but comprehensive multi-sensor studies are still scarce. Aghashahi *et al.* have released a dataset with synchronized pressure, accelerometer, and hydrophone readings for multiple leak scenarios, which is a step toward multi-sensor analysis. However, there is a research gap in developing ML models that effectively integrate heterogeneous sensor data (pressure, sound, vibration, etc.) to improve detection robustness. Exploring sensor fusion and optimal sensor placement could greatly enhance leak localization and characterization, yet few published works have ventured into multi-sensor leak detection in real or large-scale networks.

Leak detection datasets often contain significant noise and outliers due to sensor faults, spurious events, or variability in demand. However, there is limited use of systematic outlier removal techniques in current studies. Data preprocessing is sometimes overlooked, which can degrade model performance. Notably, the interquartile range (IQR) method for outlier filtering – a simple yet effective technique to eliminate extreme readings – has seen little application in leak detection research. Only recently have some authors started to

mention its use: for example, Chikodili *et al.* applied z-score and IQR rules to clean a leak dataset, removing anomalous points before model training. Such practices are not yet common, as many works either assume clean simulated data or do minimal filtering. The potential impact of rigorous outlier removal on leak prediction accuracy is therefore not well documented. This gap suggests an opportunity: employing statistical outlier detection (like IQR filtering) could improve ML model reliability by preventing a few extreme noisy samples from skewing the results. Future research should investigate preprocessing steps (including outlier removal, normalization, denoising) and quantify how they affect leak detection performance – an aspect currently underreported in the literature.

Additionally, a broader gap in the field is handling the uncertainties inherent in water networks – such as variable demand patterns, changing noise levels, and even multiple simultaneous leaks – which many current ML models do not explicitly address. Most studies train and test models under relatively controlled conditions (steady flows, single-leak scenarios, known boundary conditions). There is a need for research that considers these uncertainties and assesses model robustness in more realistic, dynamic network conditions. This gap ties into the motivation for more comprehensive and resilient leak detection frameworks, as discussed next.

In light of the above gaps, current research efforts are motivated to develop more integrated ML-based leak detection systems for water distribution networks. Researchers aim to design unified models that can simultaneously detect a leak, classify its type, and estimate its severity. Instead of treating these as separate problems, the goal is a one-stop ML solution that takes sensor inputs and outputs a full diagnosis (e.g. “severe orifice leak” vs “minor joint leak”). Achieving this entails creating multiclass or multi-output models

trained on richly labeled data covering different leak types and sizes. By combining tasks, such models could leverage shared features – for example, certain frequency components might indicate both the presence of a leak and hint at the leak type. Early work in this direction is leveraging datasets with multiple leak classes (like longitudinal crack, circumferential crack, etc., at various demand flow rates) to train advanced classifiers and assess how well a single model can differentiate both leak existence and characteristics. This integrated approach is expected to fill the gap of dual leak characterization and provide more actionable information to utilities.

There is a strong motivation to utilize MEMS-based sensors (small, low-power accelerometers, pressure sensors, etc.) in deployments that capture a wide range of leak scenarios. Current research is conducting extensive data collection campaigns using MEMS sensor arrays on test networks to gather signals for different leak types (LC, CC, OL, GL) under various conditions. The focus is on evaluating how well these inexpensive sensors perform across different leak modalities and whether their data can be combined for better results. For instance, an accelerometer might detect a leak’s vibration signature while a pressure transducer simultaneously records a transient; fusing these could improve confidence in detection. Studies are also exploring optimal sensor placement strategies in looped and branched networks so that MEMS sensors can effectively capture leak signatures throughout the system. The use of MEMS technology is driven by practical deployment considerations – these sensors enable creating an Internet of Things (IoT) for water pipes, where many nodes collect data that ML algorithms analyze in real time. The motivation is that a dense, MEMS-based sensor network can greatly reduce leak detection

time and labor, provided that the ML models can intelligently interpret the massive, heterogeneous data generated.

New research is placing emphasis on quantifying and handling uncertainty in leak detection. This includes accounting for fluctuations in water demand, pressure variability, environmental noise, and modeling errors in the ML algorithms. One approach is to train ML models on datasets that incorporate realistic variations – for example, adding different levels of background noise or simulating demand changes while leaks occur. By exposing models to a wide range of conditions (including sensor noise and drift), researchers aim to make leak detection algorithms more robust to false alarms and missing data. Another aspect is using probabilistic ML methods or uncertainty estimation techniques (like Bayesian neural networks or ensemble variance) to gauge the confidence of leak predictions. The motivation behind this is to allow water utilities to not only get a point prediction (“leak” or “no leak”) but also an uncertainty measure – so they can, for example, prioritize investigating alerts that the model flags with high confidence. Overall, acknowledging and modeling uncertainty is a current direction to bridge the gap between idealized lab studies and the messy reality of operational water networks.

The role of data preprocessing in enhancing machine learning model performance for leak detection has attracted growing attention, particularly regarding the treatment of outliers. The interquartile range (IQR) method is often employed to filter out extreme data points assumed to be anomalies. While some preliminary studies have reported that such filtering may enhance model generalization by reducing the influence of noise and spurious readings, these conclusions are not yet definitive. In fact, no conclusive evidence currently confirms whether outlier removal via IQR consistently improves or compromises model

performance, especially in the context of sensor data, which may contain important but rare event signatures mistaken for outliers. Moreover, certain studies have adopted outlier filtering as a standard preprocessing step without critically assessing its potential to discard meaningful leakage-related information, particularly in complex or real-world water network scenarios. Therefore, further dedicated investigation is essential to determine the true impact of IQR-based outlier removal on model accuracy and reliability in leak detection tasks. Until such studies are conducted, the use of this technique should be approached with careful consideration of the trade-off between noise reduction and information preservation.

Finally, an important research thrust is the comparative evaluation of deep learning models versus tree-based ensemble models for leak detection and localization. As discussed, both approaches have merits, and current studies aim to determine which is better suited under various conditions. For example, one research effort is benchmarking LSTM, RNN, and GRU models against ensemble methods like XGB, RF, and LightGBM on the same leak datasets to assess performance trade-offs. Metrics such as detection accuracy, false alarm rate, training time, and computational requirements are compared. Early findings indicate that deep neural networks may excel when dealing with high-resolution raw sensor signals or when automatic feature learning is crucial, whereas tree ensembles can rival them on engineered features and offer faster inference on edge devices. By systematically comparing these model classes, researchers are identifying hybrid strategies (e.g. using a deep model to extract features feeding into a gradient boosting classifier) that capitalize on both. The motivation behind this comparative research is pragmatic: to guide water utilities in choosing the most accurate, efficient, and robust ML

model for their leak detection system. Understanding the strengths of deep learning vs. ensemble methods will help tailor solutions to different deployment scenarios (for instance, a lightweight edge device might favor a simpler ensemble model, while a cloud-based analytic platform could leverage a deeper network). This comparative evaluation is expected to continue as more advanced algorithms emerge, ensuring that the water industry adopts the best available tools for leak management.

In summary, the recognized gaps can be drafted as follows:

1. There is a lack of unified models that can simultaneously detect leaks, classify their type, and estimate their severity in an integrated framework.
2. Multi-sensor data fusion is underexplored, with most studies relying on single-sensor inputs rather than combining complementary sensor data for improved detection.
3. Outlier detection and data preprocessing practices are inconsistent, with limited research on how techniques like IQR filtering impact model accuracy and robustness.
4. Most machine learning models fail to quantify uncertainty in predictions, limiting their reliability under real-world, variable network conditions.
5. Leak detection models are rarely tested under realistic, dynamic scenarios such as fluctuating demand, multiple leaks, or noisy environments.
6. There is a lack of systematic comparison between deep learning and ensemble-based models for leak detection under various operational constraints.
7. The real-world performance and limitations of MEMS sensors across various leak scenarios need further empirical validation.

8. Datasets with labels for both leak type and severity are rare, hindering the development of comprehensive diagnostic models.

The application of machine learning to MEMS-based sensor data is revolutionizing leak detection in water distribution networks. This literature review has highlighted how various sensors and models are used to detect leaks, classify leak types, and even gauge leak severity. It is evident that integrated approaches – combining multiple sensor inputs, classifying both leak type and severity, and employing robust algorithms – are the frontier of current research. Significant gaps, such as the lack of combined leak characterization studies and limited attention to data preprocessing, are now being actively addressed by the community. By leveraging advanced ML techniques (from ensemble learning to deep neural networks) and focusing on comprehensive frameworks, researchers aim to greatly improve the speed and accuracy of leak detection in both looped and branched WDNs. The continued development of these data-driven leak detection systems holds promise for reducing water losses and enhancing the sustainability of water supply networks worldwide.

This study aims to address the identified research gaps, specifically gaps 1, 2, 3, and 6, by providing a comprehensive contribution in the field of WDS leak detection. The primary objective is to develop ML models that simulate an integrated approach capable of both determining the type of leakage and assessing its severity. This integrated framework is intended to support improved decision-making and contribute to reducing the environmental impact on already scarce water resources. To support this objective, the study also investigates the effect of outlier removal on the performance of multi-sensor frameworks developed from a controlled laboratory-scale WDS experiment.

CHAPTER 3

MATERIALS AND METHODOLOGY

3.1 Data Description

The benchmarked dataset comprises 280 signals collected from a laboratory-scale water distribution system designed to replicate various leakage scenarios [76]. This WDS employs 152.4 mm (6 inches) diameter PVC pipes to capture a wide range of variables under controlled conditions. The diversity of signals allows for a thorough analysis of different leakage types and characteristics in water distribution systems.

- **Leak Types**

The data includes five specific leak types induced in the system to simulate common leak scenarios. The *Longitudinal Crack (LC)* involves a crack extending along the pipe's length, which affects the pressure and flow in a distinct pattern compared to other types. The *Circumferential Crack (CC)*, encircling the pipe, creates a different flow and pressure dynamic due to its shape. The *Gasket Leak (GL)* simulates leaks at pipe connections or gaskets, representing one of the most common real-world issues in distribution systems. Additionally, the *Orifice Leak (OL)* resembles a puncture or drilled hole, further diversifying the range of leakage behaviors. For comparative purposes, *Nonleak (NL)* cases are also included as baseline scenarios, capturing system behavior in the absence of any leaks.

- **Network Architecture**

Two different network architectures were used in this dataset namely, *Looped (LO)* and *Branched (BR)*, each introducing unique characteristics to the flow and pressure patterns. The Looped architecture offers a closed network where water flows along multiple paths, creating a complex distribution system where leak detection may be more challenging due to the redundancy in flow paths. The Branched architecture, in contrast, consists of a single linear path to each endpoint, which can provide simpler yet distinctive patterns in pressure and flow behavior when leaks occur. The combination of these architectures provides an opportunity to examine how network design influences leak characteristics and detection.

- **Water Demand Flow**

To simulate realistic variations in water consumption within a residential distribution system, background flow rates were established based on a hypothetical service to approximately 100 individuals. The total estimated water demand under standard conditions was around 0.44 L/s. To reflect typical diurnal fluctuations in usage, two flow rates were selected: 0.18 L/s and 0.47 L/s. The 0.18 L/s condition corresponds to moderate demand, representative of off-peak hours (e.g., early morning around 1:00 AM) when household water usage is reduced, but not entirely absent. In contrast, the 0.47 L/s rate simulates a higher demand scenario, such as the early morning rush period around 5:00 AM, when water usage typically increases due to morning routines. These demand levels were chosen not only to capture real-world variability but also to ensure that leak detection could be evaluated under different flow dynamics and signal-to-noise conditions.

Controlled water demand flow was implemented with two steady flow rates of 0.18 L/s and 0.47 L/s , representing lower and higher flow levels within the system. These flow rates provide data on how leaks manifest under various conditions of water demand. Additionally, a *transient condition* was introduced, where the flow rate abruptly changes from 0.47 L/s to 0 L/s . This sudden pressure variation helps simulate real-life scenarios where leaks could exacerbate due to abrupt changes in water usage or pressure, offering insights into the system's response to such conditions.

- **Background Sound Conditions**

Background sound conditions were varied to assess how ambient noise influences the detection and characterization of leaks. Two settings were used, *Sound Present (S)*, in which ambient noise is introduced to simulate real-world environmental conditions, such as urban or industrial background noise, which may mask subtle acoustic leak signals; and *No Sound (NS)*, where ambient noise is absent, providing a controlled environment for detecting leak signatures with minimal interference. This setup allows the dataset to encompass both ideal conditions and realistic scenarios with potential noise interference, reflecting the operational challenges of leak detection.

- **Sensor Types and Data Acquisition**

Data were collected through three sensor types: *accelerometers (A)*, *hydrophones (H)*, and *dynamic pressure transducers (P)*. Accelerometers measured vibration data, capturing mechanical disturbances caused by leaks or pressure fluctuations. Hydrophones recorded acoustic data, detecting sound variations that indicates leak

presence, with data stored in RAW format at an 8000 Hz sampling frequency to preserve high fidelity in sound detail. Dynamic pressure transducers measured fluctuations in pressure, providing direct insight into pressure anomalies caused by leak-induced flow disturbances. Each sensor type was strategically positioned in multiple locations across the looped and branched networks, capturing spatial variations in data that enrich the dataset with localized leak insights.

Figure 3.1 and Figure 3.2 show the experimental layout of the tested WDS networks. Figure 3.1a presents the perspective view of looped network from the laboratory proposed in the benchmark of the dataset while Figure 3.1b presents the schematic view of looped topology highlighting the main components of WDS and the location of the sensors.



Figure 3.1: Experimental layout of looped WDS topology in the laboratory where (a) real-time perspective view (b) schematic view.

Similarly, Figure 3.2a shows the real-time implementation of branched WDS network in the laboratory of the used dataset's study. Which is then schematically shown in Figure 3.2b where the dimensions of the network are illustrated and the components used in the WDS including the location, type and number of sensors.

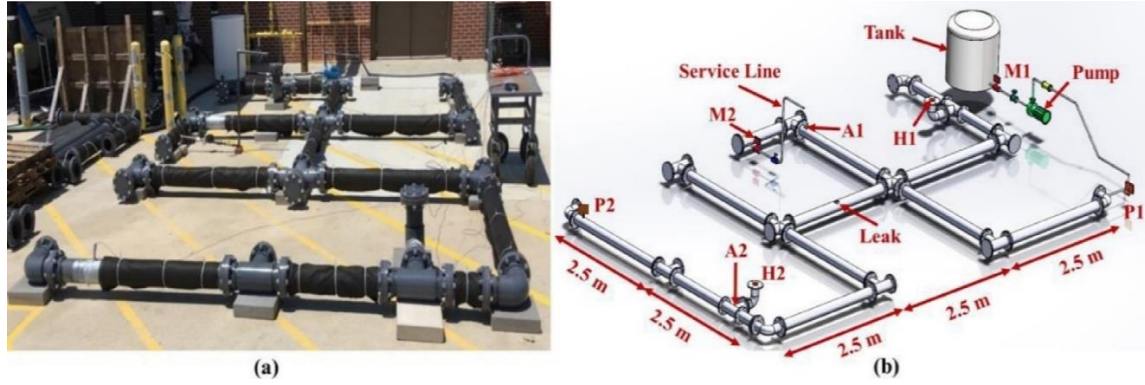


Figure 3.2: Experimental layout of branched WDS topology in the laboratory where (a) real-time perspective view (b) schematic view.

Each sensor signal was recorded over a 30-second interval, providing sufficient temporal data to capture steady-state conditions or transient events, ensuring consistent recording duration across varied conditions and enabling reliable comparative analysis. Sensor data were stored in different formats suitable for each sensor type: *accelerometer* and *dynamic pressure transducer* data in CSV format, offering easy access for processing, and *hydrophone* data in RAW format, which can be converted as needed for broader compatibility. Figure 3.3 visually shows the combination of these variations in leak type, network architecture, water demand, background sound conditions, and sensor locations. This combination establishes a robust dataset with diverse scenarios to facilitate an in-depth analysis of leak characteristics under controlled yet realistic conditions.

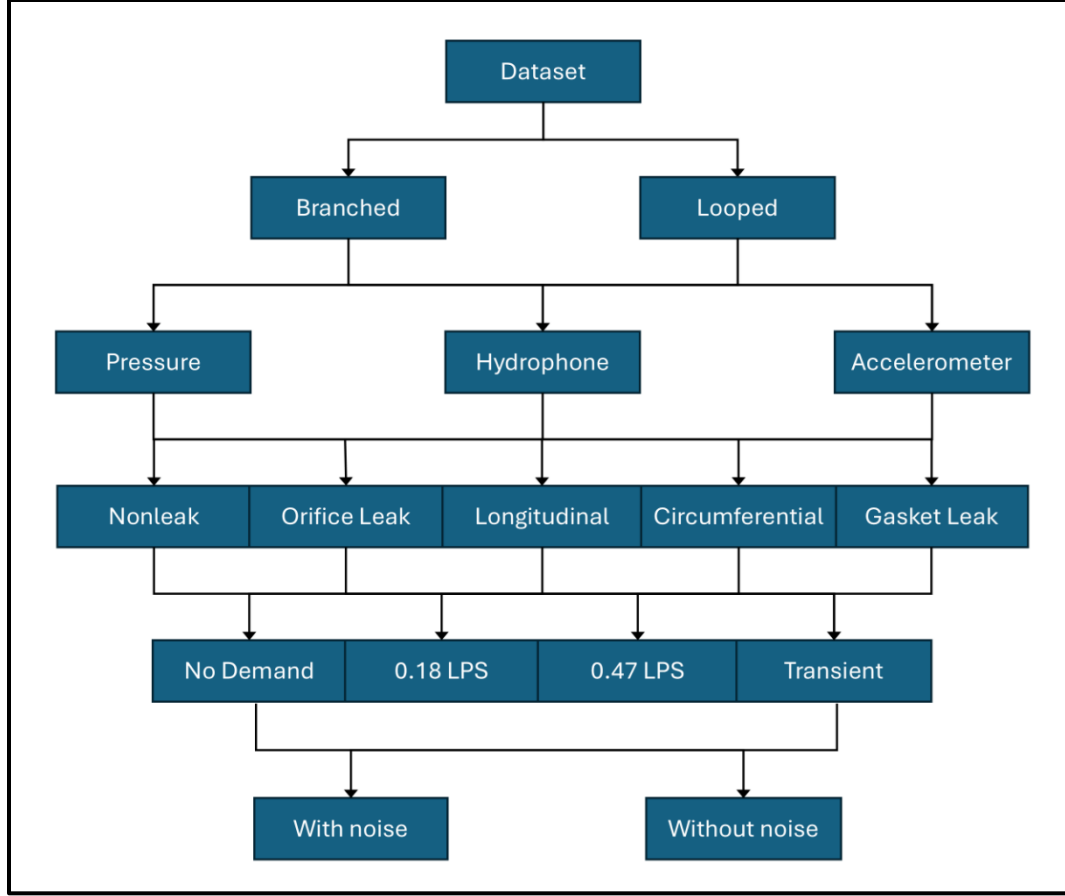


Figure 3.3: systematic presenting all possible pathways of the available WDS dataset, where each possible pathway represents two CSV/RAW files that include corresponding sensors readings (Sensor1, Sensor2).

The recorded signals encapsulated variations due to changes in network topology, background noise, and leak types, making the dataset highly representative of real-world leak detection scenarios. Each experiment was carefully labeled using a standardized naming convention to encode key experimental parameters, including network topology, leak type, background flow conditions, and sensor type. For instance, accelerometer and dynamic pressure sensor files followed the naming convention "T_L_F_S#," where "T" represented topology, "L" denoted leak type, "F" indicated background flow condition, and "S#" specified the sensor number. Hydrophone files incorporated an additional background

noise condition indicator ("B") to account for ambient noise levels. This structured organization of the dataset enhances its usability for researchers seeking to develop and validate leak detection algorithms. Table 3.1 gives a clear summary of all possible values for each of the specified naming convention characters. For example, the readings collected by hydrophones when there was a transient demand, no leakage in the system, and a looped layout with background noise are stored in the RAW file as: "LO_NL_Transient_S_H1" for the first hydrophone and "LO_NL_Transient_S_H2" for the second hydrophone. Hence the features in leakage type determination and severity assessment tasks followed the same naming pattern.

Table 3.1: All possible values of naming convention characters followed to name CSV and RAW files in the dataset.

T	L	F	S#	B
BR	CC	ND	A1, A2	NS
LO	GL	0.18 LPS	P1, P2	S
-	LC	0.47 LPS	H1, H2	-
-	NL	Transient	-	-
-	OL	-	-	-

In summary, the experimental procedure ensured the collection of high-quality data suitable for benchmarking leak detection methods in water distribution systems. The use of a meticulously designed testbed, high-resolution sensors, and a robust data acquisition system contributed to the dataset's reliability. The distinction between CSV and RAW file formats based on sensor type preserved data integrity while allowing for diverse analytical approaches. The systematic variation of leak conditions, network configurations, and background noise levels provided a comprehensive dataset capable of supporting a wide range of analytical and ML applications in leak detection and localization research.

3.2 Deep Learning Algorithms

3.2.1 Recurrent Neural Networks

In RNN-based classification mechanism of this study, we employ multiple SimpleRNN layers stacked to extract hierarchical temporal patterns [127], [128]. The network takes in sequential input data, processes it through recurrent layers, and outputs a probability distribution over classes. The overall mechanism of standard RNN models is shown in Figure 3.4.

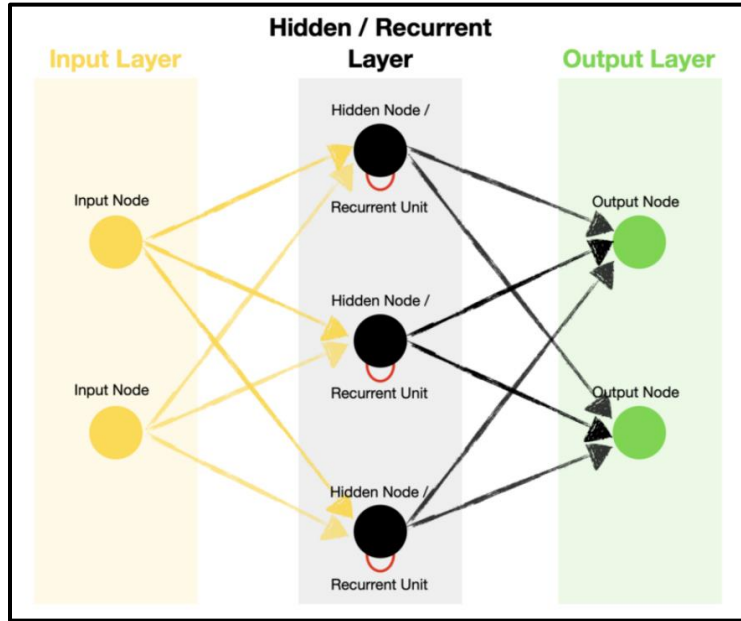


Figure 3.4: schematic diagram showing the overall learning mechanism of RNN model which consists of Input layer, Hidden SimpleRNN layers, followed by Output layer.

The input to the RNN consists of a sequence $X = [x_1, x_2, \dots, x_T]$, where each $x_t \in R^d$ represents a feature vector at time step t . The hidden state h_t at each time step is computed using Eq. (1).

$$h_t = \text{ReLU}(Wh_{t-1} + Ix_t) \quad (1)$$

where W is the recurrent weight matrix, I is the input projection matrix, and $\text{ReLU}(\cdot)$ is the Rectified Linear Unit activation function [129].

Each SimpleRNN layer in our architecture follows this update rule, with dropout layers included to mitigate overfitting. Stacking multiple RNN layers allows the network to learn more complex hierarchical representations. The hidden state at layer l is computed as per Eq. (2):

$$h_t^l = \text{ReLU}(Wh_{t-1}^l + Ih_t^{l-1}) \quad (2)$$

This formulation allows information to be processed through multiple hierarchical layers before classification.

After the final RNN layer, the last hidden state is passed through a dense layer computed as in Eq. (3):

$$z = \text{ReLU}(W_{dense} h_T + b_{dense}) \quad (3)$$

where z is the feature representation, W_{dense} and b_{dense} are learned parameters.

Finally, a softmax function is applied to obtain the class probabilities as declared in Eq. (4):

$$y = \text{softmax}(W_{out} z + b_{out}) \quad (4)$$

where y is the predicted class probability distribution, W_{out} and b_{out} are the output layer parameters [130].

The network is trained using the Adam optimizer with a learning rate of 0.0001. The loss function employed is sparse categorical cross-entropy, defined as Eq. (5):

$$\mathcal{L} = - \sum_{i=1}^N y_i \log(\hat{y}_i) \quad (5)$$

Where y_i is the ground truth label, and $\hat{y}_i$ is the predicted probability of class i .

The model is trained for 15 epochs with a batch size of 1500. The classification mechanism in the RNN model employed in this study follows a structured process: sequential feature extraction through stacked RNN layers, feature refinement via a dense layer, and final classification using a softmax layer. This architecture enables the model to learn rich temporal dependencies, making it well-suited for sequential classification tasks [129].

3.2.2 Long Short-Term Memory

Long Short-Term Memory (LSTM) networks are an advanced form of Recurrent Neural Networks (RNNs) designed to handle long-range dependencies in sequential data. Unlike standard RNNs, LSTMs utilize a gating mechanism to regulate the flow of information, preventing issues like the vanishing gradient problem [131], [132].

In our LSTM classification model, we use a stacked architecture of four LSTM layers followed by fully connected layers to make predictions. This structure enables the network to capture hierarchical temporal dependencies while learning complex patterns from sequential data [133].

The input to the RNN consists of a sequence $X = [x_1, x_2, \dots, x_T]$, where each $x_t \in R^d$ represents a feature vector at time step t . The LSTM maintains a hidden state h_t and a cell state c_t , which evolve over time as in Eq. (6). Unlike simple RNNs, LSTMs employ gates to control the flow of information [134].

$$h_t, c_t = LSTM(x_t, h_{t-1}, c_{t-1}) \quad (6)$$

At each time step t , the LSTM cell computes the following as detailed in Table 3.2, these equations ensure that the network selectively remembers useful information while forgetting irrelevant details [135]:

- **Forget Gate:** Determines how much of the previous cell state should be forgotten.
- **Input Gate:** Decides what new information to store in the cell state.
- **Cell State Update:** Updates the cell state based on the forget and input gates.
- **Output Gate:** Determines the new hidden state.

Table 3.2: LSTM time-step formulations tabulated for three gates (Forget Gate, Input Gate, and Output Gate) in addition to Cell State update

Forget Gate	$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$
Input Gate	$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$ $\tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c)$
Cell State Update	$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$
Output Gate	$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$ $h_t = o_t \odot \tanh(c_t)$

Where $\sigma(\cdot)$ is the sigmoid activation function, $\tanh(\cdot)$ is the hyperbolic tangent function, $\odot$ represents element-wise multiplication. Figure 3.5 visualizes the process of computations in LSTM model in each time step.

LSTM function of this study employs four stacked LSTM layers to extract hierarchical features. The hidden state at layer l is computed in Eq. (7):

$$h_t^l, c_t^l = LSTM^l(h_t^{l-1}, c_t^{l-1}) \quad (7)$$

Where h_t^{l-1} is the hidden state from the previous layer, and c_t^{l-1} is the cell state from the previous layer. By stacking multiple LSTM layers, the model can capture both low-level and high-level sequential features.

Similar to RNN, After the final LSTM layer, the last hidden state is passed through a fully connected dense layer and a softmax layer that converts the output into class probabilities as shown in Eq. (03) and Eq. (04) respectively. The network is trained using the Adam optimizer with a learning rate of 0.0001. The loss function is sparse categorical cross-entropy, given by Eq. (05). The model is trained for 15 epochs with a batch size of 1500, ensuring stable convergence.

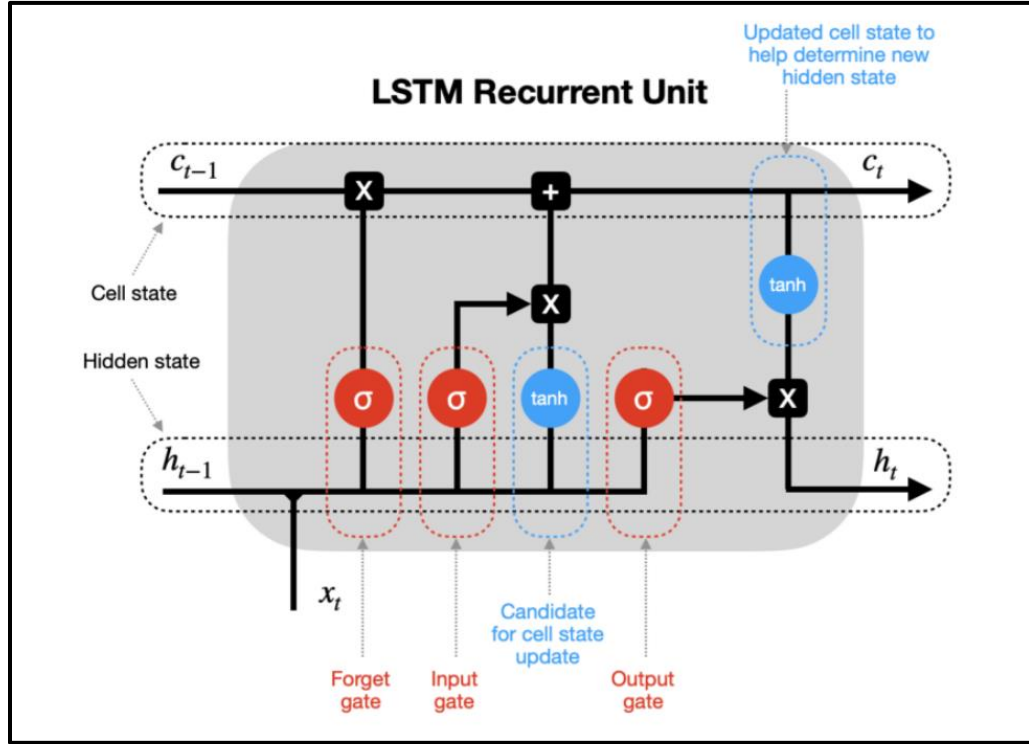


Figure 3.5: visualization of the calculations occurring inside of each LSTM layer in a multilayer LSTM ML model.

3.2.3 Gated Recurrent Unit

Gated Recurrent Units (GRUs) are an improvement over traditional recurrent neural networks (RNNs), designed to address the vanishing gradient problem while simplifying the architecture compared to Long Short-Term Memory (LSTM) networks. Introduced by

Cho *et al.*, GRUs incorporate gating mechanisms that regulate information flow within the network, allowing them to efficiently capture long-term dependencies [127].

At each time step t , a GRU maintains a hidden vector h and updates it based on the previous hidden state and current input. The computations, as defined by Cho *et al.*, are given as in the group of formulations (8):

$$\begin{aligned} r_t &= \sigma(W_r x_t + U_r h_{t-1}) \\ z_t &= \sigma(W_z x_t + U_z h_{t-1}) \\ h'_t &= \tanh(W x_t + U(r_t \odot h_{t-1})) \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot h'_t \end{aligned} \tag{8}$$

Where z_t is the update gate determining how much of the past hidden state h_{t-1} should be carried forward, r_t is the reset gate controlling how much of the previous hidden state contributes to the candidate activation, h'_t represents the candidate hidden state computed using the reset gate, h_t is the final hidden state at time t computed as an interpolation between the past hidden state and the candidate activation. W_r, W_z , and W are projection matrices applied to the input x_t . U_r , U_z , and U are recurrent weight matrices applied to the previous hidden state h_{t-1} [136].

Unlike LSTMs, GRUs use fewer gating mechanisms, making them computationally efficient while still being capable of capturing long-range dependencies [137], [138]. The absence of a separate memory cell simplifies the model while retaining effectiveness in sequential tasks [139], [140]. Figure 3.6 represents the gating mechanism of each time step in GRU model.

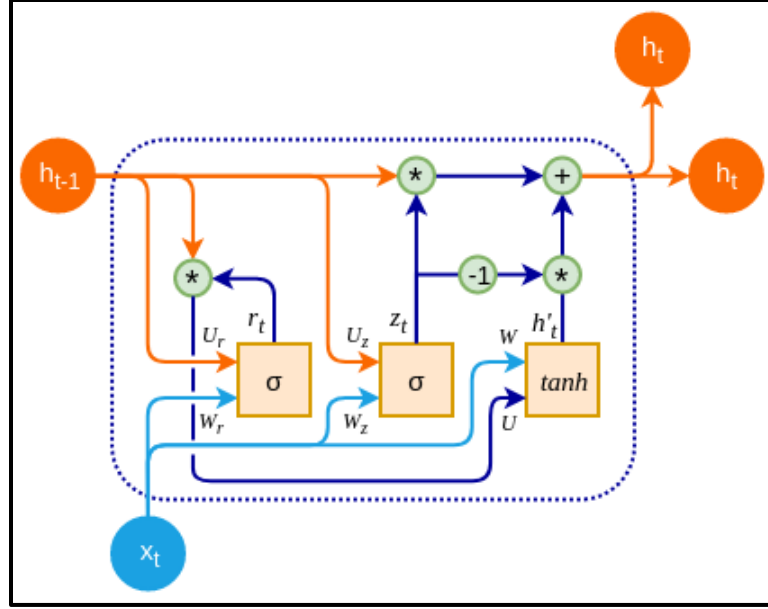


Figure 3.6: visualization of the calculations occurring inside of each GRU layer in a multilayer GRU ML model.

3.3 Ensemble Algorithms

3.3.1 Extreme Gradient Boosting

Extreme Gradient Boosting is a powerful ensemble learning algorithm that builds a strong classifier through iterative improvements on weak learners [141], [142]. It is based on gradient boosting, an optimization technique that minimizes a specific loss function by sequentially adding decision trees. The process involves weighted adjustments to the training data, ensuring that subsequent trees focus on the misclassified instances, as depicted in Figure 3.7 [143].

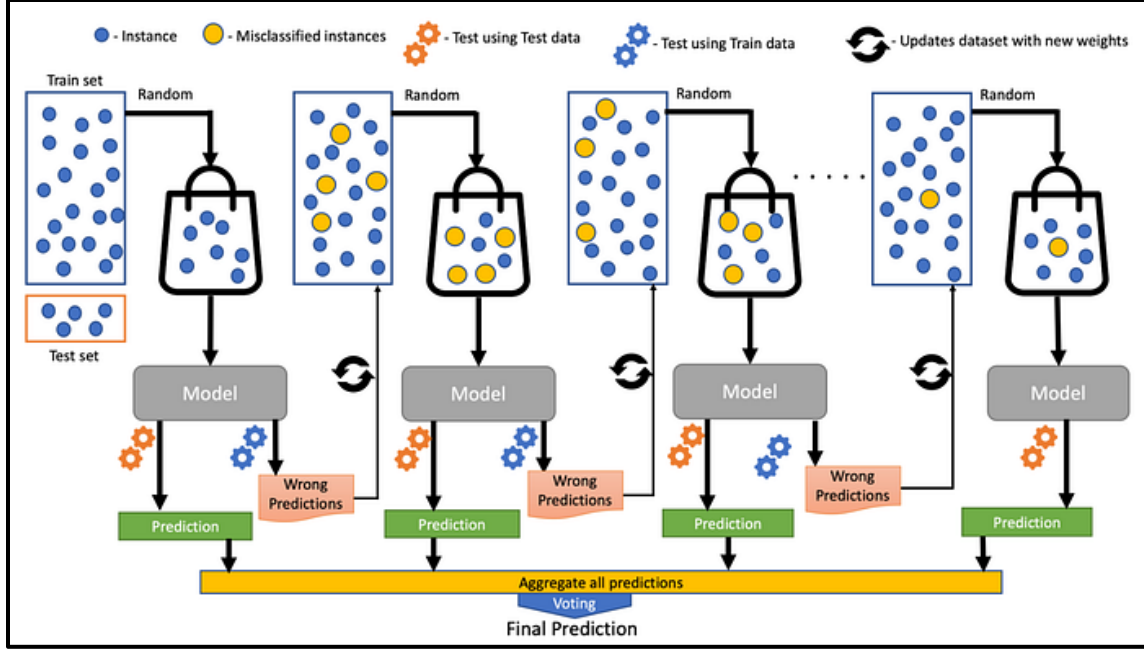


Figure 3.7: visual presentation of the classification learning mechanism of XGB classifier model starting from the input of training set and ending with the final predictions of training set.

For multi-class classification, XGB requires the target variable to be numerically encoded starting from 0. Given a dataset (X, Y) , where X represents the feature set and Y the categorical target variable, the target labels are transformed as follows in Eq. (9):

$$Y_{\text{encoded}} = \text{factor}(Y) - 1 \quad (9)$$

where the factor function ensures categorical encoding, and subtraction by 1 shifts the labels to start from 0. The dataset is then converted into an optimized data structure called DMatrix, which enhances computational efficiency by storing feature values in a compressed format. Mathematically, the feature matrix and encoded labels are represented in Eq. (10), where each D represents the training, validation, and test datasets, respectively [144].

$$\begin{aligned}
D_{\text{train}} &= (X_{\text{train}}, Y_{\text{encoded}}^{\text{train}}) \\
D_{\text{test}} &= (X_{\text{test}}, Y_{\text{encoded}}^{\text{test}}) \\
D_{\text{val}} &= (X_{\text{val}}, Y_{\text{encoded}}^{\text{val}})
\end{aligned} \tag{10}$$

XGB constructs an ensemble of decision trees in an additive manner, where each tree $h_t(X)$ is trained to correct errors made by the previous ensemble of trees. The function to be optimized is given by Eq. (11)

$$F_T(X) = \sum_{t=1}^T h_t(X) \tag{11}$$

where $F_T(X)$ is the final prediction after T boosting rounds. Each new tree $h_t(X)$ is trained to minimize the residual errors from the previous trees, guided by the gradient of the loss function. Given a loss function L , the optimal tree at iteration t is determined by Eq. (12).

$$h_t(X) = \arg \min_h \sum_{i=1}^N L(Y_i, F_{t-1}(X_i) + h(X_i)) \tag{12}$$

where Y_i is the true label for instance i , and $F_{t-1}(X_i)$ represents the prediction from previous trees. XGB utilizes gradient-based optimization to iteratively refine predictions.

To control model complexity and prevent overfitting, regularization terms are added to the objective function as shown in Eq. (13):

$$\mathcal{L} = \sum_{i=1}^N L(Y_i, F_{t-1}(X_i) + h(X_i)) + \Omega(h) \tag{13}$$

where $\Omega(h)$ is the regularization term that penalizes overly complex trees. Hyperparameters such as maximum depth (d), learning rate (η), subsampling rate (s), and feature sampling rate (f) are configured to fine-tune the model. The parameters used in the training process as listed in Table 3.3:

Table 3.3: values used in the training process of the internal parameters of XGB model in both severity assessment and leakage detection tasks.

Parameter	Value	Description
d	10	Maximum depth of trees
η	0.3	Step size shrinkage to regulate weight updates
s	0.8	Fraction of training samples used per tree
f	0.5	Fraction of features randomly selected per tree
<i>Objective Function</i>	Softmax	Softmax function for multi-class classification

A key aspect of XGB is weight adjustments on misclassified instances. The weight of instance i at iteration t is updated as in Eq. (14):

$$w_i^{(t+1)} = w_i^{(t)} e^{\alpha I(h_t(X_i) \neq Y_i)} \quad (14)$$

where $w_i^{(t)}$ is the instance weight before iteration t , $I(h_t(X_i) \neq Y_i)$ is an indicator function that equals 1 if the instance is misclassified, and α is the learning rate that controls how much weight increases for misclassified samples. Misclassified instances receive higher weights, compelling subsequent trees to focus on correcting these errors. Once the ensemble is trained, predictions are made by aggregating outputs from all trees. The final class label $\hat{Y}$ for a given instance is determined using the arg max function applied to the softmax probabilities, Eq. (15):

$$\hat{Y} = \arg \max_c \sum_{t=1}^T P(Y = c | X, h_t) \quad (15)$$

where $P(Y = c | X, h_t)$ is the probability assigned to class c by tree t . The softmax function ensures that output probabilities sum to 1, as per Eq. (16):

$$P(Y = c | X) = \frac{e^{F_T^c(X)}}{\sum_{c'} e^{F_T^{c'}(X)}} \quad (16)$$

where $F_T^c(X)$ is the accumulated score for class c across all boosting iterations. The final classification decision is based on the class with the highest probability.

The XGB algorithm enhances classification accuracy by iteratively refining decision trees through gradient boosting [145]. It optimally learns from errors using weighted instance adjustments, ensuring improved predictive performance [146]. The schematic effectively illustrates how boosting focuses on difficult samples, and the mathematical formulations provide a structured approach to model optimization [143]. XGB remains a powerful tool for multi-class classification, efficiently handling large-scale data with high accuracy and computational efficiency [147].

3.3.2 *Light Gradient Boosting Machine*

Both Light Gradient Boosting Machine (LightGBM) and XGB are advanced gradient boosting frameworks designed to improve training efficiency and predictive performance [148], [149]. They share several core similarities in their learning mechanisms. Both models employ boosting, an ensemble learning technique that sequentially adds decision trees to minimize a loss function [150], [151]. They optimize the objective function using gradient-based techniques, where the new tree at each iteration t is trained to correct the residual errors from the previous trees [152], [153]. Mathematically, the general objective function for both models is formulated in Eq. (13), but they differ in how they grow trees and handle data.

The primary difference between LightGBM and XGB lies in their tree-growing strategies. XGB follows a level-wise growth strategy, where all nodes at a given depth are expanded before moving to the next depth. This method ensures balanced trees but increases computational cost. In contrast, LightGBM uses a leaf-wise growth strategy, which grows the tree by expanding the leaf with the highest information gain. Given a tree structure T , LightGBM selects the leaf l^* , shown in Eq. (17) that maximizes:

$$l^* = \arg \max_l \left(\sum_{i \in l} g_i \right)^2 / \left(\sum_{i \in l} h_i + \lambda \right) \quad (17)$$

where g_i and h_i are the first and second derivatives of the loss function, respectively, and λ is a regularization parameter. This approach leads to deeper, more complex trees, which can improve accuracy but may risk overfitting [148], [149], [154].

Another key distinction is in feature selection and optimization. XGB enumerates all possible feature splits to find the best split point, whereas LightGBM introduces Gradient-Based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) to speed up training. GOSS retains instances with large gradients (informative samples) while down-sampling instances with small gradients, effectively maintaining model accuracy while reducing data size. If G_i represents the gradient for instance i , LightGBM selects a subset S such that satisfies Eq.(18):

$$S = \{i \mid G_i \geq \tau\} \cup \text{random subset of } \{i \mid G_i < \tau\} \quad (18)$$

where τ is a predefined threshold. This significantly enhances efficiency, especially on large datasets. XGB, on the other hand, relies more on histogram-based split finding but does not implement aggressive sampling strategies like LightGBM. As a result, LightGBM is often faster on large datasets, while XGB may provide better accuracy on smaller or structured datasets due to its exhaustive feature evaluation.

3.3.3 Random Forest

Random Forest is a widely used ensemble learning algorithm that constructs multiple decision trees to enhance classification accuracy. It operates by aggregating the predictions from individual trees, thus reducing variance and improving generalization. Given a dataset (X, Y) , where X represents the input features and Y is the target variable, the classification

process follows a structured approach involving data preparation, model training, prediction, and evaluation.

The process begins with data preprocessing, where the target variable Y is transformed into a categorical factor to ensure proper handling by the classification model. This step is essential in multiclass classification problems to allow the model to distinguish between distinct categorical outcomes effectively. Following this, the training phase employs the bootstrap aggregation (bagging) technique, where multiple bootstrap samples (X^*, Y^*) are created by randomly sampling from the original dataset with replacement. Each bootstrap sample serves as the training set for an individual decision tree, allowing the model to learn different feature interactions from diverse subsets of data. Given a total of K trees, each tree $h_k(X)$ is trained independently, incorporating random feature selection at each split to enhance model diversity and reduce correlation among trees [155], [156].

Once the ensemble of decision trees is constructed, the classification process relies on majority voting to determine the final prediction for a given input X_i . Each tree in the forest produces an independent prediction, and the final classification outcome is derived by selecting the most frequently predicted class across all trees. Mathematically, the predicted class $\hat{Y}$ for a given input is defined as per Eq. (19)

$$\hat{Y} = \arg \max_c \sum_{k=1}^K I(h_k(X) = c) \quad (19)$$

where $I(\cdot)$ is an indicator function that counts the number of votes each class c receives from the K decision trees. This voting mechanism ensures robustness against overfitting and improves the generalization capability of the model. $\arg \max$ (short for "argument of the maximum") is an operation that returns the value of the input variable at which a given

function reaches its maximum. Figure 3.8 shows the simplified overall schematic for the training mechanism of Random Forest.

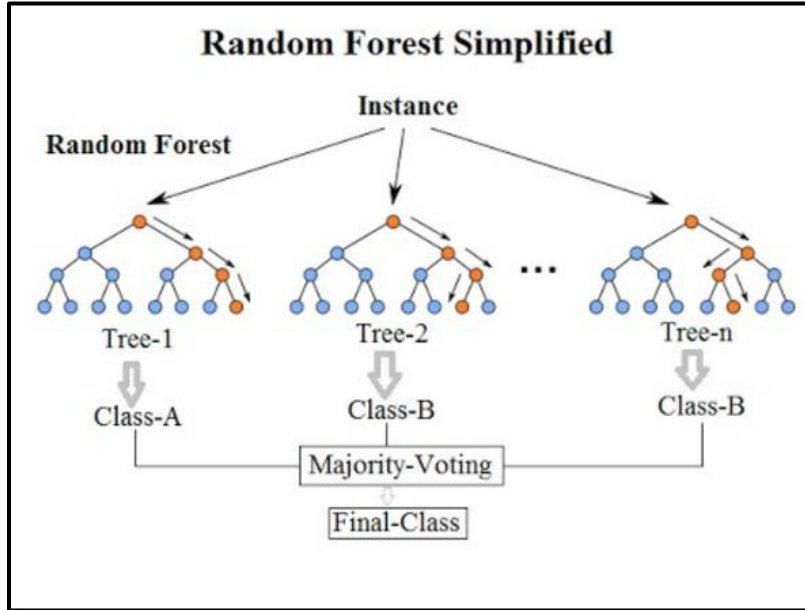


Figure 3.8: Random Forest classifier prediction process visualization.

3.4 Models Development

This study presents a multi-faceted methodology aimed at utilizing sensory data from a controlled water distribution system to predict and classify different types of leaks. By analyzing data collected from dynamic pressure transducers, accelerometers, and hydrophones, we aim to tackle four main objectives: Classifying leak types and assessing leak severity for two types of networks, Branched and Looped. Each objective uses a distinct approach in structuring and processing the data, with specific ML models and optimization algorithms applied to enhance prediction accuracy and efficiency.

To ensure the model generalizes well, the data is split into training and testing sets for each objective, and Validation set when needed. Additionally, preprocessing steps are applied across all objectives, including selecting the initial 25,000 readings from each

sensor, removing outliers using the Interquartile Range (IQR) method, and scaling the data where required by the ML algorithms.

Figure 3.9 presents a structured workflow for processing and analyzing data from CSV and RAW files to develop predictive models for leak type classification and severity assessment in a water distribution system. The process begins with the extraction of two datasets, the Leak Type Dataset and the Severity Dataset, which are separately processed. Each dataset is split into training, validation, and test sets, with the validation set used for monitoring model performance through a watchlist. Various ML and deep learning models, including LSTM, RNN, GRU, XGB, RF, and LightGBM, are trained on the respective train sets and used for prediction on the test sets. The model outputs are then evaluated through performance assessment metrics, which analyze their effectiveness in detecting leak types and assessing severity. The evaluated results are subsequently visualized to facilitate interpretation and derive insights. The figure effectively illustrates a systematic approach for training, validating, and testing ML models to enhance leak detection and severity assessment in WDS.

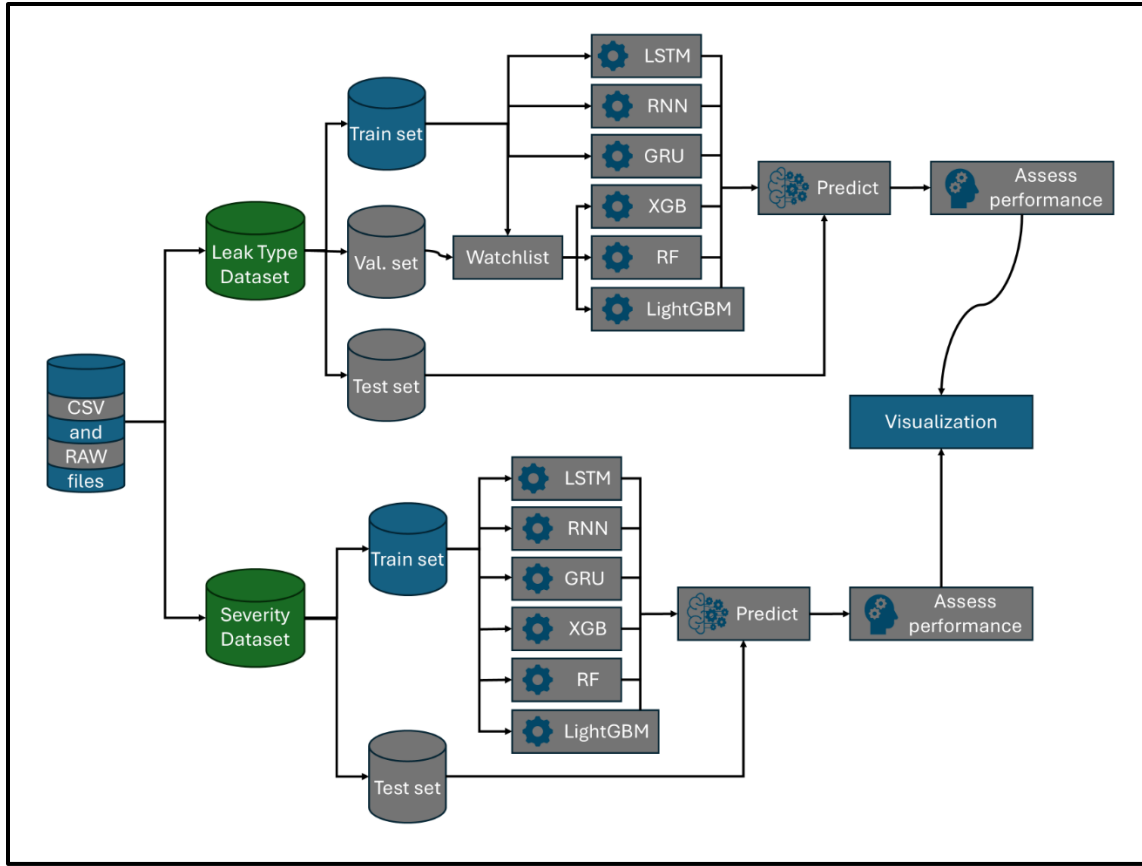


Figure 3.9: Structured workflow for processing and analyzing data from CSV and RAW files to develop predictive models for leak type classification and severity assessment in a water distribution system.

3.4.1 Leakage Type Classification

The leakage detection process is structured within a multi-sensor dataset framework, integrating data from Pressure Transducers (8 variables), Hydrophones (12 variables), and Accelerometers (8 variables), resulting in a comprehensive dataset of 28 variables. This integrated approach enhances classification accuracy across Branched and Looped water distribution networks (WDS) by leveraging the unique contributions of each sensor type: Pressure Transducers capture leak-induced pressure variations, Hydrophones detect acoustic signatures of leaks, and Accelerometers analyze structural vibrations caused by

water flow disturbances. The fusion of these data sources into a single framework enables a more data-rich and precise classification approach.

To ensure robust classification across different network topologies, two separate ML models are trained for each algorithm, one dedicated to Looped WDS and another for Branched WDS. The ML models employed in this framework include XGB, RF, LightGBM, LSTM, RNN, and GRU, each trained on structured data segmented across varying flow and transient conditions. Figure 3.10 illustrates the entire process of integrating sensor data into the multi-sensor framework, detailing how individual datasets are combined to form a unified structure. This visualization provides a clear end-to-end perspective on the development pipeline, ensuring a comprehensive evaluation of the combined sensor framework's effectiveness in leak classification.

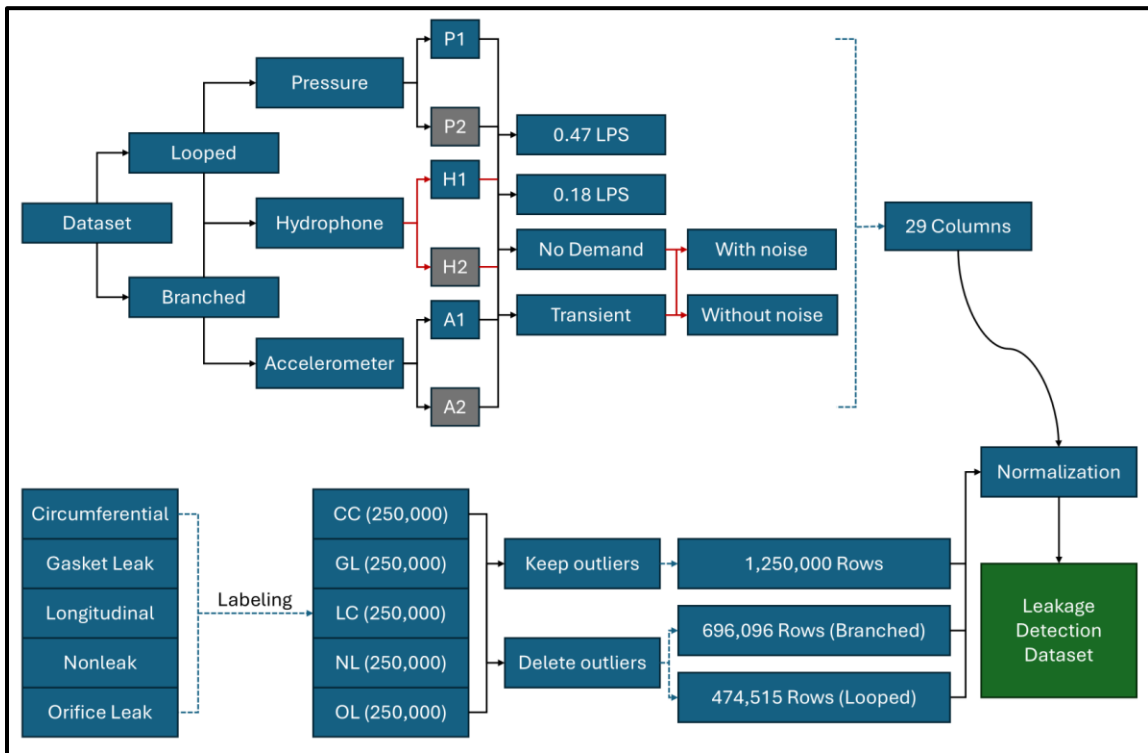


Figure 3.10: Systematic illustration of the leakage detection dataset structure for both cases (after and before removing the outliers) used in models development.

3.4.2 Leakage Severity Assessment

This study aims to assess leakage severity by categorizing leaks into three levels: severe, moderate, or minor based on sensor readings. The classification process involves labeling the dataset according to these categories, enabling ML models to detect meaningful patterns for severity assessment. The labeling is guided by an experimental observation that service demand is inversely proportional to leakage flow rate—as demand increases, leakage flow rate decreases. Consequently, scenarios with zero demand (no water usage) are labeled as severe, while 0.18 LPS demand and 0.47 LPS demand are classified as moderate and minor, respectively. The transient condition, where demand fluctuates between 0 and 0.47 LPS, presents a unique challenge. At certain points, the leakage severity may be severe, gradually decreasing to minor as demand increases. However, since these variations are unpredictable, labeling the transient demand scenario is deemed inapplicable. This systematic labeling approach ensures that ML models can effectively learn and differentiate leakage severity levels from sensor data.

Figure 3.11 illustrates the data processing pipeline for constructing the Severity Assessment Dataset from sensor data collected in a WDS. The dataset incorporates measurements from looped and branched network topologies, using three sensor types: pressure sensors (P1, P2), hydrophones (H1, H2), and accelerometers (A1, A2), which capture different leak types (CC, GL, LC, OL) as well as NL scenarios. The data is categorized by demand levels (0.47 LPS, 0.18 LPS, and no demand), with “No Demand” instances containing background noise in the case of hydrophone sensors. After structuring the data into seven columns, a normalization process is applied to prepare it for ML-based severity assessment. A key step in this process is labeling severity levels, where no demand

is categorized as severe, 0.18 LPS as moderate, and 0.47 LPS as minor, while also accounting for noise variations (neglecting No noise signals). The dataset is further divided into two sets: one keeping outliers (750,000 rows) and another removing outliers, with separate distributions for looped and branched networks. The final row counts for each leak type vary based on network topology, influencing data availability. Ultimately, the processed and normalized dataset is finalized as the Severity Assessment Dataset, which serves as a foundation for ML-based analysis of leak severity in WDS.

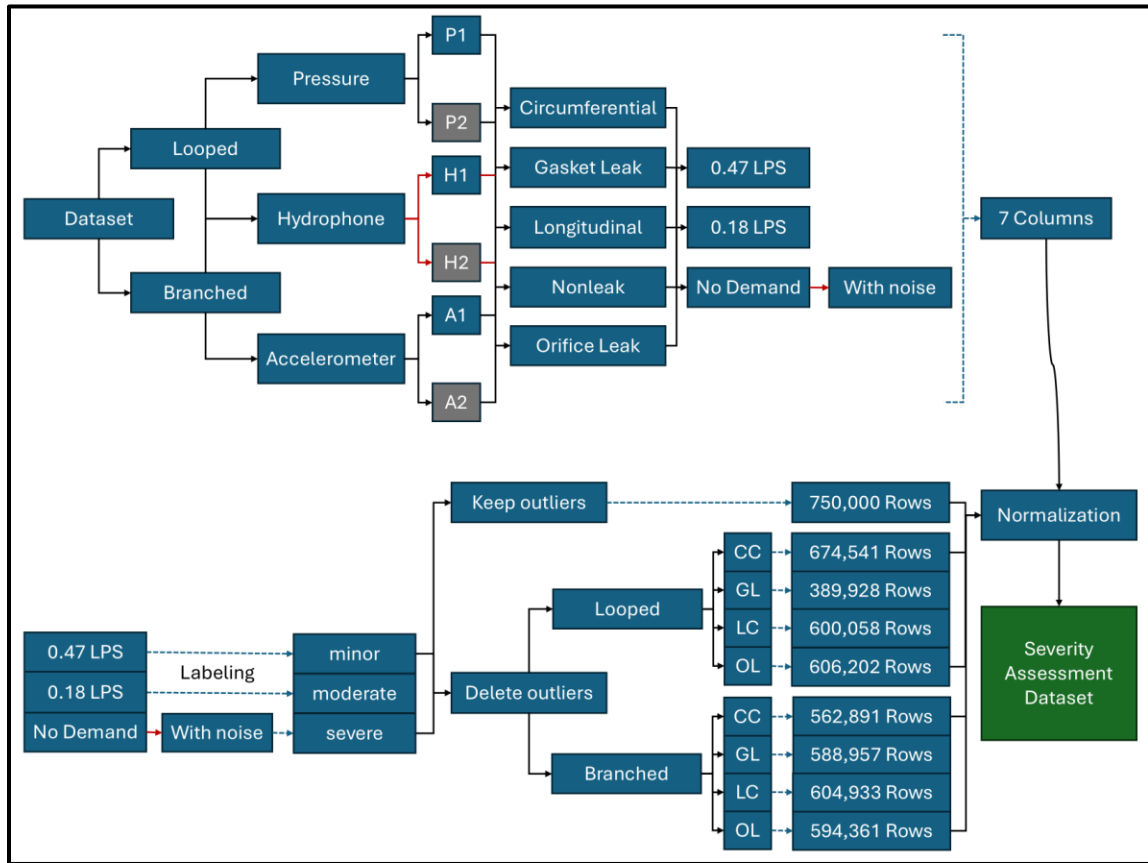


Figure 3.11: Visual representation of the labeling process, highlighting the systematic methodology and the relationships between sensor data and severity assessment.

The process is repeated independently for Branched and Looped network configurations to capture potential differences in sensor response across network topologies. Hydrophone

scenario with background noise is also tested, ensuring that environmental factors are accounted for. By combining and labeling methodologies, this comprehensive framework identifies the optimal ML model, combination, and setup for accurately assessing leakage severity in real-world water distribution systems.

3.4.3 Performance Analysis

Uncertainty analysis plays a critical role in enhancing the reliability and interpretability of ML models for classification tasks such as leakage type determination in water distribution systems. Unlike regression tasks, where uncertainty is commonly quantified through metrics like prediction intervals, correlation coefficient, and errors, classification tasks require specialized approaches to account for the categorical nature of outputs. A systematic methodology is implemented to address uncertainty, leveraging probabilistic outputs, entropy-based measures, and visualization techniques to provide actionable insights into model reliability.

The first step in uncertainty analysis involves the evaluation of ML models using appropriate classification metrics. Models such as XGB, LightGBM, RF, RNN, GRU and LSTM are trained on preprocessed datasets that include leakage type categories such as longitudinal crack, circumferential crack, gasket leak, orifice leak, and non-leak scenarios. These models are configured to output probabilistic predictions for each class, enabling a nuanced understanding of their decision-making processes. Standard performance metrics, including accuracy, precision, recall, F1-score. Table 3.4 presents the formulations used to compute overall classifier performance metrics, which serve as a comparative measure of ML models. These overall metrics provide insight into the general effectiveness of a model relative to alternative approaches. In contrast, class-specific metrics, such as F1-score,

recall, precision, and balanced accuracy, evaluate the model's ability to predict each class independently within the target variable. Analyzing these metrics helps identify whether a classifier struggles with detecting a particular class or exhibits performance deficiencies across all classes. This distinction offers valuable insights into potential areas for model refinement and optimization, guiding targeted improvements to enhance classification effectiveness.

Table 3.4: Formulations used to calculate overall performance metrics in classification ML models evaluation process based on confusion matrices.

Metric	Formula
Accuracy	$\frac{\sum TP_i}{N}$
Kappa (κ) Statistic	$\frac{Accuracy - P_e}{1 - P_e}$, where $P_e = \frac{\sum(\text{row total}_i \times \text{column total}_i)}{N^2}$
Confidence Interval (Lower Bound)	$Accuracy - Z \times \sqrt{\frac{Accuracy \times (1 - Accuracy)}{N}}$
Confidence Interval (Upper Bound)	$Accuracy + Z \times \sqrt{\frac{Accuracy \times (1 - Accuracy)}{N}}$
No Information Rate (NIR)	$\frac{\max(\text{row totals})}{N}$

The evaluation of multi-class classification models relies on fundamental metrics derived from the test-set confusion matrix, providing a robust assessment of model performance. Overall accuracy measures the proportion of correctly classified instances across all classes, offering a direct but sometimes misleading indicator of performance when class imbalances exist. It is computed as the sum of correctly classified samples for each class, divided by the total number of instances in the dataset.

To address potential biases in accuracy, the Kappa (κ) statistic is employed as a measure of agreement between predicted and actual classifications beyond random chance. The kappa statistic accounts for class imbalances by incorporating the expected classification agreement based on marginal distributions of the confusion matrix. A higher κ value indicates stronger classification reliability, while values closer to zero suggest that the model is performing only slightly better than random guessing.

Statistical confidence in classification accuracy is assessed using confidence intervals, which define a range within which the true accuracy of the model is likely to fall. The lower and upper confidence bounds are computed based on the standard error of accuracy, providing a means to evaluate the stability of model performance. These intervals are particularly useful when comparing multiple models or assessing performance across different datasets.

The No Information Rate (NIR) serves as a baseline to determine whether the classifier significantly outperforms a naive model that predicts the most frequent class in the dataset. It is calculated as the proportion of instances belonging to the most common actual class. If a model's accuracy does not substantially exceed the NIR, it suggests that the classifier may lack discriminative power. Each metric is computed per class based on the confusion matrix. The terms used in the formulas shown in Table 3.6 are defined as follows in Table 3.5.

Table 3.5: Abbreviations, and brief description of the terms used in the formulas of calculating By-class performance metrics.

Term	Symbol	Description
True Positives	TP	The number of correctly classified instances of the class.
True Negatives	TN	The number of instances correctly classified as not belonging to the class.
False Positives	FP	The number of instances incorrectly classified as the class.
False Negatives	FN	The number of instances that belong to the class but were misclassified.
Positive Cases	P	The actual instances of the class.
Negative Cases	N	The actual instances not belonging to the class.

Certain classification metrics are commonly referred to by multiple names, reflecting their role in different analytical contexts. For instance, recall is interchangeably known as sensitivity or true positive rate (TPR), as it quantifies the model's ability to correctly identify positive instances. Similarly, precision is also referred to as the positive predictive value (PPV), as it measures the proportion of correctly predicted positive cases among all instances classified as positive. Additionally, specificity is equivalent to the true negative rate (TNR), indicating the model's capacity to accurately recognize negative instances. These terminologies emphasize the overlapping yet distinct perspectives from which classification performance can be evaluated.

Table 3.6: Formulations used to calculate overall performance metrics in classification ML models evaluation process based on confusion matrices.

Metric	Formula (Per Class Calculation)	Description
Specificity (TNR)	$\frac{TN}{TN + FP}$	Measures the ability to correctly classify negative instances of the class.
Negative Predictive Value (NPV)	$\frac{TN}{TN + FN}$	Indicates how often a negative prediction is correct.
Precision (PPV) [157]	$\frac{TP}{TP + FP}$	Measures how many predicted instances of the class are actually correct.
Recall (TPR, Sensitivity) [157]	$\frac{TP}{TP + FN}$	Measures how many actual instances of the class are correctly identified.
F1-Score [157]	$\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$	Harmonic mean of precision and recall, balancing false positives and false negatives.
Balanced Accuracy	$\frac{\text{Sensitivity} + \text{Specificity}}{2}$	Adjusts for class imbalance by averaging sensitivity and specificity.

By integrating these evaluation metrics, a comprehensive understanding of model performance is achieved, ensuring that classification effectiveness is analyzed beyond simple accuracy measurements. These metrics provide deeper insights into both model reliability and uncertainty, allowing for more informed decision-making in multi-class classification tasks.

Visualization plays an integral role in uncertainty analysis, enabling stakeholders to interpret model performance effectively. Confidence histograms are generated to depict the distribution of prediction confidence across all classes, highlighting areas where the model exhibits consistent accuracy or variability. Additionally, heatmaps of confusion

matrices are employed for Severity Assessment models, providing a detailed representation of misclassification patterns and model reliability across different severity levels. Bar charts are utilized to illustrate key performance metrics, offering a comparative analysis of model effectiveness across different configurations. Furthermore, the impact of outlier removal is systematically analyzed, assessing its influence on classification performance and robustness. These visual tools are instrumental in identifying systematic errors, guiding further data collection, and informing model refinement strategies.

By systematically addressing uncertainty, this methodology enhances the practical applicability of ML models for leakage detection. The combination of tabulated outputs and visualizations ensures that predictions are interpretable and reliable, supporting informed decision-making. These advancements provide a robust framework for improving the efficiency of water distribution systems, reducing non-revenue water losses, and prioritizing maintenance efforts. This comprehensive uncertainty analysis underscores the importance of reliable predictive systems in addressing real-world challenges in infrastructure management.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Data Preprocessing

WDS are integral to urban infrastructure, ensuring the continuous supply of potable water. Leakage in these systems results in substantial water loss, economic inefficiencies, and potential structural damage. Sensor-based monitoring of WDS enables the detection of anomalies indicative of leaks. However, datasets derived from these sensor measurements often contain noise and outliers due to various factors such as sensor malfunctions, environmental disturbances, and data transmission errors. Effective preprocessing, particularly outlier detection and removal, is essential for improving the accuracy of leak detection algorithms. The Q1 method, also known as the first quartile method, is a robust statistical approach for identifying and eliminating outliers from the dataset.

Outliers are data points that significantly deviate from the distribution of the majority of the dataset. They can be caused by faulty sensor readings, abrupt environmental changes, or errors in data logging. The Quartile-Based Method for Outlier Detection will be used as a part of the study to eliminate the presence of outliers in the datasets. Quartile method, a component of the interquartile range (IQR) technique, is widely used in statistical data preprocessing. It is based on the spread of data within the first quartile (25th percentile, Q1) and third quartile (75th percentile, Q3) of a dataset. The interquartile range (IQR) is calculated in Eq. (4.1).

$$IQR = Q_3 - Q_1 \quad (4.1)$$

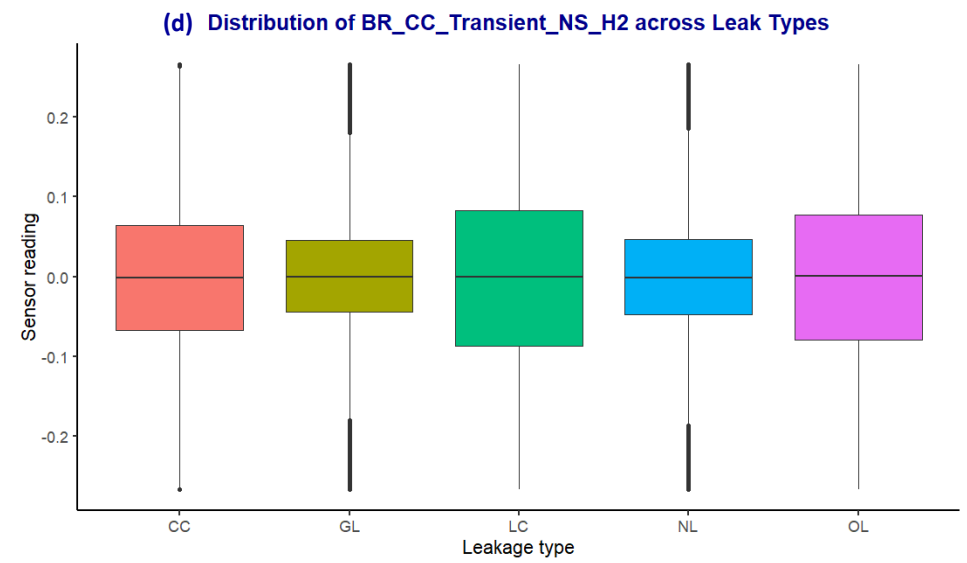
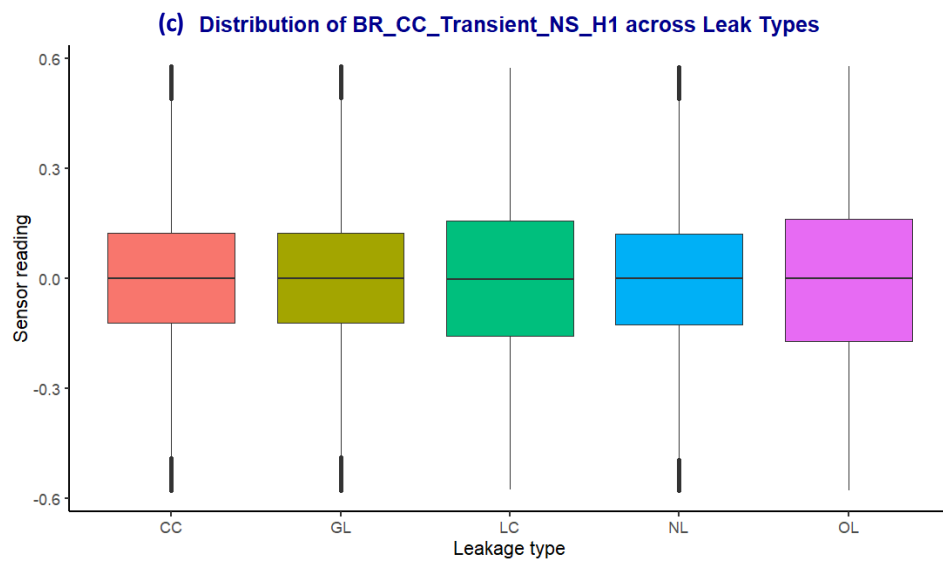
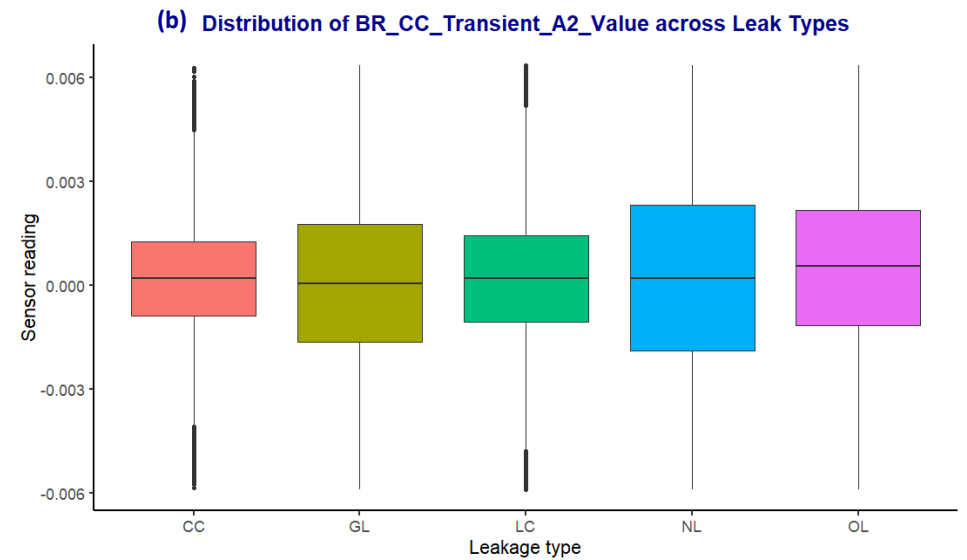
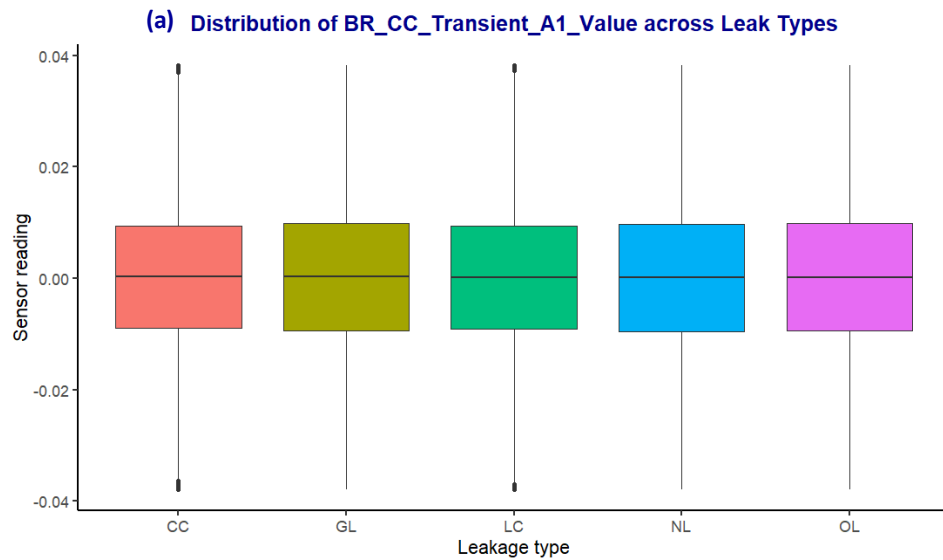
The lower bound and upper bound for detecting outliers are known as the point located 1.5 times IQR below Q_1 and 1.5 times IQR above Q_3 respectively in a box plot. These boundaries can be defined as in Eqs. (4.2 and 4.3):

$$\text{Lower boundary} = Q_1 - 1.5 * IQR \quad (4.2)$$

$$\text{Upper boundary} = Q_3 + 1.5 * IQR \quad (4.3)$$

This study provides an in-depth analysis of the impact of outlier removal using the IQR method on sensor readings employed in classification-based ML predictions, such as leakage detection. The study involved training ML models with and without the removal of outliers. The effect of outlier removal was then assessed in terms of the performance of these ML models, focusing on their overall predictive accuracy as well as their efficiency in classifying different leak conditions. This evaluation enables a comprehensive understanding of how data preprocessing influences ML-based leak detection systems.

Figure 4.1 presents boxplots of sensor readings (A, H, P) categorized by leak type across branched network at transient demand levels. The presence of outliers in Figures 4.1 (a - h) highlight the significant impact of outlier removal on detecting certain leak types. This discrepancy can lead to data scarcity for specific leak categories, potentially affecting the classifier's ability to distinguish between different leak conditions. The removal of extreme values may enhance model accuracy in some cases but could also introduce class imbalance, necessitating careful handling to ensure a robust leak detection framework.



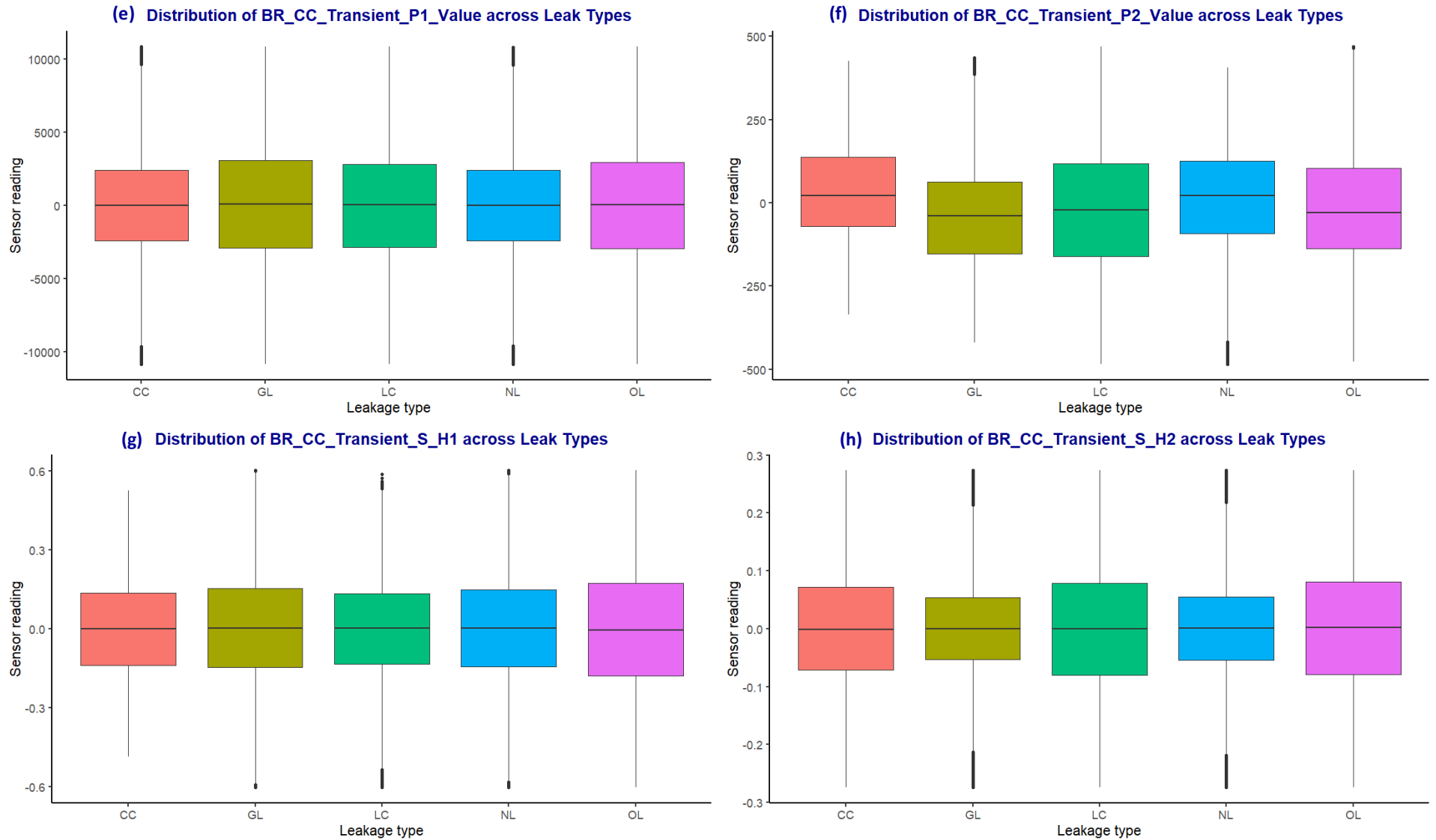


Figure 4.1 Box plot visualization of the outliers in transient flow demand data across different leak types for Branched network. (a) Accelerometer 1 (b) Accelerometer 2 (c) No sound Hydrophone 1 (d) No sound Hydrophone 2 (e) Pressure transducer 1 (f) Pressure transducer 2 (g) Sound present Hydrophone 1 (h) Sound present Hydrophone 2.

The preprocessing stage also involves data scaling, which is essential due to the varying ranges of sensor readings. For instance, pressure transducer P1 records values between -250 and 250, while P2 exhibits a significantly broader range from -500,000 to 500,000. These discrepancies necessitate scaling to ensure consistency across features. Additionally, hydrophone readings fall within a much smaller range of -0.3 to 0.3, further highlighting the importance of normalization to prevent dominance of features with larger magnitudes in subsequent analyses. Scaling is crucial for ML neural network models such as LSTM, GRU, and RNN, as these models are sensitive to feature magnitudes. However, it is less significant for tree-based models like XGB, RF, and LightGBM, which do not rely on feature scaling for performance. For deep learning models, normalization was applied using the base function 'scale' in R. Although scaling is not essential for tree-based models, it was still performed to ensure consistency and comparability across models during validation.

Following the preprocessing steps of scaling and outlier removal, the next critical stage in preparing the dataset for ML models is labeling. Labeling in the context of leakage detection involves adding a designated column to the dataset that classifies each data entry based on the type of leak. This classification can be represented using categorical abbreviations (CC, LC, GL, OL, NL) or numerical values, depending on the requirements of the ML algorithm being employed. For algorithms that require numerical input, these categories can be encoded using numeric values. The numerical encoding can start either from 0 to 4 or from 1 to 5, depending on the specifications of the predefined ML function. The choice of encoding is dictated by the format expected by the ML framework in use.

For instance, in R, the XGB algorithm is capable of handling multi-class classification where the target variable can remain in its categorical form (e.g., abbreviations such as CC, LC, etc.). In contrast, deep learning models like LSTM in Python require the target variable to be converted into numeric labels. In such cases, categorical values must be transformed into numerical factors, typically starting from 0. Ensuring proper labeling is essential for the effectiveness of classification models. Mislabeling or inconsistencies in encoding may lead to errors in model training and prediction. Thus, careful attention to the formatting and compatibility of labels with the chosen ML approach is crucial for achieving accurate and reliable results in leakage detection.

The leakage detection dataset for each network type (Branched and Looped) consists of 29 variables, including the target variable 'Leak_Type'. It comprises 8 variables from accelerometer sensors, capturing vibration-related data, 8 variables from pressure transducers, measuring pressure fluctuations where both are recorded at different demand levels, and 12 variables from hydrophones, collected under varying demand levels and noise scenarios to analyze acoustic signals. This structured dataset ensures a comprehensive representation of sensor measurements, facilitating the development of robust ML models for leak detection across different network configurations.

Labeling for Severity Assessment dataset comes according to the amount of water volume lost due to leaks per unit time, in the laboratory experiment, the leak flowrate is measured as the difference between the supplied flowrate and the service flowrate (demand). The difference represents the non-revenue water of the WDS. In the Laboratory, there are no Commercial or Authorized losses which means in this case the difference value

is entirely of physical losses and more specifically WDS physical leaks. Table 4.1. shows the calculations of the flowrates benchmarked with the dataset [76].

Table 4.1: Measurements of Leak flowrate based on demand and supply flowrates for both looped and branched WDS.

Network	Leak Type	Design Demand (L/s)	Supply Flow (L/s)	Service Flow (L/s)	Measured Leak Flow (L/s)	Leak Rate (%)
Looped	OL	0.00	0.047	0	0.049	-
		0.18	0.237	0.193	0.044	19
		0.47	0.512	0.472	0.04	8
	LC	0.00	0.032	0	0.032	-
		0.18	0.222	0.196	0.026	12
		0.47	0.497	0.473	0.024	5
	CC	0.00	0.026	0	0.026	-
		0.18	0.214	0.189	0.025	12
		0.47	0.498	0.475	0.023	5
	GL	0.00	0.053	0	0.053	-
		0.18	0.24	0.191	0.049	20
		0.47	0.524	0.476	0.048	9
Branched	OL	0.00	0.042	0	0.042	-
		0.18	0.239	0.199	0.04	17
		0.47	0.51	0.475	0.035	7
	LC	0.00	0.03	0	0.03	-
		0.18	0.222	0.198	0.024	11
		0.47	0.495	0.475	0.02	4
	CC	0.00	0.024	0	0.024	-
		0.18	0.216	0.193	0.023	11
		0.47	0.494	0.476	0.018	4
	GL	0.00	0.075	0	0.075	-
		0.18	0.265	0.194	0.071	27
		0.47	0.541	0.475	0.066	12

The severity of leakage was observed to increase as demand decreases, as illustrated in Figure 4.2. The figure visualizes the leakage flow rates for each leak type in both branched and looped networks, highlighting a consistent trend where leakage flow rates decrease as demand levels increase across all cases and network types. This phenomenon can be attributed to several factors. One possible explanation is that at lower demand levels,

the internal pressure within the WDS increases, leading to a higher leakage flow rate due to the greater force exerted on pipeline weaknesses. Additionally, lower demand results in reduced overall flow velocity, allowing more time for pressurized water to escape through leaks, thereby increasing leakage severity. Conversely, at higher demand levels, the overall system pressure is lower, reducing the driving force behind leak flow rates and subsequently decreasing the severity of leakage.

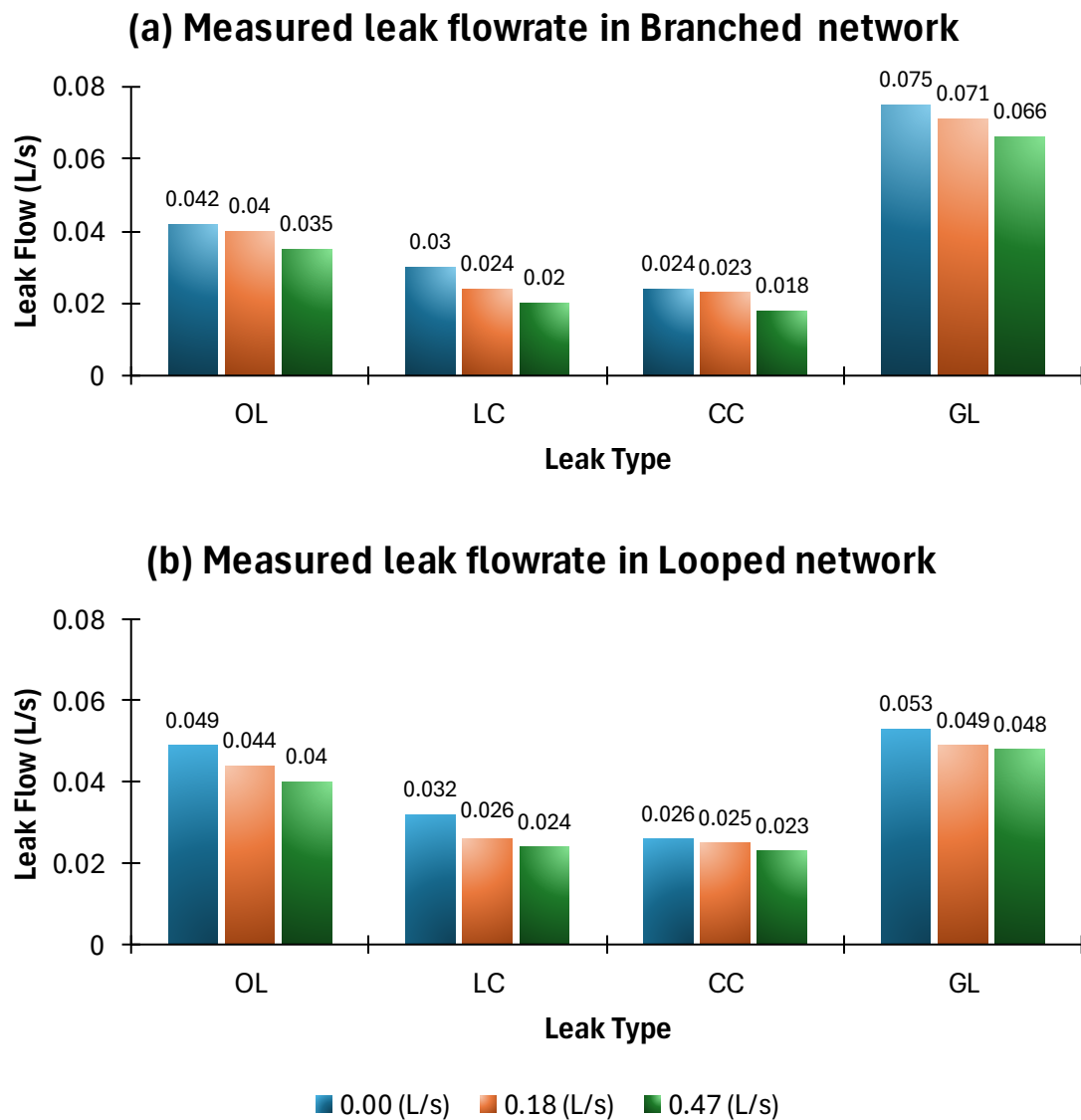


Figure 4.2: Severity of leakage observed to increase as demand decreases, visualization of leakage flow rates for each leak type in both systems (a) branched (b) looped.

Therefore, sensor signals corresponding to a demand level of 0.47 LPS are categorized as minor leakage severity, as the higher water demand results in lower internal pressure, reducing the driving force for leakage. Signals recorded at 0.18 LPS are labeled as moderate leakage severity, representing an intermediate state where the system pressure is higher than at 0.47 LPS but lower than at 0.00 LPS, leading to increased leakage severity. Finally, sensor signals associated with no demand (0.00 LPS) are classified as severe leakage, as the absence of outflow causes a buildup of pressure within the system, maximizing the force exerted on existing pipeline weaknesses and consequently increasing leakage rates. This classification is essential for analyzing how demand variations influence leak severity, allowing for a structured assessment of leakage behavior under different operating conditions.

To facilitate a focused evaluation of leak severity, the dataset is partitioned into eight separate subsets. Specifically, the dataset is divided into four distinct datasets for the Looped WDS, each corresponding to a different leak type: CC, LC, GL, and OL, with the Non-Leak (NL) case being excluded from this classification. Similarly, four datasets are created for the Branched WDS, mirroring the classification structure used for the Looped WDS. This division ensures that the datasets maintain consistency across network configurations while allowing for an independent evaluation of leak severity for each specific leak type. By structuring the data in this way, ML models can be effectively trained to recognize the severity of leaks under different demand conditions, ultimately aiding in the development of predictive and diagnostic tools for water distribution system monitoring and management.

The final step in the preprocessing pipeline for both Severity Assessment and Leakage Detection datasets involves applying a 0.7 ratio stratified splitting. Stratified splitting is a data partitioning technique that ensures the distribution of classes in the target variable remains proportional in both the training and test sets. This method is particularly important in classification problems, as it prevents any class imbalance that might arise from random splitting. By maintaining the proportion of each class, the dataset ensures that the training set includes adequate representations of all leakage severity levels or leak types, which is crucial for training robust ML models.

Stratified splitting helps mitigate class imbalance issues that could otherwise lead to biased model predictions. If the dataset were split randomly without stratification, some leakage severity levels or leak types might end up underrepresented in the training set, leading the model to favor more frequently occurring classes and struggle to recognize underrepresented ones. By ensuring that each class has a sufficient number of training samples, this approach enhances the generalizability of the model and prevents it from disproportionately favoring certain classes over others during prediction. This preprocessing step is fundamental in maintaining model fairness and reliability, particularly in real-world applications where accurate classification across all leak types and severities is essential for effective water distribution system monitoring.

For the Leak Detection task, 50% of the test set is allocated as a validation set, specifically for tree-based algorithms, using a stratified splitting technique. This step is necessary because models such as XGB and LightGBM require a validation set as a watchlist to monitor and optimize performance during training. The validation set helps fine-tune hyperparameters and assess model generalization without overfitting. And to

ensure that the model does not unintentionally memorize values from the test set, a separate validation set is generated, reinforcing the integrity of model evaluation. In the case of the Looped WDS dataset for leak detection (before outlier removal), it consists of 1,250,000 observations (where each signal contains 250,000 observations across five leak types). The dataset is then split into 875,000 observations for training, 185,000 observations for validation, and 185,000 observations for testing, maintaining a balanced representation across leak types to enhance model robustness and reliability.

The issue of imbalanced classes, which was initially addressed using the stratified splitting technique, can also arise during the process of outlier elimination. Figure 4.3 illustrates the difference in the number of observations before and after outlier removal in the Looped network dataset. It is evident that the NL (No Leak) and OL (Orifice Leak) classes experience a significant reduction in observations compared to the other three classes, leading to data scarcity in specific categories. This reduction poses a challenge, as insufficient training data for certain classes can negatively impact the performance of ML models, making them less effective at predicting underrepresented leak types. The impact of class imbalance on model accuracy and generalization is further explored in this study.

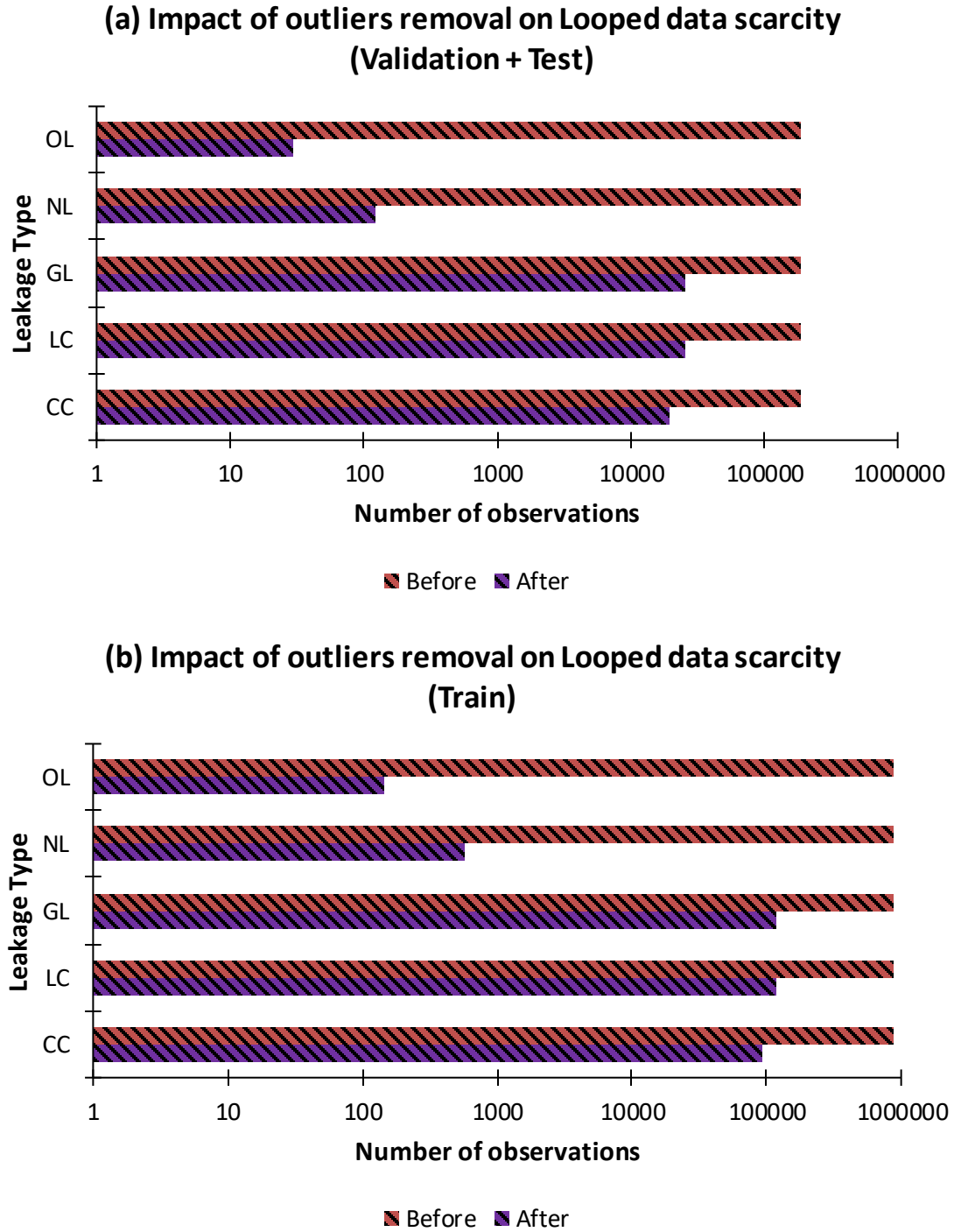


Figure 4.3: Impact of outlier removal on data scarcity in the looped WDS. The bar charts illustrate the number of observations before and after outlier removal for different leakage types (a) Effect on the validation and test datasets (b) Effect on the training dataset.

4.2 Branched Topology

In practical applications of leak detection, the standard approach involves directly inputting raw sensor readings into a ML trained model, with prior knowledge that the WDS follows a Branched network topology. The objective is to classify whether these readings correspond to a leakage scenario or a NL condition and, in the presence of a leak, determine its specific type among the four possible categories: OL, GL, CC, or LC. This classification is a crucial step in leak detection, as it provides a foundational understanding of the issue within the WDS. However, leak detection alone is insufficient for effective water loss mitigation; it is also essential to quantify the severity of the leakage to assess the magnitude of water loss. To achieve this, the ML model is further utilized to estimate the severity of the detected leakage, under the assumption that the WDS operates under a steady flow condition with a constant demand level across all sensor readings. This assumption simplifies the estimation process and allows for a more controlled assessment of leak severity.

Despite this assumption, real-world WDS networks do not operate under strictly steady-state conditions, as water demand levels continuously fluctuate due to variations in usage patterns. Such variations introduce complexities in analyzing sensor signals, as the same leakage may exhibit different signal characteristics under different demand conditions. Consequently, a direct assessment of sensor signals without accounting for these variations would lead to unreliable classifications and severity estimations. To overcome this challenge, transient signals present in the dataset are leveraged to simulate dynamic demand scenarios. Transient signals refer to sensor readings captured during periods of demand fluctuation, representing system behavior when flow conditions change

over time. These signals inherently contain rich information about the response of the WDS to varying conditions, making them valuable for simulating realistic operational environments. By carefully analyzing transient signals, ML models can be trained to recognize leakage patterns under fluctuating demand, improving the adaptability and accuracy of leak detection and severity estimation.

To make transient signals more usable in a ML context, they are decomposed into sub-signals, each representing an approximate steady demand level. Specifically, each transient signal is divided into multiple sub-signals, with each sub-signal consisting of 1,000 observations captured within a 30-second interval. This segmentation allows for an approximation of steady demand levels within a short time window, enabling ML models to assess leakage characteristics in a manner comparable to that of a true steady-state condition. By employing this approach, it becomes possible to extract meaningful insights from transient conditions while preserving the stability required for accurate ML-based classification. The subdivision of transient signals into structured sub-signals ensures that the dataset can simulate a wide range of real-world operational conditions, making the trained models more robust and generalizable across different demand variations. This methodology enhances the practicality of ML-based leak detection systems by bridging the gap between idealized steady-state assumptions and the inherently dynamic nature of WDS networks, ultimately leading to more reliable and effective water loss management strategies.

4.2.1 Without outliers

The performance metrics presented in this section evaluate the ability of Deep learning and Tree-based ML models to classify leakage severity across four different leak types (CC,

LC, GL, and OL) in a Branched WDS. All models were trained and tested using datasets that had been preprocessed through the IQR outlier removal method, ensuring a cleaner and more representative dataset. This preprocessing step is critical, as it removes extreme values that could skew model predictions and hinder performance, particularly in identifying subtle or severe leaks.

4.2.1.1 Tree-based models

Table 4.2 presents a comparative evaluation of three Tree-based models (XGB, RF, and LightGBM) across four different leak types in a Branched WDS. The performance metrics include Accuracy, Kappa, and Confidence Intervals (CI Lower and CI Upper), providing a robust assessment of model effectiveness trained on a dataset subject to IQR outliers removal. Across all leak types, LightGBM consistently achieves the highest Accuracy and Kappa scores, particularly for BR_CC and BR_LC, suggesting superior predictive performance compared to RF and XGB. RF follows closely, showing strong generalization capabilities with slightly lower Accuracy and Kappa values than LightGBM but consistently outperforming XGB. XGB, while performing well in BR_CC and BR_LC, shows the lowest performance for BR_GL and BR_OL, indicating that it may struggle with these specific leak types. The Confidence Intervals in Table 4.2 further illustrate the reliability of these models, with tighter intervals for LightGBM and RF, indicating lower variance and higher stability in predictions.

Figure 4.4 visually reinforces these trends by illustrating the distribution of Accuracy, Kappa, and Confidence Intervals across models and leak types. The bar heights for LightGBM remain consistently higher, particularly in BR_CC and BR_LC, supporting its superior classification capabilities. RF maintains competitive performance, as seen in

BR_GL and BR_CC, while XGB exhibits more fluctuations, particularly in BR_GL and BR_OL, where its Accuracy and Kappa values drop. The clear distinction in CI ranges, with LightGBM and RF exhibiting tighter confidence bounds, confirms their robustness in handling varying leak types. Overall, it is demonstrated that LightGBM is the most effective Tree-based model for Severity Assessment in a Branched WDS, followed closely by RF, with XGB showing more performance variability across leak types.

Table 4.2: Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over branched network after removing the outliers.

model	Metric	BR_CC	BR_LC	BR_GL	BR_OL
XGB	Accuracy	0.9010722	0.8774559	0.8117497	0.7751952
	Kappa	0.8403608	0.8095885	0.7015737	0.6424472
	CI Lower	0.8993133	0.875595	0.8095054	0.7728112
	CI Upper	0.9028103	0.8792987	0.8139787	0.7775657
	No Info Rate	0.4278227	0.3944952	0.3962306	0.40674
RF	Accuracy	0.9083649	0.8773402	0.8516003	0.8206979
	Kappa	0.8520877	0.8091487	0.7649749	0.7147129
	CI Lower	0.9066646	0.8754785	0.8495573	0.8185052
	CI Upper	0.9100442	0.8791837	0.853626	0.8228749
	No Info Rate	0.4278227	0.3944952	0.3962306	0.40674
LightGBM	Accuracy	0.9312471	0.9091623	0.8581968	0.830818
	Kappa	0.8892368	0.8590528	0.7759076	0.7317566
	CI Lower	0.9297534	0.907529	0.8561915	0.8286745
	CI Upper	0.9327185	0.9107759	0.8601844	0.8329454
	No Info Rate	0.4278227	0.3944952	0.3962306	0.40674

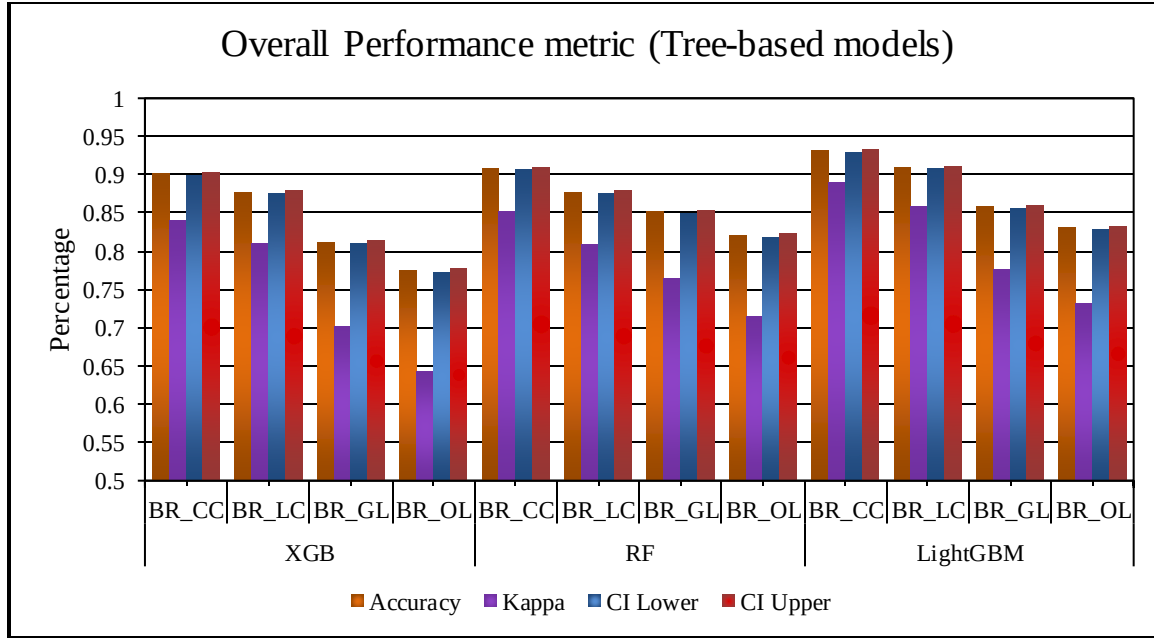


Figure 4.4: Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over branched network after removing the outliers.

The evaluation of all proposed ML Severity Assessment models' performance focuses on key classification metrics, including Positive Predictive Value (Pos Pred Value), Negative Predictive Value (Neg Pred Value), Precision, Recall, F1 Score, and Balanced Accuracy, across three severity classes: minor, moderate, and severe. By leveraging stratified splitting and an optimized ML framework, the models aim to provide a robust assessment of leakage severity, ensuring balanced and accurate predictions across all classes and leak types. The following discussion delves into the results, emphasizing the models' strengths and limitations in addressing the challenges of leakage severity assessment.

Table 4.3 presents a detailed performance assessment of RF models trained for four different leak types in a Branched WDS, highlighting the effectiveness of these models in

evaluating leakage severity. For each leak type (CC, LC, GL, and OL), the results demonstrate that the RF models achieve strong metrics, particularly for severe leakage detection. For instance, in the CC leak type, the model achieves an F1 Score of 0.9317 and a Recall of 0.9757 for the severe class, showcasing its capability in identifying critical leaks. However, the performance shows variability across other severity classes and leak types. For example, in the GL leak type, the F1 Score for minor leaks drops to 0.6380, indicating that the model struggles to reliably classify less pronounced leaks. These variations underscore the need for further optimization to handle leak types and severity levels with distinct sensor signal characteristics.

Figure 4.5 presents the confusion matrices for the four RF models (BR_CC, BR_GL, BR_LC, and BR_OL), offering a granular view of their classification patterns. A key observation across all matrices is the frequent misclassification of severe leaks as minor or moderate. For example, in the BR_CC model, 46,994 severe cases are misclassified as minor, while in BR_GL, 45,110 severe cases are similarly misclassified. This trend suggests overlapping sensor signal characteristics between severity levels, particularly for severe leaks, which complicates their classification. Conversely, the moderate class consistently achieves high true positive rates, as evidenced by the large number of correctly classified moderate cases in all models. This indicates the model's ability to capture features associated with intermediate severity levels. These findings reveal that while RF models perform reliably for moderate leaks, there is a need for enhanced feature extraction and additional adjustments to minimize the misclassification of severe leaks and improve overall classification accuracy.

Table 4.3: By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

RF Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9864	0.9657	0.9240	0.8246	0.8715	0.9055
	moderate	0.9503	0.9156	0.9229	0.8717	0.8965	0.9110
	severe	0.9112	0.9805	0.8915	0.9757	0.9317	0.9435
BR_LC	minor	0.9709	0.9166	0.8774	0.7026	0.7803	0.8367
	moderate	0.9215	0.9278	0.8715	0.8812	0.8763	0.9013
	severe	0.9154	0.9826	0.8824	0.9751	0.9265	0.9452
BR_GL	minor	0.9816	0.8962	0.8954	0.5803	0.7042	0.7810
	moderate	0.8726	0.9209	0.8163	0.8831	0.8483	0.8778
	severe	0.9068	0.9764	0.8718	0.9665	0.9167	0.9366
BR_OL	minor	0.9776	0.8791	0.8605	0.5069	0.6380	0.7423
	moderate	0.8626	0.8855	0.7840	0.8172	0.8003	0.8399
	severe	0.8713	0.9916	0.8405	0.9892	0.9088	0.9302

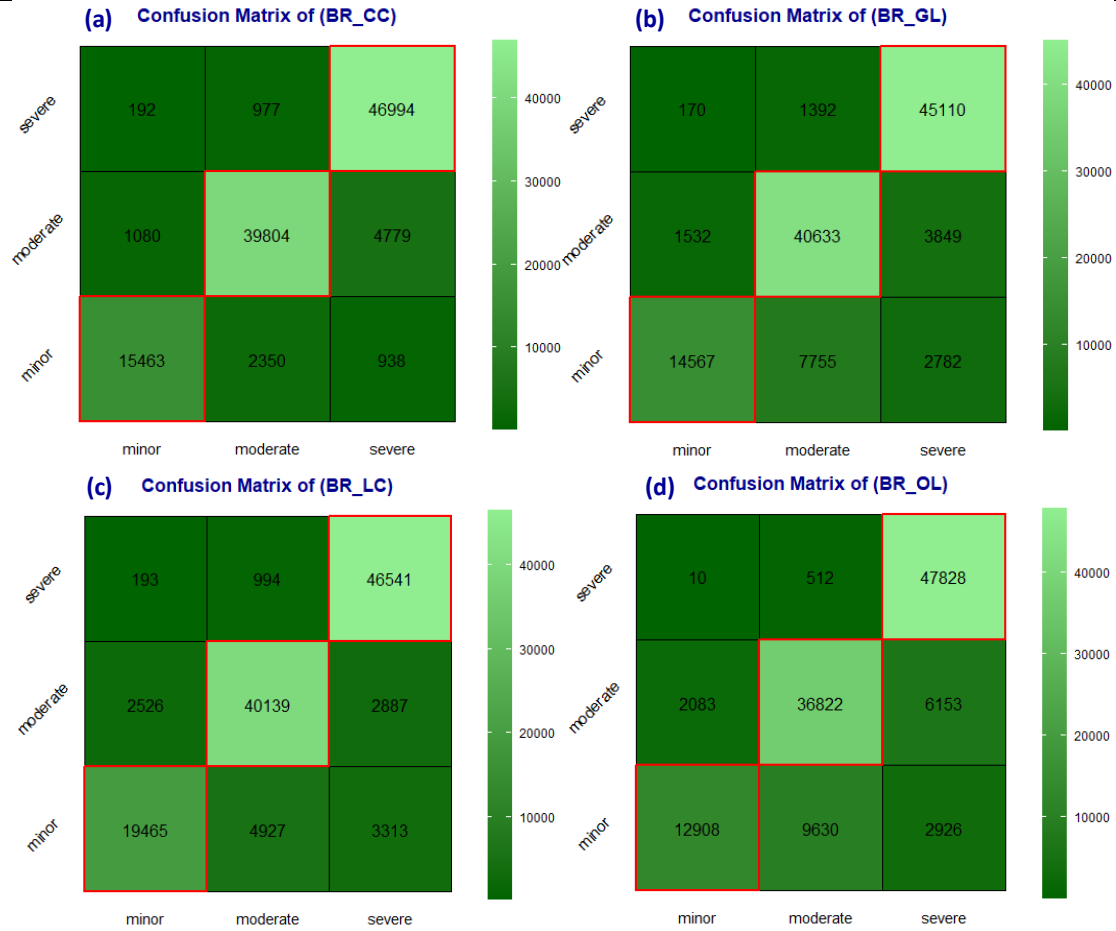
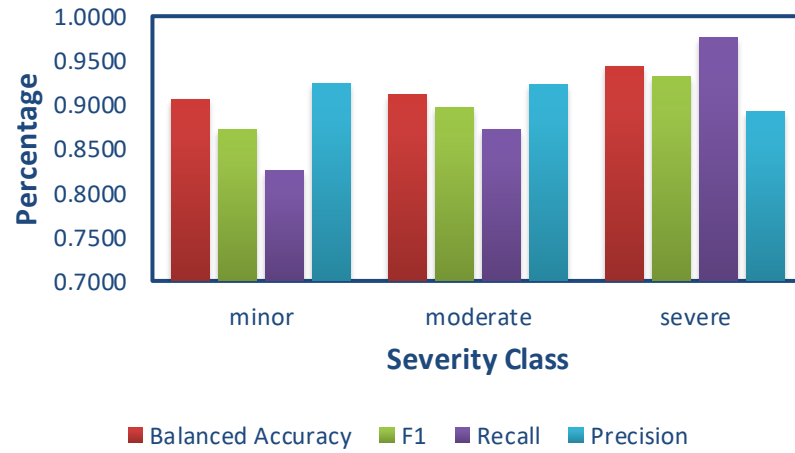


Figure 4.5: Confusion matrices for four Branched network RF models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.

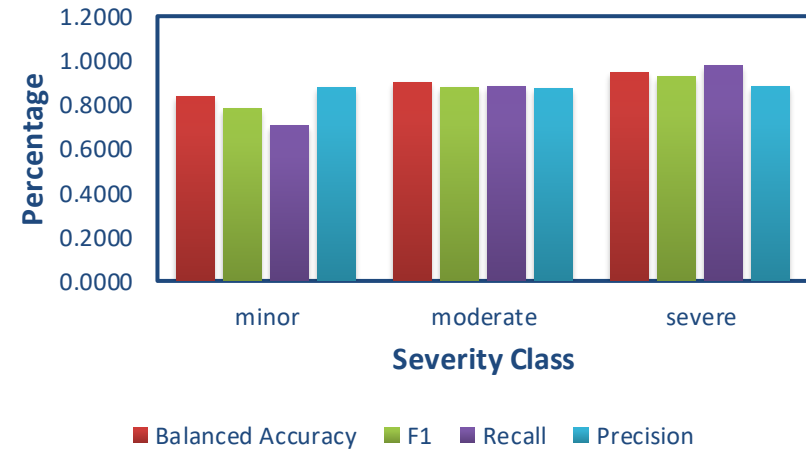
The visualization graphs (Figure 4.6) further illustrate these results, with each graph corresponding to a specific leak type. For example, Figure 4.6a, representing CC, shows balanced performance across all severity classes, with Balanced Accuracy consistently above 0.9, reflecting strong overall performance. In contrast, Figure 4.6c, which corresponds to GL, highlights a more noticeable disparity between the minor and severe classes, with minor class metrics significantly lower. This trend indicates that the RF model is more adept at detecting severe leaks but faces challenges in reliably identifying minor leaks under certain conditions. Additionally, while the moderate severity class demonstrates consistent performance across all leak types, variability in other classes highlights opportunities for improvement. Overall, the combined insights from Table 4.3 and Figure 4.6 underscore the RF model's capabilities in leakage severity assessment while identifying areas requiring additional refinement.

In the leak detection task for a Branched WDS, the performance metrics for the RF model, as presented in Table 4.4 and visualized in Figure 4.7, demonstrate its effectiveness. Table 4.4 provides detailed metrics for Precision, Recall, F1 Score, and Balanced Accuracy across the training, testing, and validation datasets. The performance is exceptionally high across all datasets and leak types (CC, GL, LC, NL, OL), with values consistently approaching 1.000 for each metric. This indicates that the RF model performs with near-perfect accuracy and reliability in identifying the presence or absence of leaks and classifying leak types within the Branched WDS. Notably, the Balanced Accuracy remains high across all datasets and classes, underscoring the model's robustness and ability to handle class imbalances effectively.

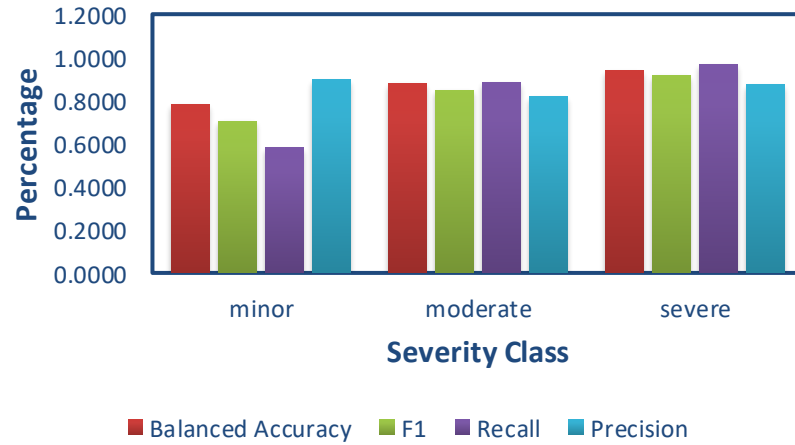
(a) Random Forest - Branched network - CC



(b) Random Forest - Branched network - LC



(c) Random Forest - Branched network - GL



(d) Random Forest - Branched network - OL

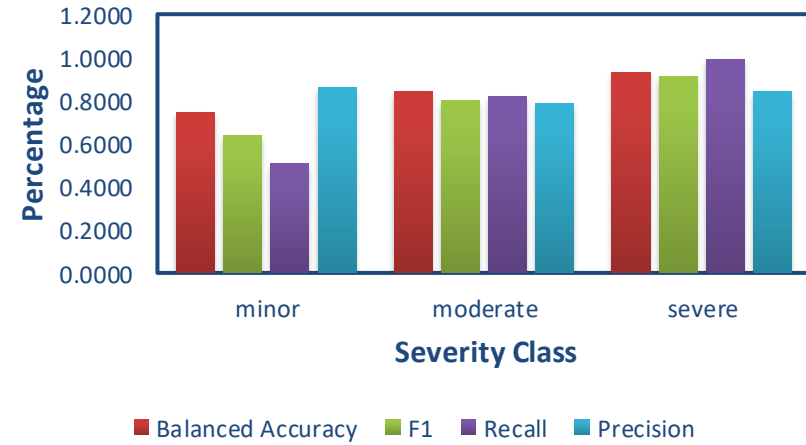


Figure 4.6: Bar charts presenting classification performance of RF model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The radar charts in Figure 4.7 provide a complementary visualization of these metrics. Figure 4.7a, representing Balanced Accuracy, shows an almost uniform distribution near the maximum value of 1.000 across all leak types, further validating the consistency of the model's performance. Similarly, Figure 4.7b and Figure 4.7c illustrate strong and uniform performance, with only minor deviations for the test dataset in certain leak types, such as LC and NL. Figure 4.7d, which visualizes Precision, highlights comparable results, indicating the model's reliability in predicting true positives while minimizing false positives. These visualizations collectively reinforce the findings in Table 4.4, confirming the RF model's capability in achieving high predictive accuracy and consistency across all datasets and leak types for Branched WDS leak detection. Despite the slight variations in specific metrics for the test dataset, the overall performance remains exceptional, indicating the effectiveness of preprocessing and model optimization techniques in this study.

Table 4.4: By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

dataset	Class	Pos Pred Value	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.9996	1.0000	0.9996	0.9998	0.9997	0.9999
	GL	1.0000	0.9999	1.0000	0.9998	0.9999	0.9999
	LC	0.9999	0.9999	0.9999	0.9997	0.9998	0.9999
	NL	0.9996	1.0000	0.9996	1.0000	0.9998	0.9999
	OL	1.0000	1.0000	1.0000	0.9997	0.9999	0.9999
Test	CC	0.9999	1.0000	0.9999	0.9998	0.9999	0.9999
	GL	1.0000	0.9999	1.0000	0.9995	0.9997	0.9997
	LC	0.9998	1.0000	0.9998	0.9999	0.9998	0.9999
	NL	0.9994	1.0000	0.9994	1.0000	0.9997	0.9999
	OL	1.0000	1.0000	1.0000	0.9999	0.9999	0.9999
Train	CC	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	GL	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	LC	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	NL	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	OL	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

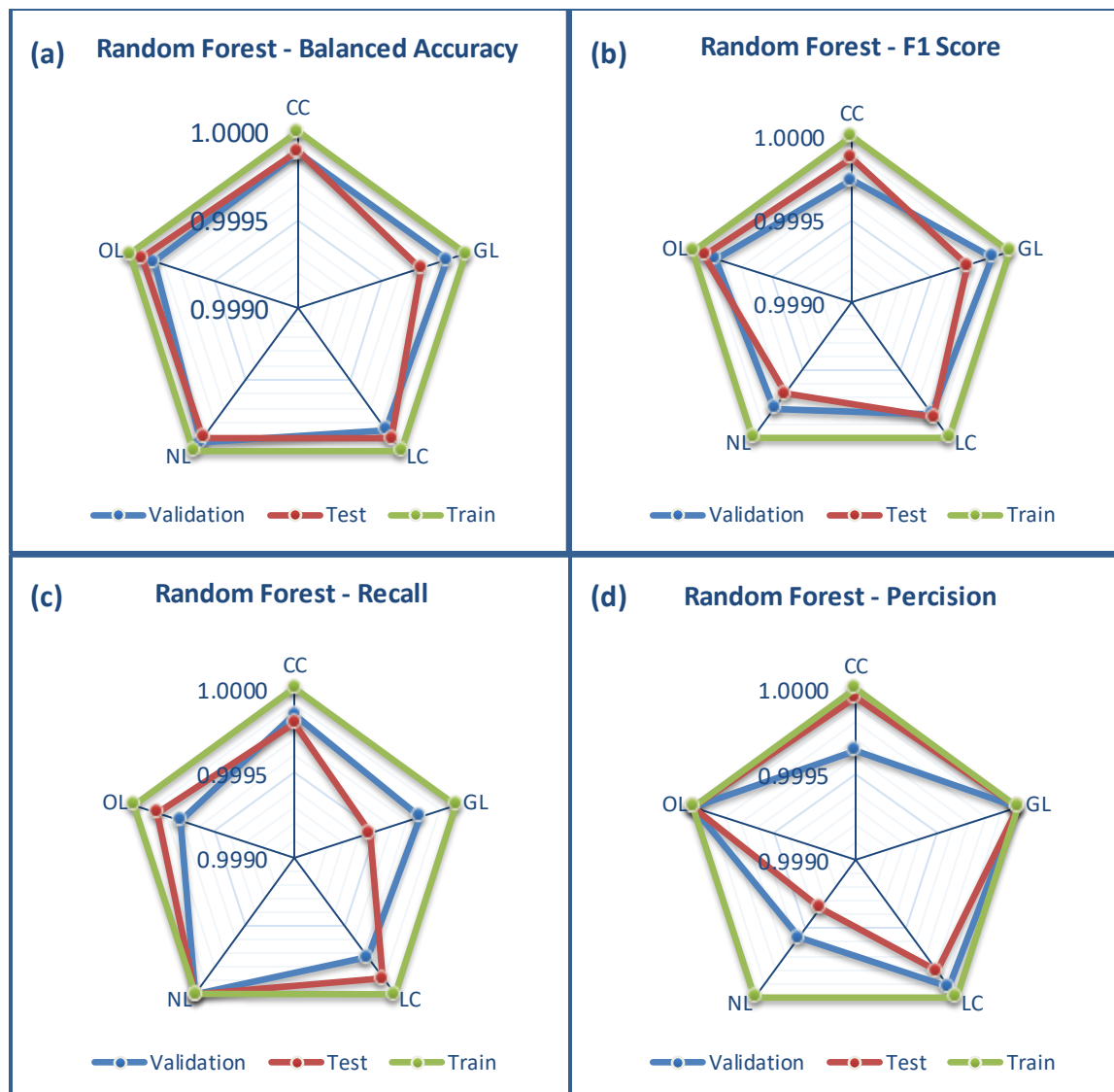


Figure 4.7: Radar charts illustrating RF model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The classification performance of LightGBM models for branched severity assessment was evaluated using multiple metrics, as summarized in Table 4.5. Each model's effectiveness in predicting minor, moderate, and severe cases was assessed through specificity, negative predictive value, precision, recall, F1-score, and balanced accuracy.

Among the models, BR_CC demonstrated high specificity across severity levels, with values of 0.99 (minor), 0.97 (moderate), and 0.92 (severe). The precision was also strong, particularly for moderate cases (0.95) and minor cases (0.96). However, the recall for severe cases (0.82) suggests that a considerable proportion of actual severe instances were misclassified. These trends impacted the balanced accuracy score, which remained high at 0.93 but was slightly lower for the severe class. BR_LC exhibited similar classification trends, with specificity values of 0.97 (minor), 0.94 (moderate), and 0.92 (severe). While its precision for severe cases (0.91) was higher than BR_CC, the recall for severe cases (0.78) indicated challenges in capturing all severe instances correctly. The F1-scores remained high across all classes, particularly for the moderate class (0.91), leading to an overall balanced accuracy of 0.88.

The BR_GL model, while maintaining good specificity for minor (0.98) and moderate (0.88) cases, showed a lower precision for severe cases (0.87) and a significantly lower recall for severe cases (0.57). This contributed to a relatively lower F1-score (0.71) for severe cases and an overall balanced accuracy of 0.78, highlighting a misclassification tendency for severe cases. Similarly, BR_OL showed a decrease in recall for severe cases (0.98), which contrasted with its lower precision (0.82). The model's balanced accuracy (0.91) suggests a reasonable trade-off, but the misclassification of severe cases remained evident.

The confusion matrices in Figure 4.8 further illustrate the classification behavior of these models. In Figure 4.8a, there were 47,312 misclassified severe cases as minor, which had a significant impact on the recall score. A similar misclassification pattern was observed in Figure 4.8b, where 44,698 severe cases were confused with minor cases,

resulting in a severe class recall of 0.57. Figure 4.8c revealed 46,811 misclassified severe cases into the minor class, reinforcing the trend of misclassification for extreme cases. Finally, Figure 4.8d showed the highest misclassification of severe cases (47,219 as minor), emphasizing the difficulty of distinguishing extreme severity levels.

The bar charts in Figure 4.9 provide a visual representation of the key performance metrics discussed in Table 4.5, reinforcing the observed classification trends of the LightGBM models. The charts illustrate the distribution of balanced accuracy, F1-score, recall, and precision across different severity levels (minor, moderate, and severe), further validating the numerical findings from the table and the confusion matrices in Figure 4.8. These figures collectively validate the claim that LightGBM models perform well in distinguishing minor and moderate cases but struggle with severe classification, necessitating refinements in decision thresholds, class weighting strategies, and feature representation techniques to improve predictive reliability.

In the leak detection task for a Branched WDS, the LightGBM model demonstrates near-perfect accuracy across all datasets and leak types (CC, GL, LC, NL, OL), as shown in Table 4.6. Balanced Accuracy remains exceptionally high, exceeding 0.9998 in all cases, with only slight variations in specific leak types. For example, the test dataset shows a Balanced Accuracy of 0.99982397 for CC and 0.999980634 for LC and NL, indicating strong generalization. Similarly, F1 Scores remain consistently high, exceeding 0.9999 across all categories, reinforcing the model's ability to balance precision and recall effectively.

Table 4.5: By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

LightGBM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9926	0.9769	0.9599	0.8828	0.9197	0.9377
	moderate	0.9665	0.9324	0.9481	0.8973	0.9220	0.9319
	severe	0.9254	0.9859	0.9078	0.9823	0.9436	0.9538
BR_LC	minor	0.9797	0.9377	0.9195	0.7808	0.8445	0.8803
	moderate	0.9450	0.9469	0.9093	0.9122	0.9107	0.9286
	severe	0.9324	0.9868	0.9044	0.9808	0.9410	0.9566
BR_GL	minor	0.9833	0.9056	0.9095	0.6217	0.7385	0.8025
	moderate	0.8787	0.9234	0.8240	0.8863	0.8540	0.8825
	severe	0.9094	0.9704	0.8740	0.9577	0.9140	0.9336
BR_OL	minor	0.9825	0.8977	0.9019	0.5891	0.7127	0.7858
	moderate	0.8842	0.8846	0.8104	0.8110	0.8107	0.8476
	severe	0.8592	0.9817	0.8262	0.9766	0.8951	0.9179

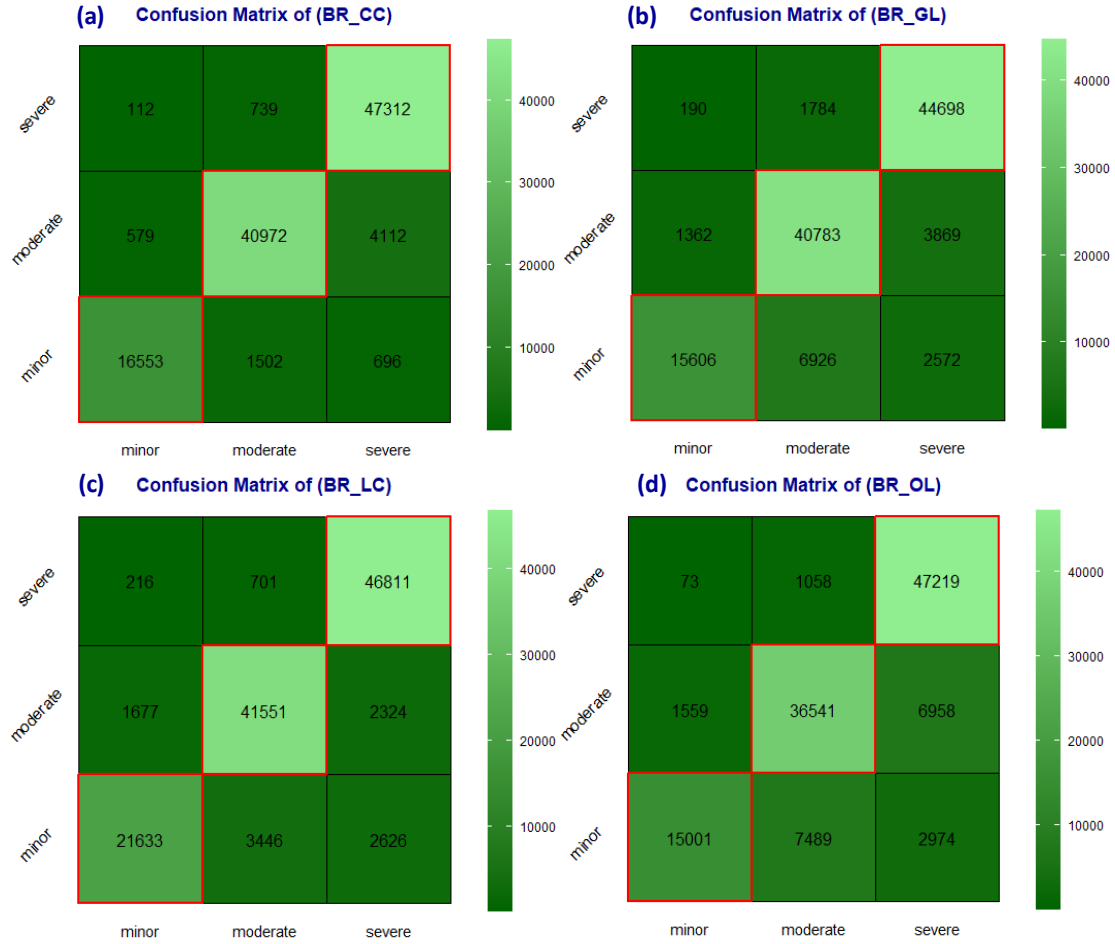
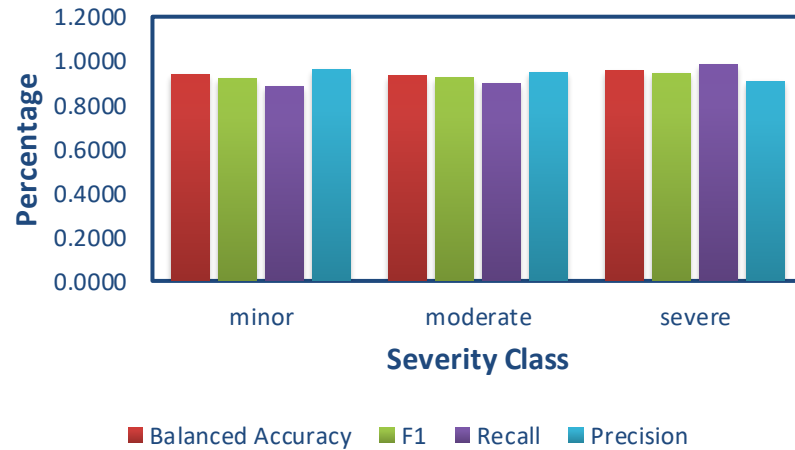
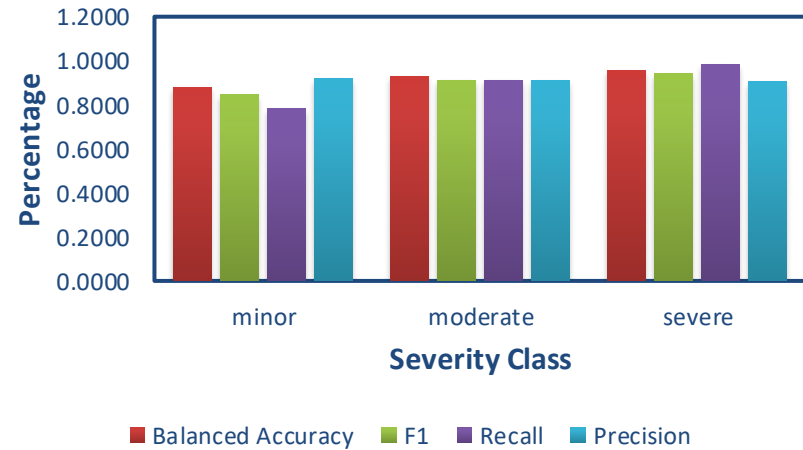


Figure 4.8: Confusion matrices of four Branched network LightGBM models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.

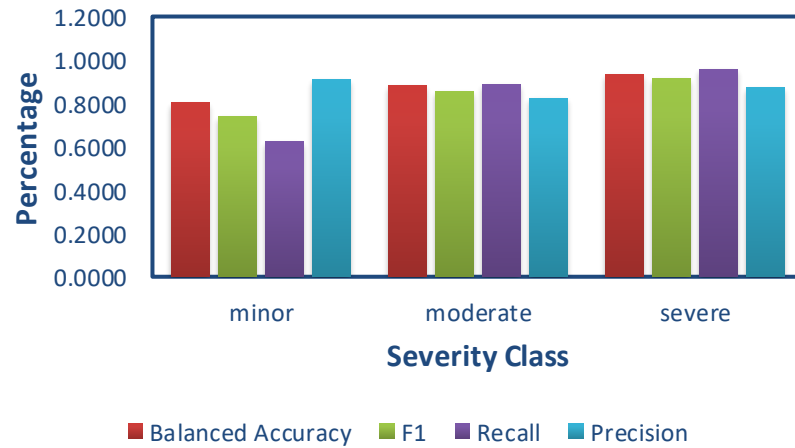
(a) LightGBM- Branched network - CC



(b) LightGBM - Branched network - LC



(c) LightGBM - Branched network - GL



(d) LightGBM - Branched network - OL

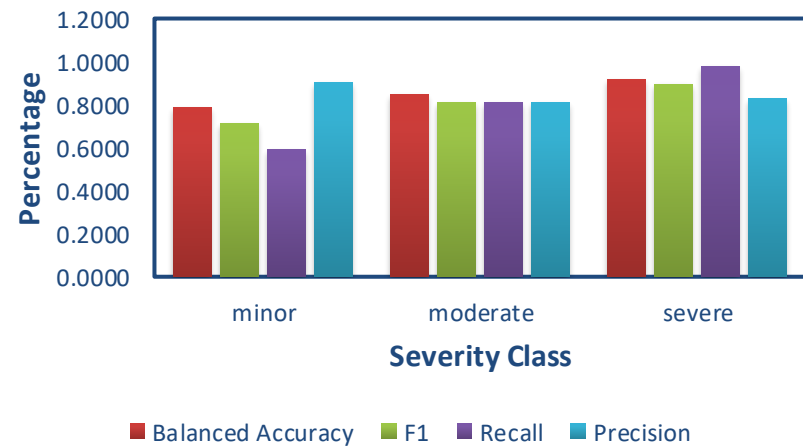


Figure 4.9: Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The radar charts in Figure 4.10 visualize these metrics, revealing minor deviations in the validation and test datasets. Precision for OL in the test dataset is 0.99992761, slightly lower than the training dataset's perfect score, suggesting rare misclassifications. Recall for LC in validation is 0.99991111, slightly below 1, indicating a minor drop in sensitivity. Despite these negligible variations, the model performs with remarkable consistency, highlighting its robustness in real-world leak detection scenarios. Further refinements, such as optimizing thresholds for LC and OL, could enhance performance even further.

Table 4.6: By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.999965	1	0.99984	1	0.99992	0.99998
	GL	1	0.9999756	1	0.99991	0.99995	0.99995
	LC	1	1	1	1	1	1
	NL	1	1	1	1	1	1
	OL	1	0.999989	1	0.99992	0.99996	0.99996
Test	CC	0.999965	1	0.99984	1	0.99992	0.99998
	GL	1	0.9999756	1	0.99991	0.99995	0.99995
	LC	1	0.999987	1	0.99996	0.99998	0.99998
	NL	1	1	1	1	1	1
	OL	0.999989	0.999989	0.99992	0.99993	0.99992	0.99996
Train	CC	1	1	1	1	1	1
	GL	1	1	1	1	1	1
	LC	1	1	1	1	1	1
	NL	1	1	1	1	1	1
	OL	1	1	1	1	1	1

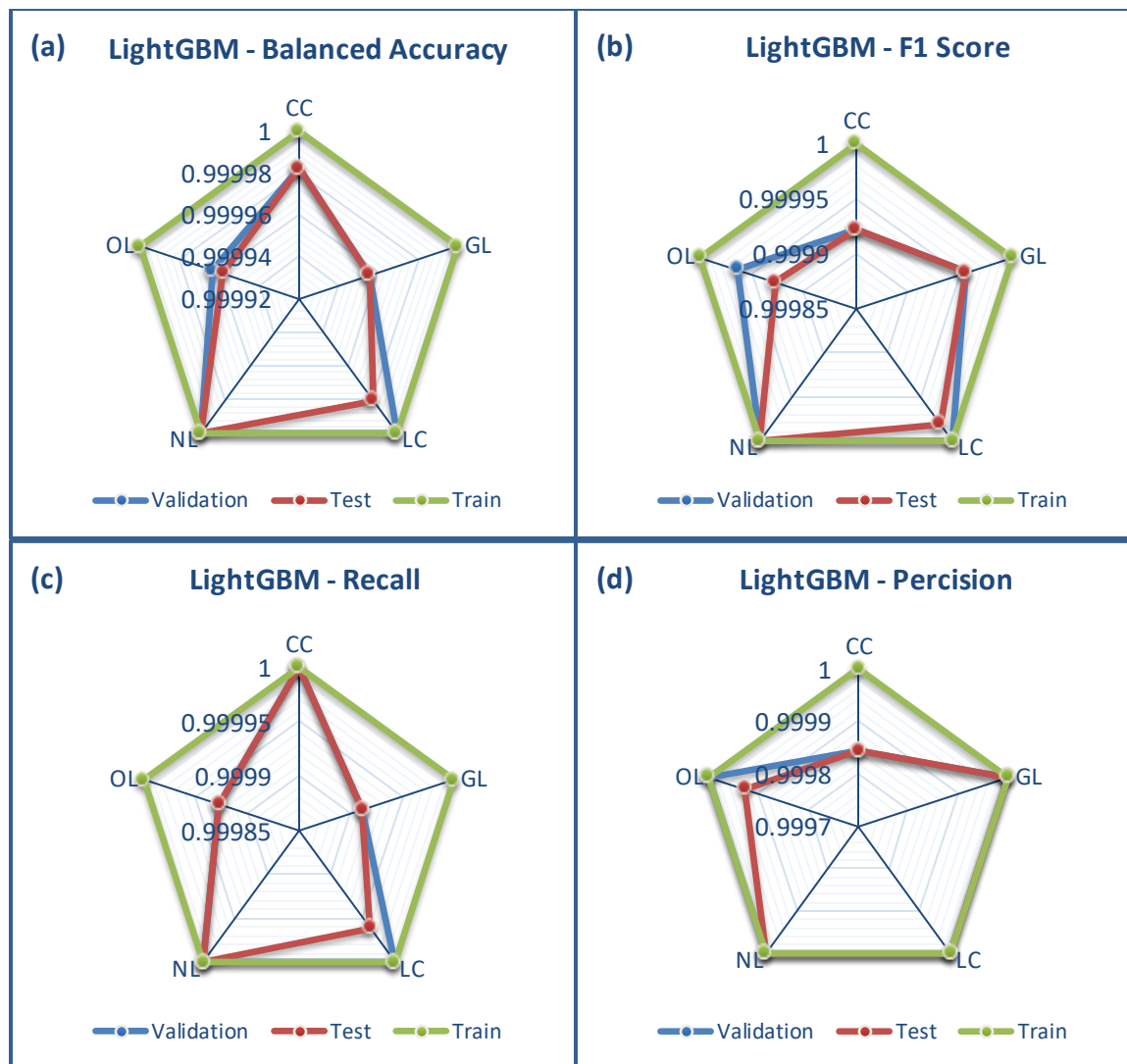


Figure 4.10: Radar charts illustrating LightGBM model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The evaluation of XGB models in Table 4.7 reveals that they generally perform well in distinguishing between moderate and severe cases but struggle more with minor case classification. The confusion matrices in Figure 4.11 indicate that certain severity classes, particularly minor cases, suffer from relatively lower recall, potentially due to data scarcity in those classes.

Among the models, BR_CC shows strong classification performance across severity levels. The confusion matrix in Figure 4.11a indicates that 15,580 minor cases, 39,344 moderate cases, and 46,516 severe cases were correctly classified, aligning well with the high recall and precision scores in Table 4.7. However, a notable proportion of minor cases were misclassified as moderate (2,286) and severe (885), suggesting that distinguishing minor cases from more severe ones remains a challenge.

Similarly, BR_LC exhibits good classification performance for moderate and severe cases, with 40,038 moderate cases and 46,200 severe cases correctly identified, as shown in Figure 4.11c. However, minor cases have a lower recall (0.71), as indicated by 19,921 minor cases being correctly classified, but a significant number (4,664) being misclassified as moderate. This suggests that the model struggles to differentiate between minor and moderate cases, potentially due to class imbalances in the training data.

Table 4.7: By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

XGB Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9868	0.9669	0.9265	0.8309	0.8761	0.9089
	moderate	0.9445	0.9091	0.9137	0.8616	0.8869	0.9030
	severe	0.9040	0.9725	0.8826	0.9658	0.9223	0.9349
BR_LC	minor	0.9687	0.9207	0.8721	0.7190	0.7882	0.8439
	moderate	0.9223	0.9266	0.8722	0.8790	0.8756	0.9006
	severe	0.9175	0.9778	0.8844	0.9680	0.9243	0.9428
BR_GL	minor	0.9700	0.8822	0.8250	0.5216	0.6391	0.7458
	moderate	0.8407	0.8894	0.7711	0.8370	0.8027	0.8388
	severe	0.8880	0.9595	0.8468	0.9429	0.8923	0.9155
BR_OL	minor	0.9666	0.8730	0.7980	0.4843	0.6028	0.7255
	moderate	0.8450	0.8423	0.7447	0.7408	0.7428	0.7929
	severe	0.8276	0.9682	0.7925	0.9604	0.8684	0.8940

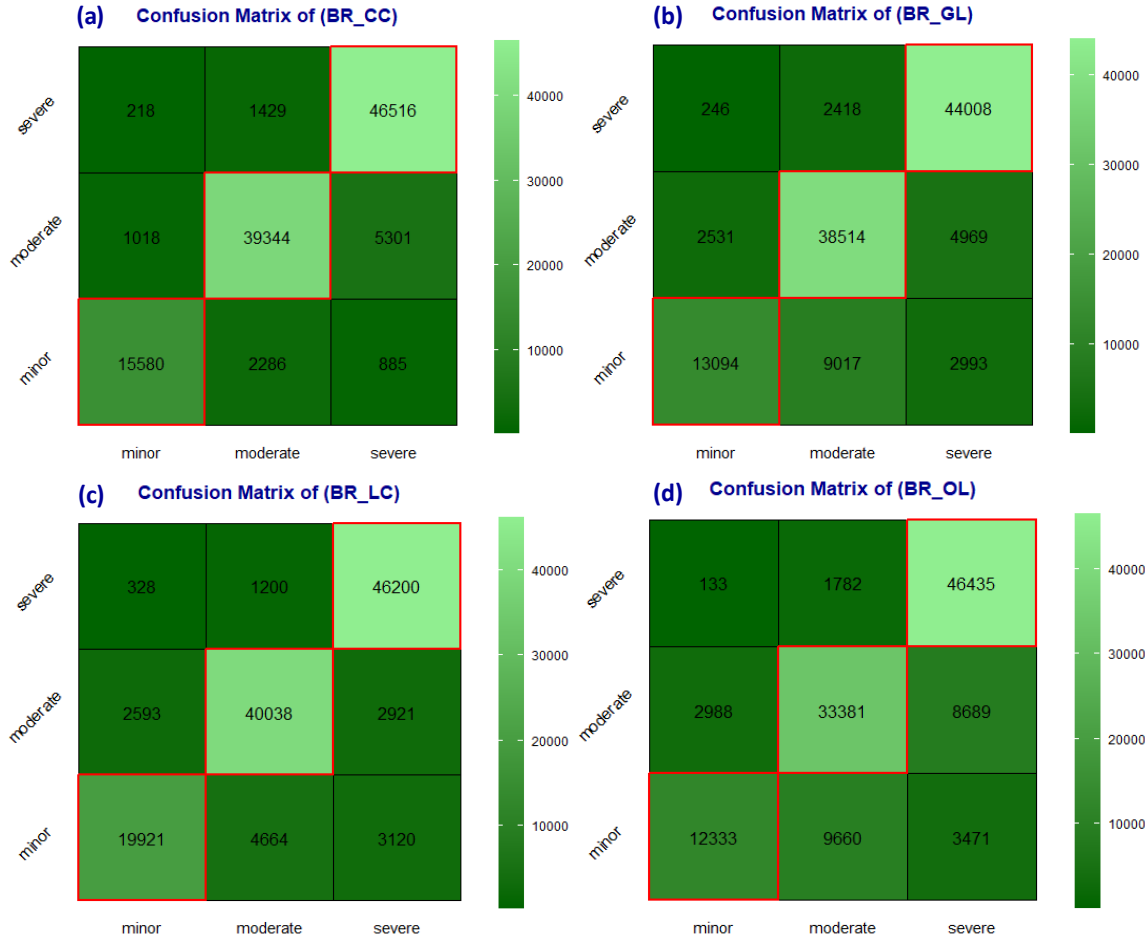


Figure 4.11: Confusion matrices for four Branched network XGB models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.

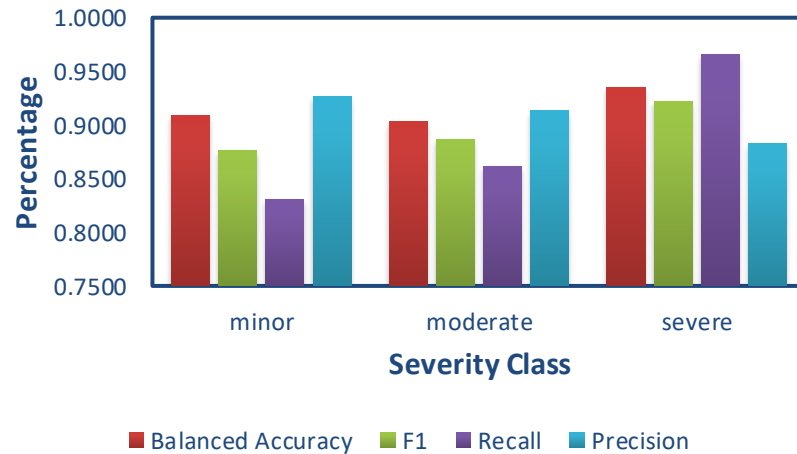
The BR_GL model further highlights this challenge, particularly in minor case classification. As seen in Figure 4.11b, 13,094 minor cases were correctly classified, but a high number (9,017) were confused with moderate cases, aligning with the lower recall for minor cases (0.52) in Table 4.7. In contrast, moderate (38,514 correctly classified) and severe (44,008 correctly classified) cases show stronger performance, reinforcing the model's tendency to favor higher severity levels at the expense of misclassifying minor cases.

A similar trend is observed in BR_OL, where Figure 4.11d shows that 12,333 minor cases were correctly classified, but a substantial portion (9,660) were misclassified as moderate, explaining the low recall (0.48) for minor cases in Table 4.7. Meanwhile, moderate and severe cases demonstrate better classification performance, with 33,381 moderate and 46,435 severe cases correctly identified, further supporting the idea that the model favors higher severity classes over minor ones.

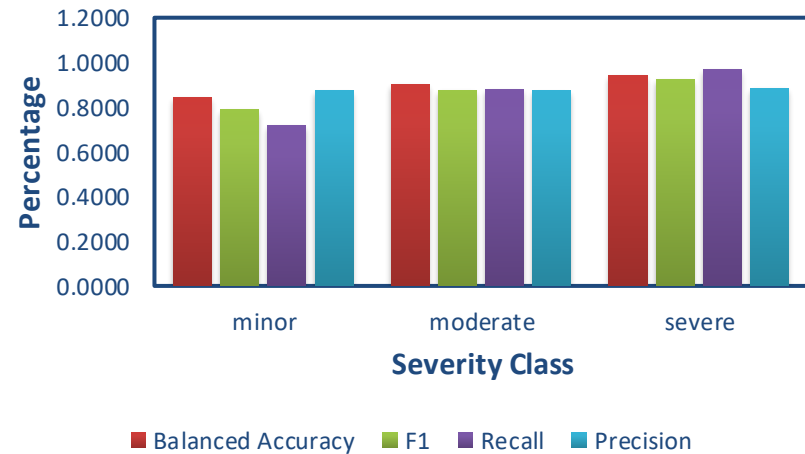
The bar charts in Figure 4.12 further illustrates these trends. Figure 4.12a shows relatively well-balanced precision, recall, and F1-scores across severity levels, confirming that BR_CC maintains solid overall performance. However, Figure 4.12d highlights a drop in recall for minor cases, aligning with the confusion matrix trends.

The most significant discrepancy is observed in Figure 4.12c and Figure 4.12d, where recall for minor cases is notably lower than for moderate and severe cases, reflecting the substantial misclassification of minor instances into higher severity classes. This further supports the notion that XGB models are biased towards classifying cases as moderate or severe, potentially due to an uneven distribution of training data favoring these classes over minor cases.

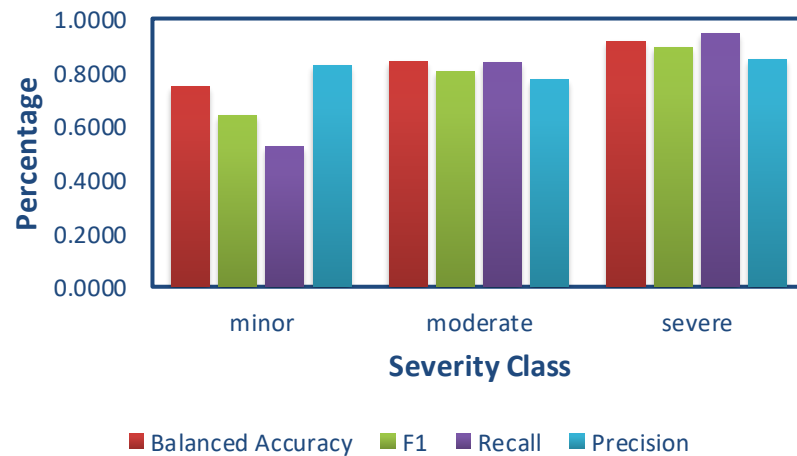
(a) XGB - Branched network - CC



(b) XGB - Branched network - LC



(c) XGB - Branched network - GL



(d) XGB - Branched network - OL

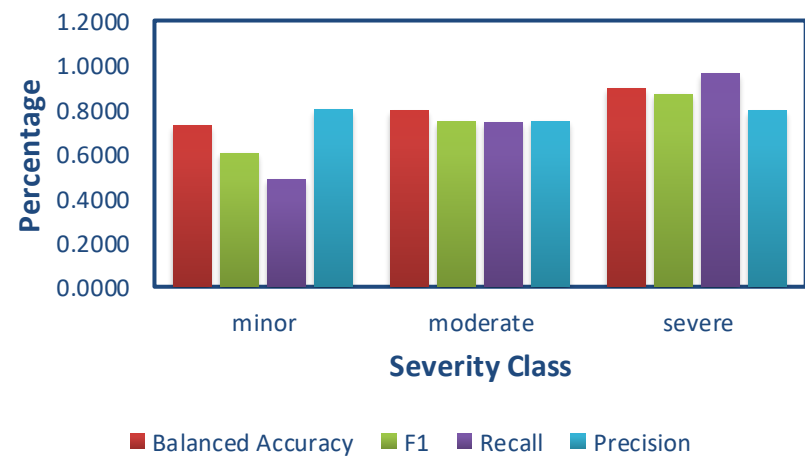


Figure 4.12: Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The XGB models demonstrate strong classification performance for moderate and severe cases but struggle with minor case classification, as evidenced by lower recall values and substantial misclassification of minor instances into moderate and severe classes. The most impacted models appear to be BR_GL and BR_OL, where minor case recall is particularly low (0.52 and 0.48, respectively), suggesting a systemic difficulty in identifying these cases correctly. This imbalance could indicate that the training dataset contains significantly fewer minor cases compared to moderate and severe cases, leading the models to favor higher severity classifications. Addressing this issue could involve data augmentation strategies, class-weighted learning approaches, or improved feature representation for minor cases to ensure a more balanced and accurate classification across all severity levels.

For the leak detection task in a Branched WDS, the XGB model demonstrates exceptional accuracy across all datasets, as detailed in Table 4.8. Balanced Accuracy remains consistently high, surpassing 0.9995 for most classes, with OL showing 0.9996 in validation and 0.9997 in the test dataset. F1 Scores reflect a similar trend, exceeding 0.9998 for CC and GL across all datasets, highlighting the model's ability to maintain strong precision and recall. Figure 4.13 visualizes the performance across metrics, with radar charts indicating minimal deviation between validation and test results. The lowest observed precision is 0.9995 (NL - validation), while recall scores remain close to 1, ensuring minimal false negatives. With such consistent results, the XGB model exhibits superior generalization and reliability in leak classification within a Branched WDS, making it a robust choice for high-accuracy detection.

Table 4.8: By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99993	0.99995	0.99969	0.9998	0.99974	0.99986
	GL	0.99997	0.99993	0.9999	0.9997	0.9998	0.99985
	LC	0.99987	0.99989	0.99961	0.9996	0.9996	0.99976
	NL	0.9999	1	0.9996	1	0.9998	0.99995
	OL	0.99998	0.9999	0.99986	0.9993	0.9996	0.9997
Test	CC	0.999988	0.9999	0.99995	0.99974	0.9998	0.99986
	GL	0.99995	0.99996	0.9998	0.99987	0.9998	0.9999
	LC	0.99992	0.99994	0.99978	0.9998	0.99979	0.99986
	NL	0.99995	1	0.9998	1	0.99991	0.99997
	OL	0.999956	0.99993	0.9997	0.9996	0.9996	0.99976
Train	CC	1	1	1	1	1	1
	GL	1	1	1	1	1	1
	LC	1	1	1	1	1	1
	NL	1	1	1	1	1	1
	OL	1	1	1	1	1	1

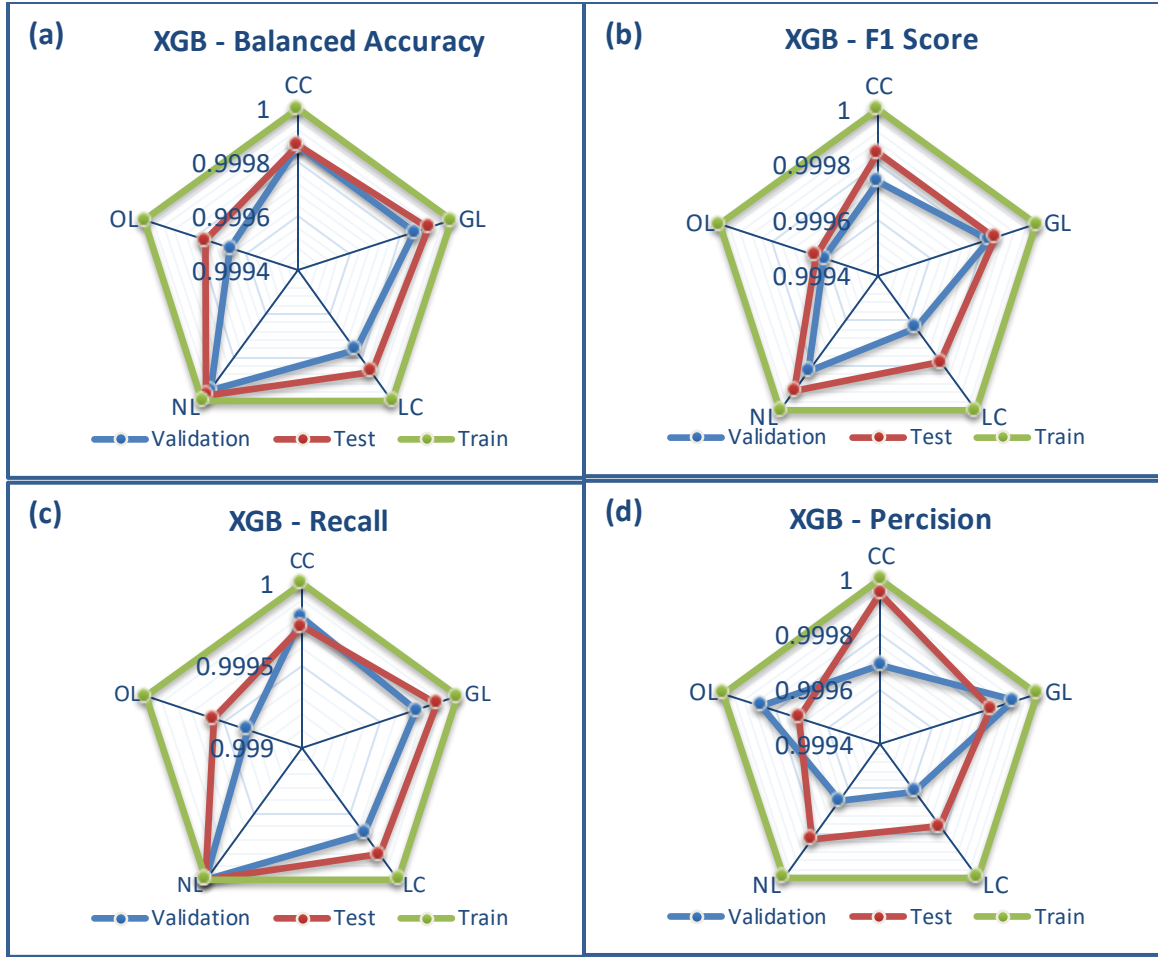


Figure 4.13: Radar charts illustrating XGB model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

4.2.1.2 Deep Learning models

Table 4.9 provides a comparative analysis of three deep learning models (LSTM, RNN, and GRU) across four leak types in a Branched WDS network, using a dataset processed by IQR outlier removal. The evaluation metrics include Accuracy, Kappa, and Confidence Intervals (CI Lower and CI Upper), offering a comprehensive insight into the models' predictive performance. Among the models, GRU demonstrates the highest performance in terms of Accuracy and Kappa for most leak types, particularly BR_CC and BR_LC, indicating its effectiveness in detecting leaks. LSTM shows competitive performance,

closely following GRU, especially in BR_CC and BR_LC, while exhibiting a notable decrease in performance for BR_OL, suggesting sensitivity to certain leak types. RNN, on the other hand, displays balanced performance with Accuracy and Kappa values generally between those of LSTM and GRU, showing relatively lower variance in predictions. The Confidence Intervals further highlight the reliability of these models, with narrower intervals for GRU, suggesting more stable and consistent predictions compared to LSTM and RNN.

Figure 4.14 provides a visual comparison of the performance of three deep learning models (LSTM, RNN, and GRU) across various leak types in a Branched WDS network, following IQR-based outlier removal. The bar heights, representing Accuracy, Kappa, and Confidence Intervals, highlight the effectiveness of each model for different leak types. GRU consistently achieves higher Accuracy and Kappa values, particularly for BR_CC and BR_LC, suggesting its superior capability in leak detection. LSTM shows comparable performance, especially for BR_CC and BR_LC, but experiences noticeable declines for BR_OL, indicating a sensitivity to certain leak conditions. RNN demonstrates more balanced performance, with moderate Accuracy and Kappa values across all leak types, suggesting a stable but less dominant predictive ability. The Confidence Intervals reveal narrower bounds for GRU, implying more reliable predictions, while LSTM and RNN exhibit slightly wider intervals, suggesting higher variance. Overall, the figure illustrates that GRU is the most reliable model for leak detection in a Branched WDS, followed by LSTM and RNN, with distinct performance variations across different leak types.

Table 4.9: Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over branched network after removing the outliers.

Model	Metric	BR_CC	BR_LC	BR_GL	BR_OL
LSTM	Accuracy	0.8512955	0.8330069	0.7460014	0.6821314
	Kappa	0.7598312	0.7398513	0.5939791	0.4860092
	CI Lower	0.8492037	0.8308931	0.7435053	0.6794757
	CI Upper	0.8533693	0.8351046	0.7484854	0.6847782
	No Info Rate	0.4278151	0.3944969	0.3962239	0.4067366
RNN	Accuracy	0.8516864	0.8351972	0.744295	0.6878686
	Kappa	0.7592895	0.7428387	0.586176	0.4914454
	CI Lower	0.8495968	0.8330945	0.7417935	0.6852257
	CI Upper	0.8537578	0.8372838	0.7467846	0.6905022
	No Info Rate	0.4278151	0.3944969	0.3962239	0.4067366
GRU	Accuracy	0.8519706	0.834007	0.7490577	0.6895174
	Kappa	0.7606271	0.7414128	0.5983085	0.5014325
	CI Lower	0.8498826	0.8318983	0.7465714	0.6868783
	CI Upper	0.8540404	0.8360996	0.7515317	0.6921472
	No Info Rate	0.4278151	0.3944969	0.3962239	0.4067366

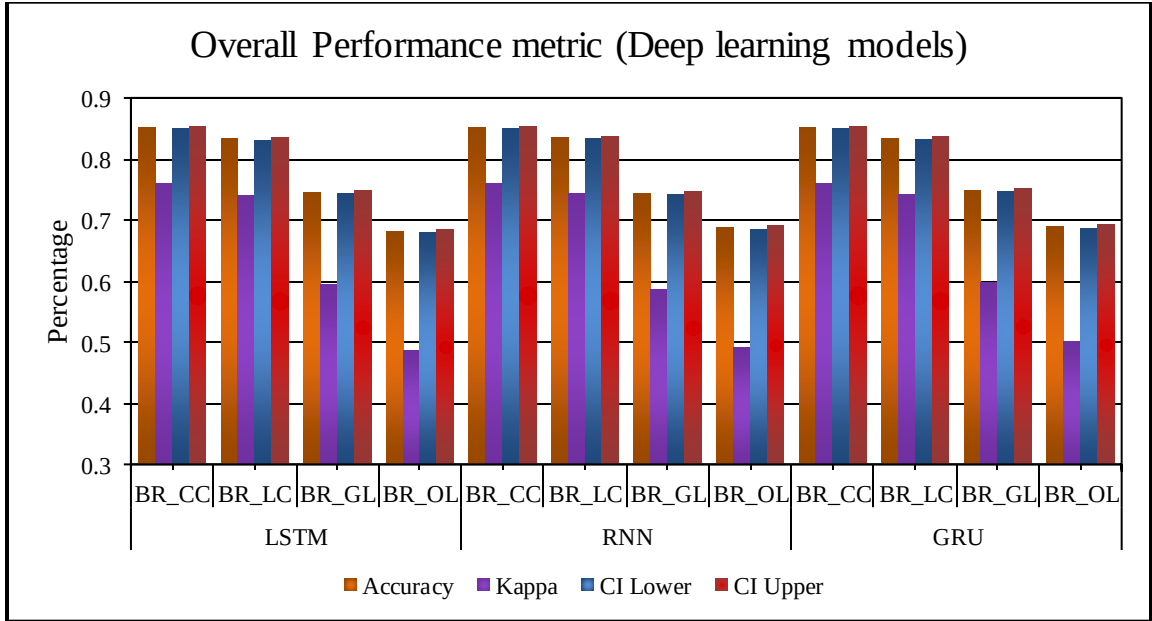


Figure 4.14: Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over branched network after removing the outliers.

Table 4.10: By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

LSTM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9741	0.9535	0.8548	0.7621	0.8058	0.8681
	moderate	0.9179	0.8642	0.8676	0.7886	0.8262	0.8532
	severe	0.8631	0.9549	0.8378	0.9455	0.8884	0.9043
BR_LC	minor	0.9560	0.8968	0.8095	0.6295	0.7082	0.7927
	moderate	0.8865	0.9002	0.8167	0.8373	0.8269	0.8619
	severe	0.8971	0.9630	0.8570	0.9471	0.8998	0.9221
BR_GL	minor	0.9622	0.8464	0.7182	0.3555	0.4756	0.6589
	moderate	0.7746	0.8516	0.6919	0.7894	0.7374	0.7820
	severe	0.8560	0.9377	0.8063	0.9133	0.8565	0.8846
BR_OL	minor	0.9680	0.8196	0.6500	0.2183	0.3268	0.5931
	moderate	0.7504	0.7882	0.6209	0.6697	0.6443	0.7100
	severe	0.7679	0.9476	0.7348	0.9380	0.8241	0.8530

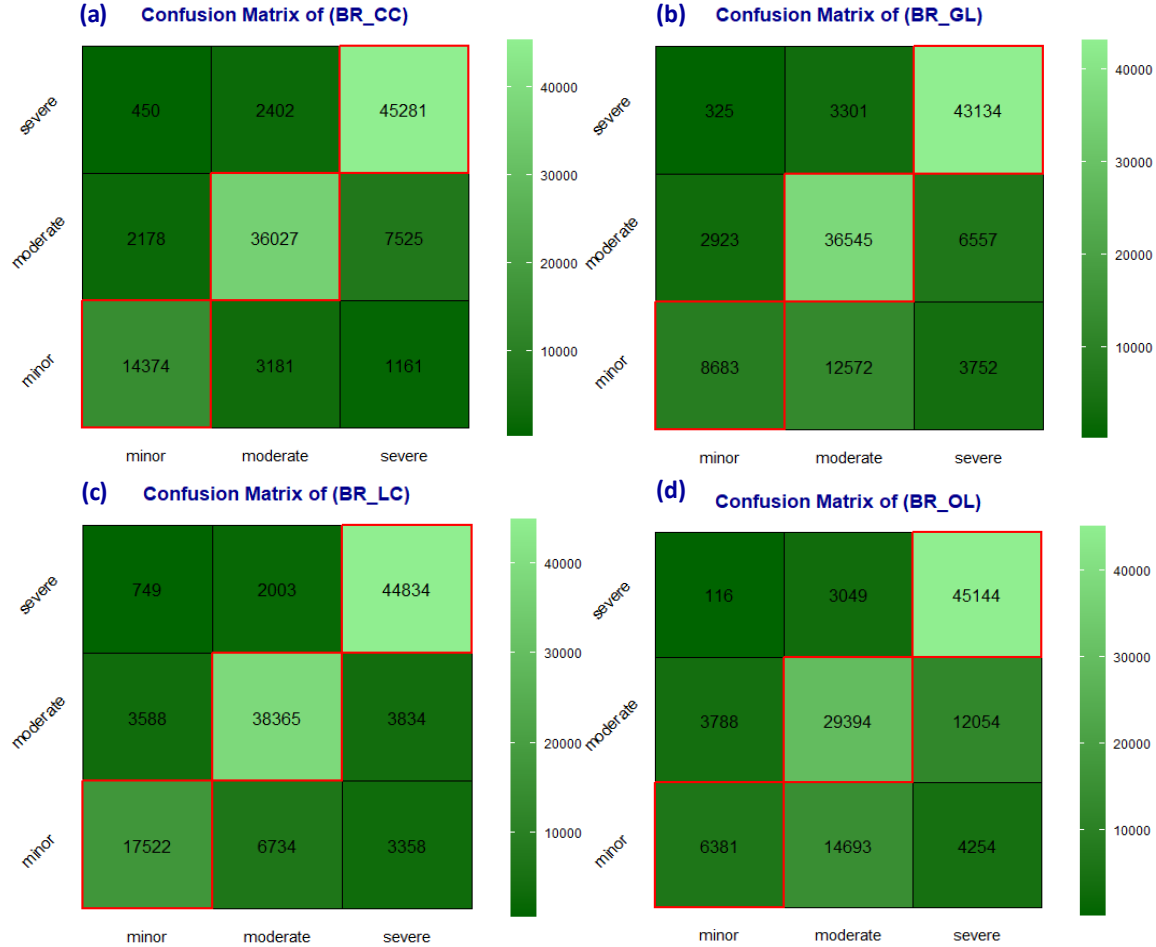


Figure 4.15: Confusion matrices for four Branched network LSTM models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.

The LSTM model, as evaluated in Table 4.10, demonstrates varying degrees of effectiveness across severity classes, with a notable decline in performance for minor cases. The confusion matrices in Figure 4.15 further highlight the model's classification tendencies, while the bar charts in Figure 4.16 provide a visual representation of the model's overall performance across different severity classes.

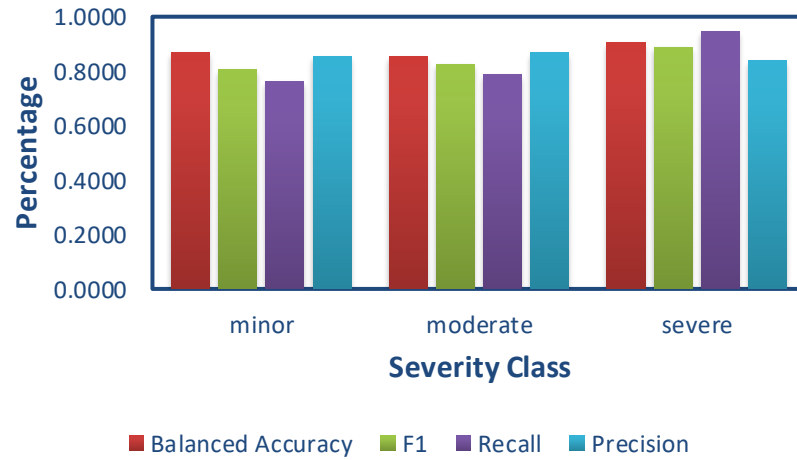
The BR_CC model achieves high specificity across all severity levels (0.97 for minor, 0.91 for moderate, and 0.86 for severe), indicating that it correctly identifies negative cases with strong reliability. However, the recall for minor cases is 0.76, which is lower compared to moderate (0.78) and severe cases (0.94), as seen in Table 4.10. This suggests that while the model captures most severe instances, it tends to under-predict minor cases. Figure 4.15a confirms this by showing that 14,374 minor cases were correctly classified, while 3,181 were misclassified as moderate and 1,161 as severe. The high misclassification rate for minor cases indicates potential class imbalance in the training data. The BR_LC model follows a similar pattern, with a strong recall for severe cases (0.94) but a significantly lower recall for minor cases (0.62). This is evident in Figure 4.15c, where 17,522 minor cases were correctly classified, but 6,734 were misclassified as moderate and 3,358 as severe. The difficulty in distinguishing minor cases suggests that the model has a bias toward classifying instances as moderate or severe, possibly due to an uneven distribution of training data.

The BR_GL model struggles more significantly with minor case classification. As seen in Table 4.10, the recall for minor cases drops to 0.35, which is the lowest among all models. This aligns with Figure 4.15b, where only 8,683 minor cases were correctly classified, while 12,572 were misclassified as moderate and 3,752 as severe. The drastic

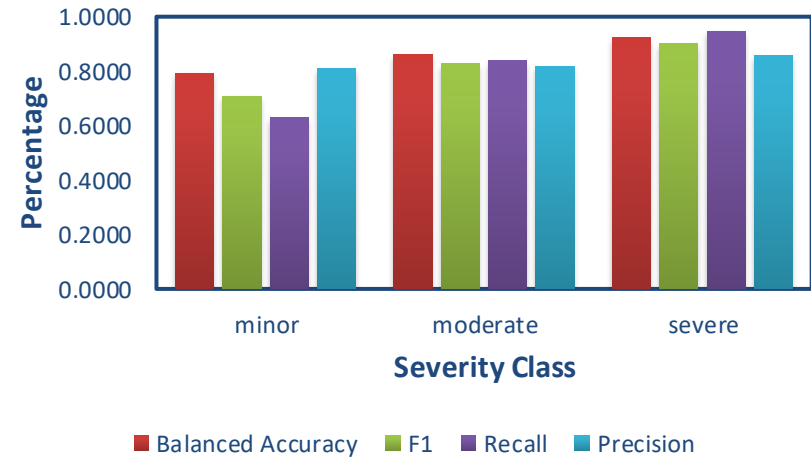
drop in recall indicates that the model lacks sufficient features or representative samples for minor cases, leading to frequent misclassification into more severe categories. The BR_OL model performs the worst in classifying minor cases, with a recall of only 0.21, as seen in Table 4.10. This is reflected in Figure 4.15d, where only 6,381 minor cases were correctly classified, while 14,693 were misclassified as moderate. In contrast, moderate and severe cases show better classification performance, reinforcing the model's tendency to favor higher severity levels at the expense of minor case recognition.

The bar charts in Figure 4.16 provide additional insights into the model's classification tendencies. Figure 4.16a demonstrates relatively balanced performance across all metrics, although recall for minor cases is slightly lower than for severe cases. Figure 4.16b highlights a sharper drop in recall for minor cases, aligning with the confusion matrix trends. The most pronounced performance gap is observed in Figure 4.16c and Figure 4.16d, where recall for minor cases is substantially lower than for moderate and severe cases. This confirms that LSTM models struggle significantly in distinguishing minor severity levels, further supporting the hypothesis that the dataset contains significantly fewer minor instances compared to moderate and severe cases.

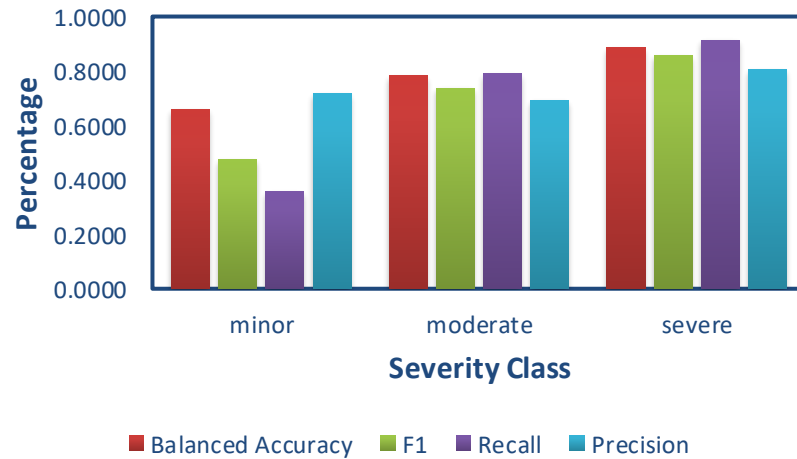
(a) LSTM - Branched network - CC



(b) LSTM - Branched network - LC



(c) LSTM - Branched network - GL



(d) LSTM - Branched network - OL

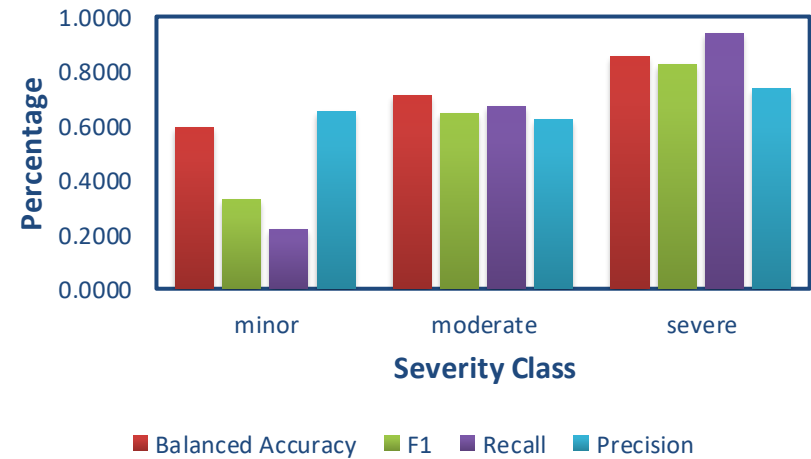


Figure 4.16: Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The LSTM models show strong performance in classifying moderate and severe cases, but their effectiveness in identifying minor cases is significantly lower, as evidenced by poor recall scores and high misclassification rates in Figure 4.15. The most impacted models appear to be BR_GL and BR_OL, where minor case recall is critically low (0.35 and 0.21, respectively), suggesting a major difficulty in correctly recognizing these cases. This imbalance likely stems from an uneven distribution of training samples, where minor cases are underrepresented compared to moderate and severe cases. To improve the performance of the LSTM models, future work should consider data augmentation, re-weighting loss functions, and incorporating additional feature engineering techniques to ensure better classification balance across all severity levels.

For the leak detection task in a Branched WDS, the LSTM model delivers strong performance across all leak types, as shown in Table 4.11. Balanced Accuracy remains high for CC (0.9773) and GL (0.9740) in the test dataset, while OL presents slightly lower values (0.9353), indicating moderate difficulty in classifying this type. The F1 Scores are similarly strong across most categories, with values above 0.95, ensuring a balanced trade-off between precision and recall.

Figure 4.17 visualize the model's performance, revealing minimal deviation between training and test datasets. Precision for CC and GL surpasses 0.96, while recall values remain above 0.95 for most classes, except OL (0.8765). These results suggest that while the LSTM model generalizes well, minor misclassifications persist for certain leak types, which could be improved through additional tuning. Overall, the model demonstrates a reliable capability for leak detection in a Branched WDS.

Table 4.11: By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.9865	0.99275	0.94184	0.968	0.95475	0.97727
	GL	0.99186	0.988	0.96994	0.95617	0.9630	0.9740
	LC	0.98216	0.98953	0.946887	0.96837	0.9575	0.9753
	NL	0.99389	0.99859	0.9788	0.9951	0.98688	0.9945
	OL	0.99421	0.98141	0.9585	0.87649	0.91566	0.93535
Train	CC	0.9869	0.9930	0.94355	0.9693	0.95626	0.9781
	GL	0.992577	0.98895	0.9726	0.9596	0.96608	0.9761
	LC	0.98305	0.990	0.9495	0.969987	0.9596	0.9765
	NL	0.99455	0.99875	0.98108	0.99563	0.9883	0.99509
	OL	0.99456	0.98238	0.9612	0.88299	0.92042	0.9388

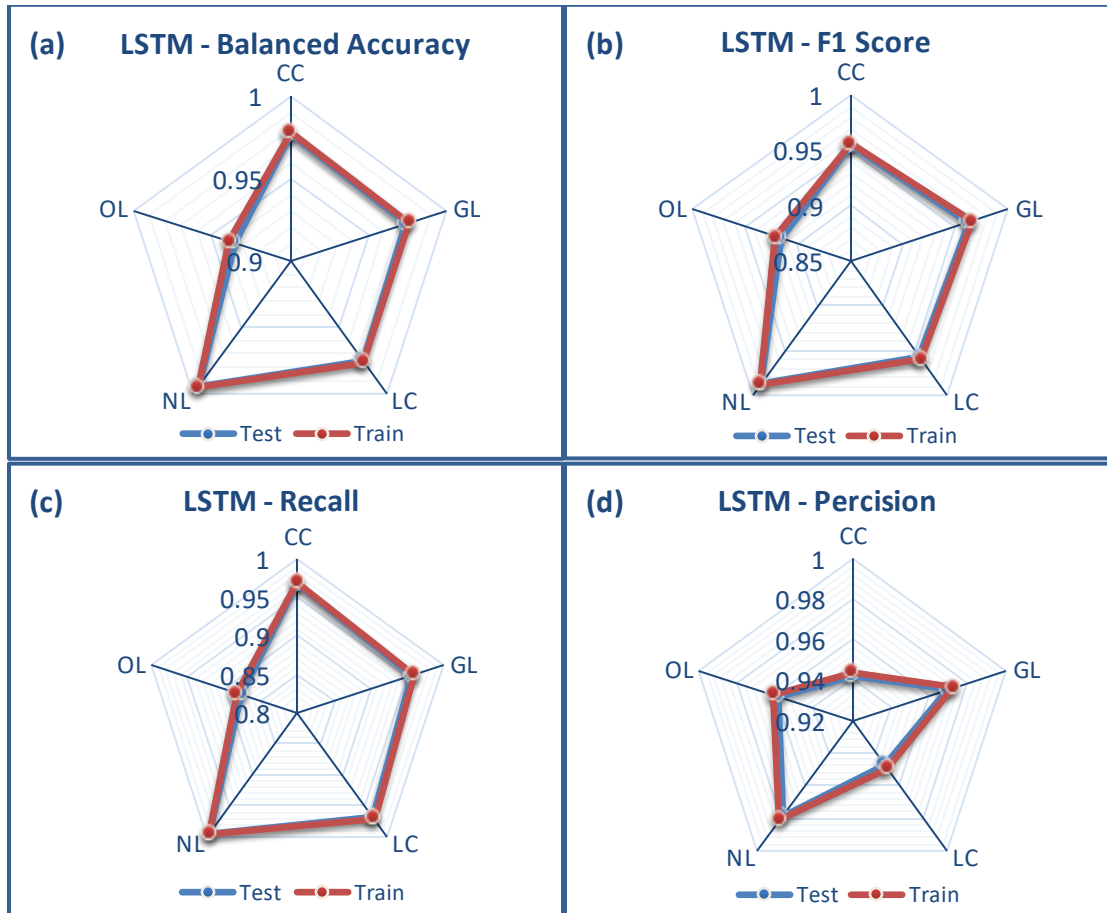


Figure 4.17: Radar charts illustrating LSTM model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The RNN model, as evaluated in Table 4.12, exhibits a performance pattern similar to that observed in other deep learning models, with strong classification ability for severe cases but a noticeable decline in recall for minor cases. The confusion matrices in Figure 4.18 further illustrate these trends, while the bar charts in Figure 4.19 provide a visual representation of how the model performs across different severity classes.

The BR_CC model achieves relatively high specificity across severity levels (0.98 for minor, 0.91 for moderate, and 0.85 for severe), indicating that the model effectively identifies negative instances. However, recall for minor cases (0.72) is notably lower than for moderate (0.78) and severe cases (0.95), as shown in Table 4.12. This aligns with Figure 4.18a, which reveals that 13,862 minor cases were correctly classified, while 3,374 were misclassified as moderate and 1,480 as severe. The lower recall for minor cases suggests that the model is biased toward classifying instances as moderate or severe, likely due to class imbalance in the training dataset. The BR_LC model follows a similar pattern, with strong recall for severe cases (0.96) but significantly lower recall for minor cases (0.62). This is reflected in Figure 4.18c, where 17,551 minor cases were correctly classified, but 6,395 were misclassified as moderate and 3,668 as severe. The model's tendency to misclassify minor cases as moderate indicates difficulty in learning the distinct features of minor severity, possibly due to an insufficient number of minor instances in the dataset.

Table 4.12: By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

RNN Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9814	0.9478	0.8867	0.7296	0.8005	0.8555
	moderate	0.9199	0.8643	0.8704	0.7884	0.8274	0.8542
	severe	0.8512	0.9654	0.8281	0.9592	0.8888	0.9052
BR_LC	minor	0.9609	0.8951	0.8249	0.6207	0.7084	0.7908
	moderate	0.8928	0.8978	0.8242	0.8316	0.8279	0.8622
	severe	0.8880	0.9736	0.8485	0.9631	0.9022	0.9255
BR_GL	minor	0.9871	0.8284	0.8369	0.2452	0.3793	0.6162
	moderate	0.7425	0.8756	0.6753	0.8355	0.7469	0.7890
	severe	0.8533	0.9440	0.8050	0.9228	0.8599	0.8880
BR_OL	minor	0.9860	0.8113	0.7557	0.1585	0.2621	0.5723
	moderate	0.7403	0.8035	0.6231	0.7034	0.6608	0.7219
	severe	0.7642	0.9588	0.7346	0.9521	0.8294	0.8582

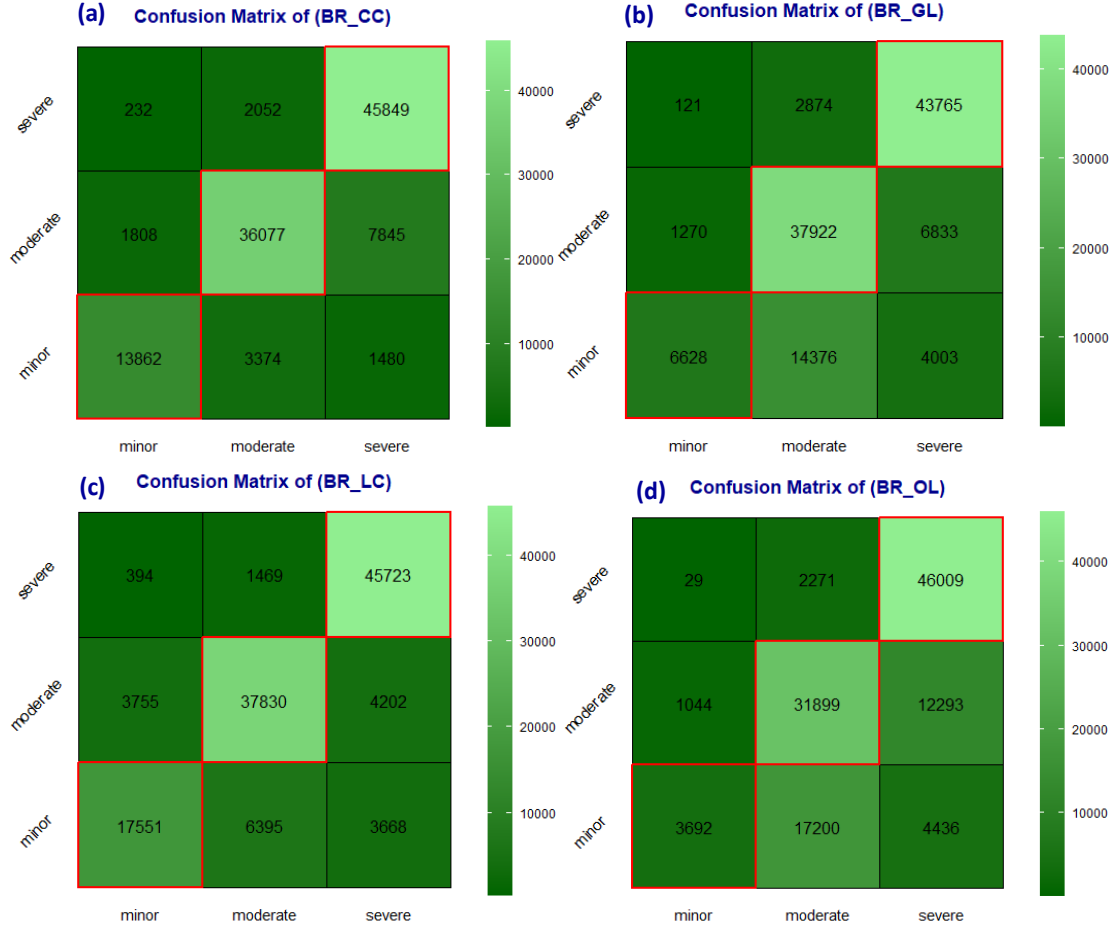


Figure 4.18: Confusion matrices for four Branched network RNN models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.

The BR_GL model performs the worst for minor cases, as evidenced by the lowest recall (0.25) among all models, as shown in Table 4.13. Figure 4.18b further confirms this issue, with only 6,628 minor cases correctly classified, while a substantial 14,376 were misclassified as moderate and 4,003 as severe. This suggests that the model struggles significantly in distinguishing minor cases from more severe ones, likely due to an uneven distribution of training data. The BR_OL model also suffers in minor case classification, with recall dropping to 0.15, the lowest among all RNN-based models. Figure 4.18d confirms this poor performance, as only 3,692 minor cases were correctly classified, while 17,200 were misclassified as moderate. In contrast, moderate and severe cases show better classification performance, with 31,899 moderate cases and 46,009 severe cases correctly identified. This highlights the model's tendency to favor moderate and severe classifications, reinforcing the idea that minor cases are underrepresented in the training set.

The bar charts in Figure 4.19 visually depict these trends. Figure 4.19a demonstrates relatively balanced performance, with minor case recall being slightly lower than moderate and severe cases. Figure 4.19b shows a sharper decline in recall for minor cases, aligning with confusion matrix trends. The most significant discrepancy is observed in Figure 4.19c and Figure 4.19d, where recall for minor cases is significantly lower than for moderate and severe cases, confirming that the RNN models struggle with minor case classification. This suggests that the dataset contains substantially fewer minor instances compared to moderate and severe cases, leading to a classification bias towards higher severity levels.

(a) RNN - Branched network - CC



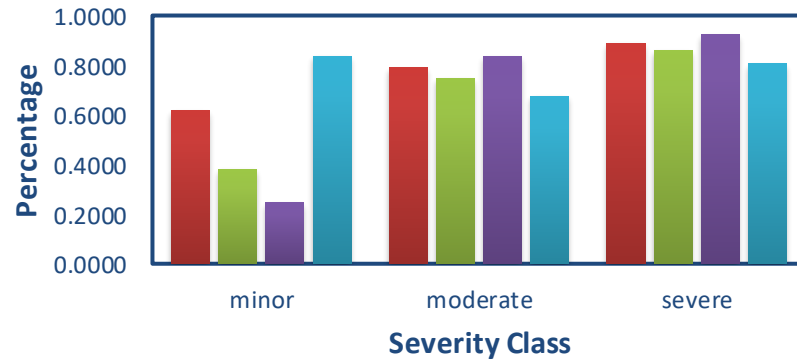
■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(b) RNN - Branched network - LC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(c) RNN - Branched network - GL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(d) RNN - Branched network - OL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

Figure 4.19: Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The RNN models exhibit strong classification performance for moderate and severe cases, but they fail to effectively distinguish minor cases, as evidenced by low recall scores and frequent misclassification in Figure 4.18. The most impacted models appear to be BR_GL and BR_OL, where minor case recall is critically low (0.25 and 0.15, respectively), indicating a systemic challenge in identifying minor instances. This performance discrepancy suggests that minor cases are underrepresented in the training dataset, causing the model to favor moderate and severe classifications. Addressing this imbalance could involve data augmentation techniques, re-weighted loss functions, or improved feature extraction methods to enhance the classification performance for minor cases and ensure a more equitable distribution of classification accuracy across all severity levels.

For the leak detection task in a Branched WDS, the RNN model demonstrates excellent predictive performance, as reflected in Table 4.13. Balanced Accuracy remains high across all classes, exceeding 0.995 for CC and GL, while OL shows a slightly lower score of 0.9909 in the test dataset. The F1 Scores are also consistently strong, with values above 0.99, indicating a well-balanced trade-off between precision and recall.

Figure 4.20 visualizes these metrics, showing minimal variation between training and test results. Precision and Recall values remain high across most leak types, with OL achieving 0.9961 precision but slightly lower recall at 0.9813 in the test dataset. Despite minor fluctuations, the RNN model proves to be a reliable option for leak classification in a Branched WDS, offering robust generalization and accuracy.

Table 4.13: By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99919	0.99968	0.99641	0.99859	0.99750	0.99889
	GL	0.99984	0.99871	0.99943	0.99530	0.99736	0.99757
	LC	0.99707	0.99987	0.99116	0.99959	0.99536	0.99833
	NL	0.99904	0.99997	0.99663	0.99990	0.99826	0.99947
	OL	0.99994	0.99725	0.99961	0.98192	0.99069	0.99093
Train	CC	0.99931	0.99974	0.99696	0.99885	0.99790	0.99908
	GL	0.99993	0.99913	0.99974	0.99682	0.99828	0.99837
	LC	0.99774	0.99993	0.99315	0.99979	0.99646	0.99876
	NL	0.99946	0.99999	0.99810	0.99997	0.99903	0.99971
	OL	0.99996	0.99791	0.99975	0.98628	0.99297	0.99312

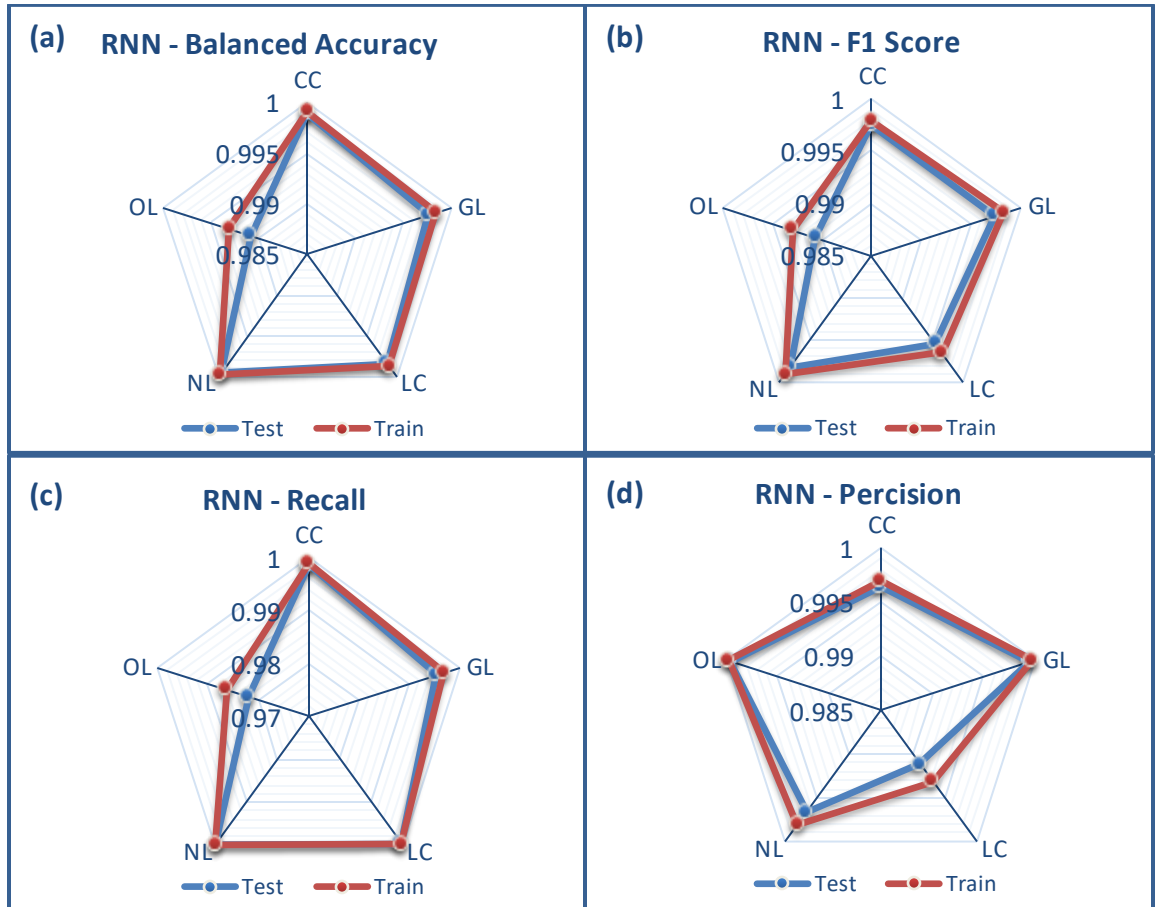


Figure 4.20: Radar charts illustrating RNN model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

For the GRU model, the performance metrics summarized in Table 4.14 indicate strong classification capabilities, particularly in the severe class, where recall values are relatively high across all configurations, suggesting the model effectively identifies severe cases. The balanced accuracy for BR_CC is 0.864, BR_LC is 0.862, BR_GL is 0.785, and BR_OL is 0.854, showing a consistent trend across the different network architectures. However, the recall for minor class instances is relatively low in certain cases, particularly in BR_GL and BR_OL, where recall drops to 0.348 and 0.286, respectively. This may be indicative of data scarcity for this class, leading the model to misclassify minor cases more frequently.

The confusion matrices in Figure 4.21 further illustrate the misclassification tendencies. In BR_CC (Figure 4.21a), a large number of severe cases (45,318) were correctly classified, but misclassification of minor instances remains notable. Similarly, BR_GL (Figure 4.21b) exhibits a high number of correctly classified severe instances (43,628) but misclassifies a significant portion of minor cases into other classes. This trend continues in BR_LC (Figure 4.21c) and BR_OL (Figure 4.21d), where minor cases are often confused with moderate ones, suggesting an imbalance in the model's ability to handle the lower severity levels.

Figure 4.22 provide a visual representation of the model's classification performance. The precision values remain relatively high across all severity classes, but recall shows a drop for minor cases, aligning with the patterns observed in the confusion matrices. This suggests that the GRU model is better suited for recognizing moderate and severe cases, but its ability to accurately classify minor cases remains limited, likely due to an imbalance in the dataset or a lack of sufficiently distinctive features for minor cases.

Table 4.14: By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over branched network after removing the outliers.

GRU Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9766	0.9517	0.8650	0.7520	0.8046	0.8643
	moderate	0.9103	0.8700	0.8589	0.8006	0.8287	0.8554
	severe	0.8687	0.9506	0.8425	0.9396	0.8884	0.9041
BR_LC	minor	0.9563	0.8972	0.8109	0.6310	0.7097	0.7937
	moderate	0.8876	0.9002	0.8181	0.8371	0.8275	0.8623
	severe	0.8973	0.9642	0.8575	0.9489	0.9009	0.9231
BR_GL	minor	0.9662	0.8455	0.7360	0.3483	0.4728	0.6572
	moderate	0.7793	0.8533	0.6967	0.7909	0.7409	0.7851
	severe	0.8512	0.9442	0.8029	0.9234	0.8589	0.8873
BR_OL	minor	0.9556	0.8309	0.6376	0.2869	0.3957	0.6212
	moderate	0.7667	0.7866	0.6330	0.6594	0.6459	0.7130
	severe	0.7797	0.9418	0.7432	0.9297	0.8260	0.8547

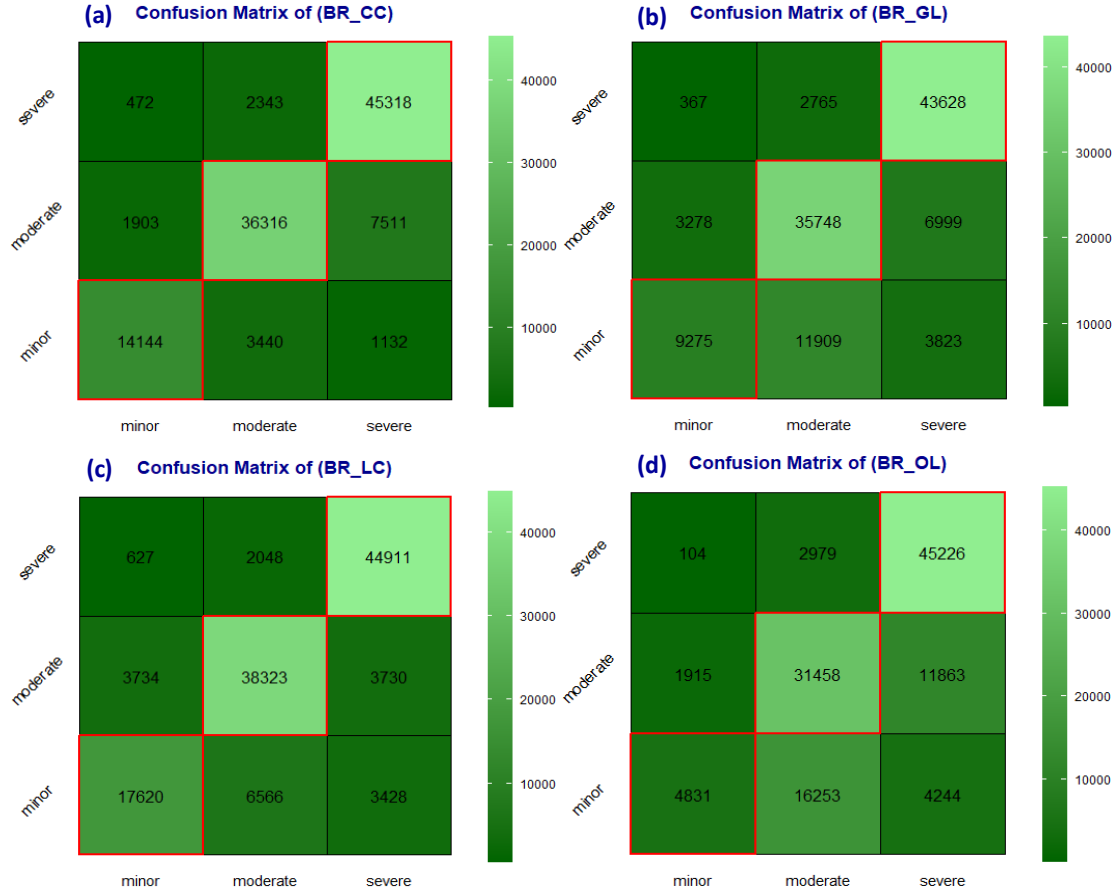
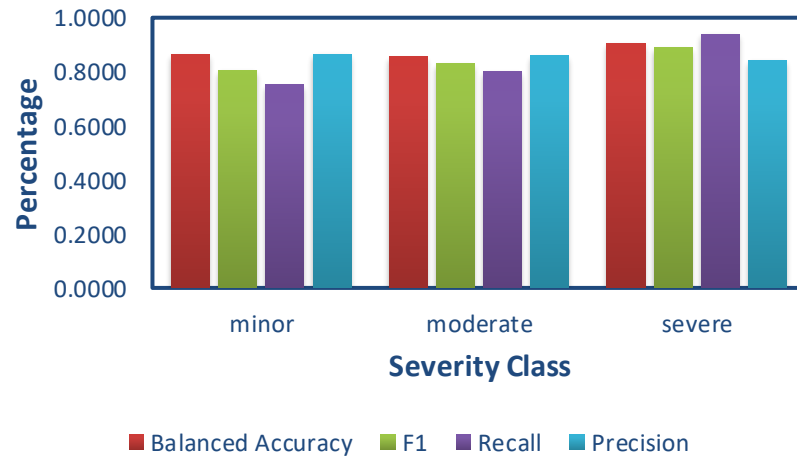
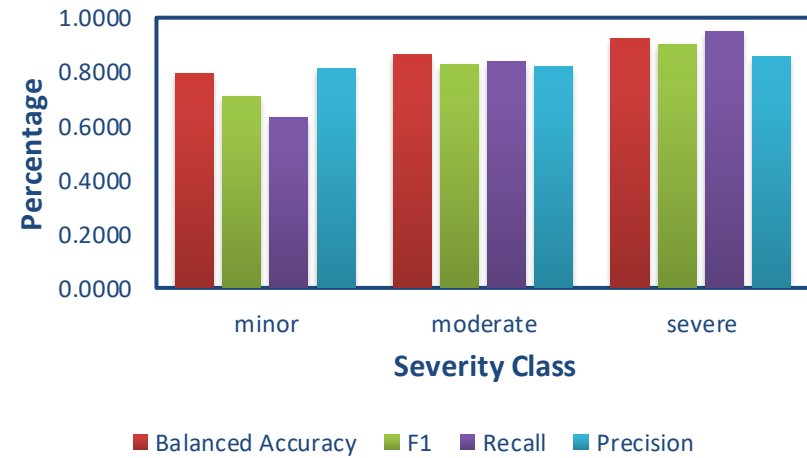


Figure 4.21: Confusion matrices for four Branched network GRU models, providing a granular view of classification accuracy in severity assessment (a) BR_CC (b) BR_GL (c) BR_LC (d) BR_OL.

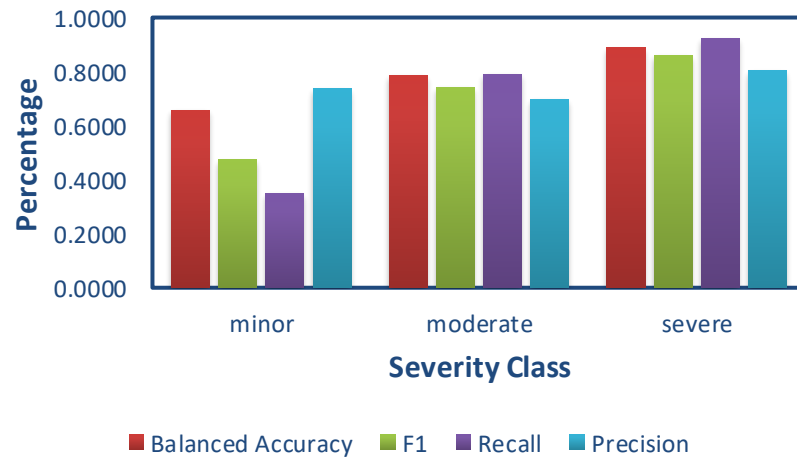
(a) GRU - Branched network - CC



(b) GRU - Branched network - LC



(c) GRU - Branched network - GL



(d) GRU - Branched network - OL

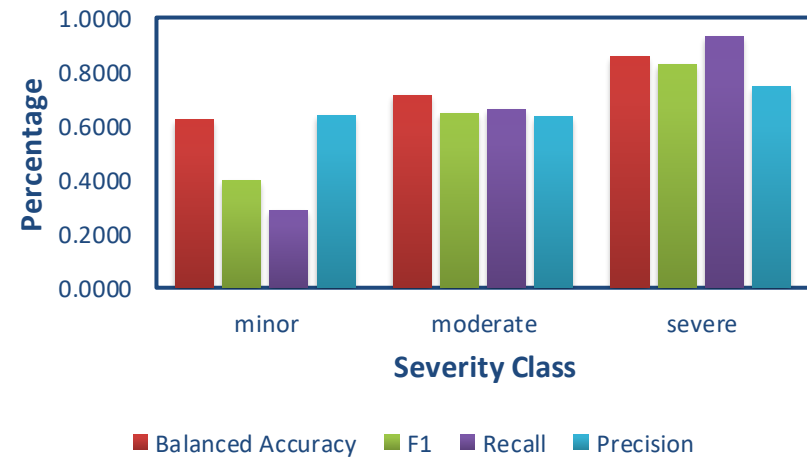


Figure 4.22: Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task after outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The performance of the GRU model in Branched Network Leakage Detection is detailed in Table 4.15 and visualized in Figure 4.23. The model demonstrates high specificity, with values exceeding 0.99 for all classes in both train and test sets, ensuring minimal false positives. The precision scores remain strong, particularly for classes CC, GL, and LC, all above 0.99, while OL records a slightly lower 0.9932 in the test set. Recall values indicate effective detection, with most exceeding 0.99, except for OL, which drops to 0.9301, reflecting a minor weakness in identifying this category. The F1-scores follow similar trends, reaching 0.9970 for CC and 0.9852 for GL, confirming a stable balance between precision and recall.

The radar charts in Figure 4.23 further illustrate these trends. Figure 4.23a highlights balanced accuracy, consistently above 0.99, with a slight dip for OL at 0.9646, aligning with its lower recall. Figure 4.23b presents F1-scores, where all classes perform well, though OL shows a minor decrease, reinforcing its lower recall. Figure 4.23c depicts recall values, emphasizing the 0.9301 recall for OL, while other classes remain above 0.99. Finally, Figure 4.23d confirms precision stability, with all values closely aligned, demonstrating strong classification reliability. Overall, the GRU model performs robustly, though its sensitivity to detecting OL cases could be enhanced.

Table 4.15: By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over branched network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99209	0.99808	0.96579	0.99152	0.97849	0.99181
	GL	0.99807	0.99390	0.99286	0.97770	0.98522	0.98788
	LC	0.99083	0.99673	0.97258	0.99009	0.98126	0.99046
	NL	0.99637	0.99937	0.98733	0.99779	0.99253	0.99708
	OL	0.99903	0.98945	0.99322	0.93013	0.96064	0.96458
Train	CC	0.99294	0.99810	0.96936	0.99161	0.98036	0.99227
	GL	0.99835	0.99488	0.99390	0.98130	0.98756	0.98982
	LC	0.99187	0.99710	0.97565	0.99122	0.98338	0.99155
	NL	0.99702	0.99948	0.98959	0.99819	0.99387	0.99760
	OL	0.99914	0.99074	0.99404	0.93877	0.96561	0.96896

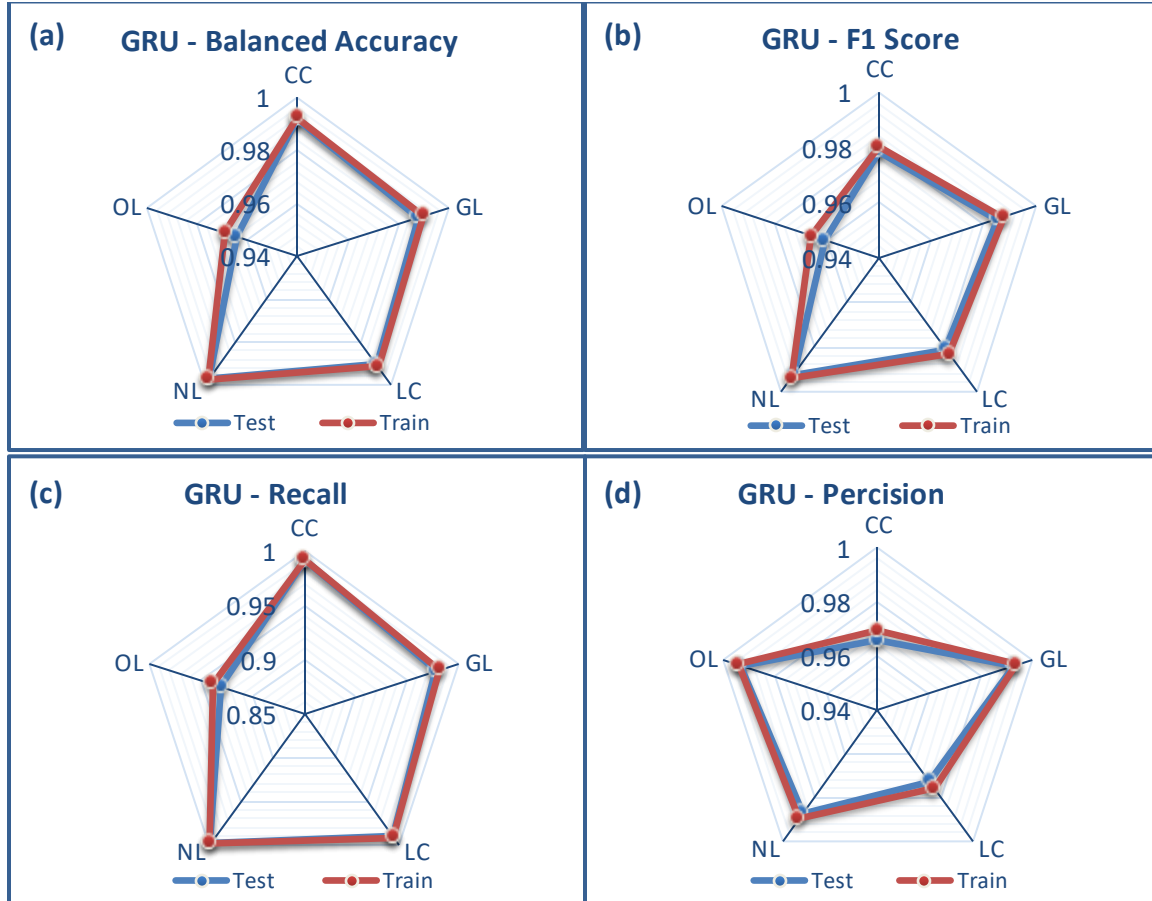


Figure 4.23: Radar charts illustrating GRU model performance after removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

4.2.2 With outliers

4.2.2.1 Tree-based models

The comparison between Table 4.2 and Table 4.16 reveals that tree-based models (XGB, RF, and LightGBM) generally perform better before outlier removal than after outlier removal. For instance, LightGBM shows a slight decline in performance post-outlier removal, with Accuracy for BR_CC decreasing from 0.9310 to 0.9312 and Kappa dropping from 0.8961 to 0.8892. Similarly, RF's Accuracy and Kappa for BR_CC are marginally higher in Table 4.16 (0.9253 Accuracy, 0.8879 Kappa) compared to Table 4.2 (0.9084 Accuracy, 0.8521 Kappa). XGB also displays a drop in performance after outlier removal, with Accuracy for BR_CC decreasing from 0.9189 (Table 4.16) to 0.9011 (Table 4.2) and Kappa falling from 0.8784 to 0.8404. Interestingly, the Confidence Intervals before outlier removal are slightly tighter than those after removing outliers, suggesting higher prediction stability. This trend across all models and leak types implies that the outliers might contain valuable information that enhances the models' ability to classify leak severities more accurately, challenging the assumption that outliers are purely noise.

Analyzing the Random Forest (RF) model performance across different configurations in Table 4.17 (models trained with the full dataset) and Table 4.3 (models trained without outliers) reveals notable differences in key classification metrics. One of the most significant observations is the variation in recall for minor leaks across different network configurations. For instance, in BR_OL, the recall for minor leaks improves from 0.4775 in Table 4.3 to 0.7482 in Table 4.17, demonstrating a considerable enhancement in detecting minor leaks when training with outliers. Similarly, in BR_GL, the recall for minor

leaks increases from 0.5719 to 0.7196, highlighting a more balanced classification for smaller leaks when including the full dataset.

Table 4.16: Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over branched network before removing the outliers.

Model	Metric	BR_CC	BR_LC	BR_GL	BR_OL
XGB	Accuracy	0.91896	0.89197333	0.83687333	0.80223333
	Kappa	0.87844	0.83796	0.75531	0.70335
	CI Lower	0.91756753	0.89039153	0.83499369	0.80020845
	CI Upper	0.92033625	0.89353996	0.83873994	0.80424653
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333
RF	Accuracy	0.92532	0.89342667	0.87604	0.84336
	Kappa	0.88798	0.84014	0.81406	0.76504
	CI Lower	0.92397813	0.89185417	0.87436174	0.8415107
	CI Upper	0.92664541	0.89498394	0.87770371	0.84519602
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333
LightGBM	Accuracy	0.93100667	0.91758	0.87318667	0.83547333
	Kappa	0.89651	0.87637	0.80978	0.75321
	CI Lower	0.92971241	0.9161769	0.87149213	0.83358729
	CI Upper	0.93228423	0.91896694	0.87486676	0.8373464
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333

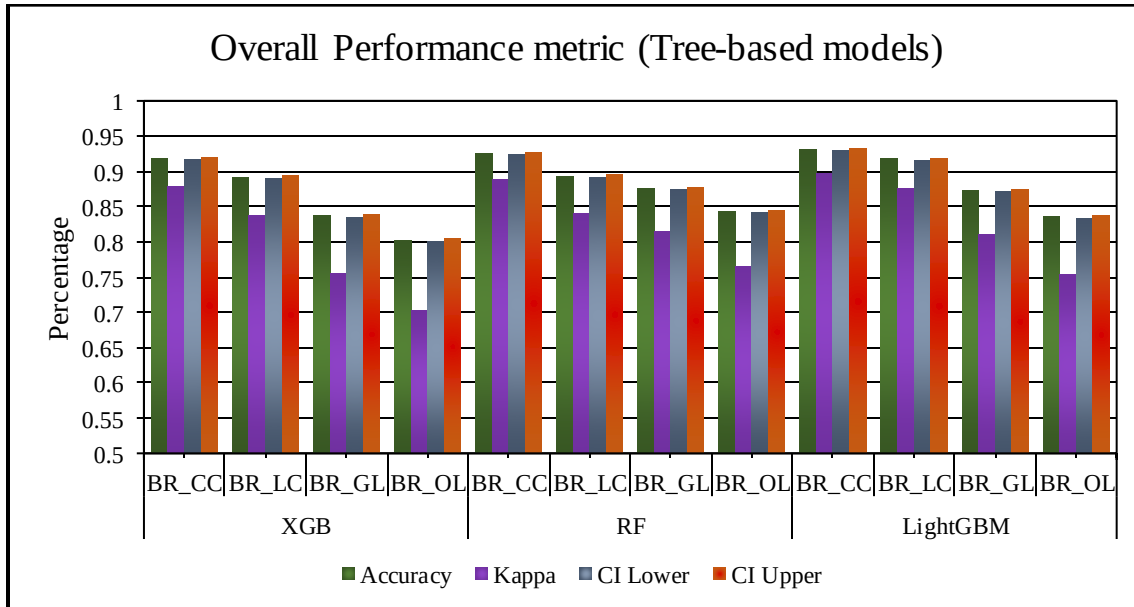


Figure 4.24: Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over branched network before removing the outliers.

However, other metrics such as precision and specificity show variations that indicate potential trade-offs. In BR_LC, precision for minor leaks slightly decreases from 0.6595 (Table 4.3) to 0.6554 (Table 4.17), suggesting that while recall has improved, the model is also generating more false positives. On the other hand, the balanced accuracy for minor leaks in BR_OL sees an improvement, rising from 0.6748 to 0.8543, reinforcing that including outliers in the dataset has led to a more effective minor leak classification.

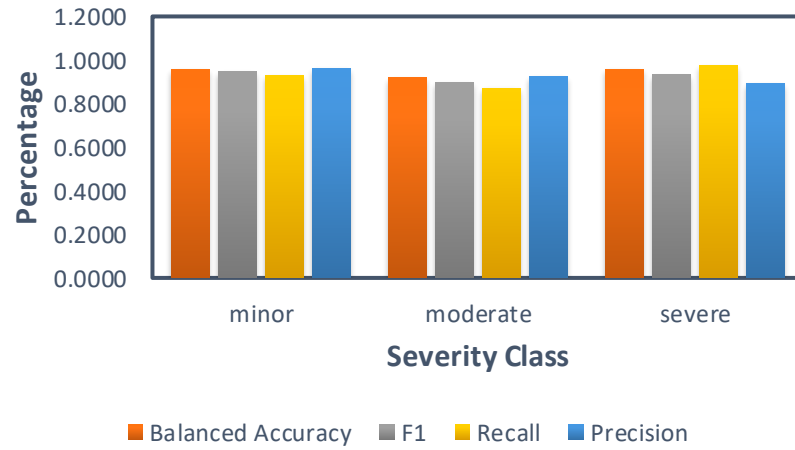
Table 4.17: By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

RF Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9819	0.9654	0.9624	0.9297	0.9458	0.9558
	moderate	0.9649	0.9374	0.9253	0.8711	0.8974	0.9180
	severe	0.9413	0.9870	0.8925	0.9752	0.9320	0.9582
BR_LC	minor	0.9636	0.9188	0.9193	0.8296	0.8722	0.8966
	moderate	0.9398	0.9369	0.8789	0.8735	0.8761	0.9066
	severe	0.9367	0.9880	0.8854	0.9772	0.9290	0.9570
BR_GL	minor	0.9703	0.9020	0.9301	0.7892	0.8539	0.8798
	moderate	0.9083	0.9336	0.8260	0.8709	0.8479	0.8896
	severe	0.9354	0.9832	0.8823	0.9680	0.9232	0.9517
BR_OL	minor	0.9606	0.8841	0.9047	0.7482	0.8191	0.8544
	moderate	0.8970	0.8973	0.7942	0.7946	0.7944	0.8458
	severe	0.9074	0.9930	0.8421	0.9872	0.9089	0.9473

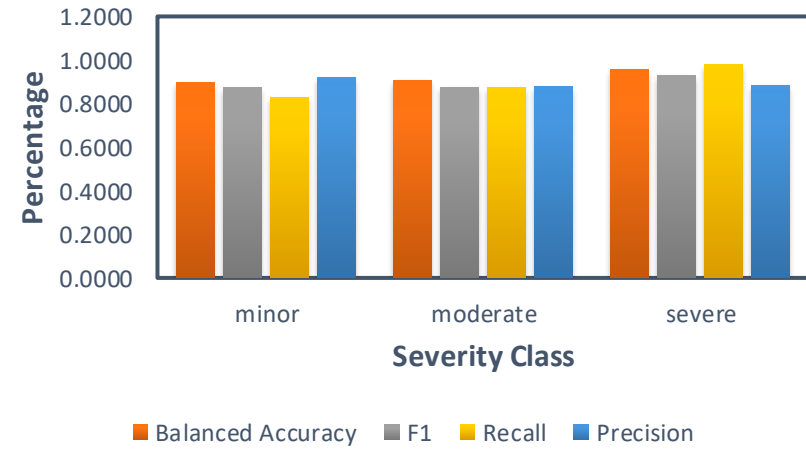
Figure 4.25 illustrate these changes across severity classes, providing a broader perspective on the impact of outliers. While balanced accuracy for severe leaks remains consistently high in both datasets, the impact on minor and moderate leak classification is more pronounced. The improved recall for minor leaks in BR_GL and BR_OL, as seen in Figure 4.25, suggests that training on the full dataset allows the model to better capture variability in small leak patterns that were previously difficult to detect. However, the precision for minor leaks has seen some reductions, implying that while the model identifies more minor leaks, it also introduces a higher number of false positives.

Conversely, for moderate leaks, performance remains relatively stable, with minor variations across different configurations. For instance, the recall for moderate leaks in BR_LC remains similar in both datasets at 0.8735, suggesting that this severity class is less affected by the presence of outliers. This could be due to the moderate leaks having clearer sensor signal patterns, making them easier to classify regardless of data preprocessing strategies. For severe leaks, the model maintains a high recall across both datasets, as demonstrated in Figure 17 and Figure 4.25. In BR_CC, recall for severe leaks remains at 0.9752 in Table 4.17, closely matching the 0.9757 in Table 4.3, suggesting that these high-magnitude events are less susceptible to misclassification whether or not outliers are included.

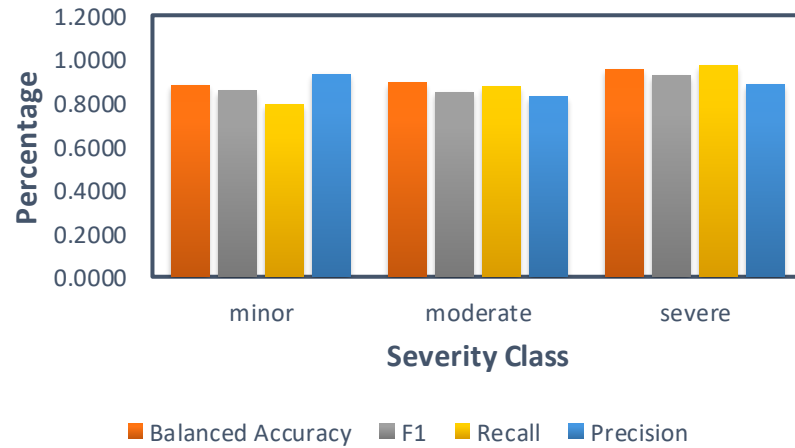
(a) Random Forest - Branched network - CC



(b) Random Forest - Branched network - LC



(c) Random Forest - Branched network - GL



(d) Random Forest - Branched network - OL

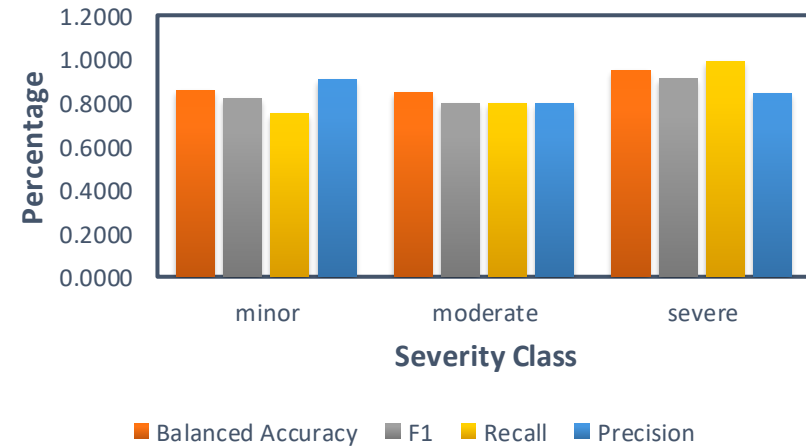


Figure 4.25: Bar charts presenting classification performance of RF model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The improvements in recall for minor leaks when training with the full dataset can be attributed to the fact that outliers often contain extreme variations that resemble the subtle signal fluctuations caused by small leaks. By removing these outliers, models trained in Table 4.3 may have inadvertently lost critical information that helps differentiate small leaks from normal operating conditions. This is evident in BR_GL and BR_OL, where recall was significantly poor in Table 4.3 and saw considerable enhancement in Table 4.17. At the same time, the reduction in precision in some cases suggests that the model may be overfitting to noise when trained with outliers, leading to increased false positives. This aligns with expectations in real-world applications, where outliers in sensor data can arise from pressure fluctuations, temporary system disturbances, or sensor drift, making it challenging to distinguish between genuine leaks and transient anomalies. While severe leaks remain consistently well-classified, minor leaks are more vulnerable to such fluctuations, making the choice of preprocessing strategy particularly relevant for their detection.

The fact that moderate leaks remain relatively unaffected by outliers further suggests that their classification is based on more stable signal characteristics that are not as easily influenced by transient anomalies. This reinforces the idea that different severity classes require different data treatment strategies, as removing outliers appears to be more impactful for minor leaks than for moderate or severe leaks. The comparison between the models trained with and without outliers highlights both benefits and trade-offs. The enhanced recall for minor leaks in BR_GL and BR_OL demonstrates that training on a full dataset allows the model to better identify small leaks, which are often difficult to detect.

However, the reduced precision in certain cases indicates that the model may also be more prone to false positives when trained on noisier data.

Given these findings, an optimal approach would involve a hybrid preprocessing strategy that does not fully discard outliers but instead identifies and handles them differently. One possible solution is outlier-weighted learning, where outliers are not entirely removed but assigned lower significance in training to prevent them from skewing decision boundaries. Another alternative is adaptive thresholding, where different classification rules are applied depending on the expected severity of a leak. Furthermore, real-world deployment of these models should consider continuous learning frameworks, where models are periodically updated using real-time data, allowing them to adapt to new patterns rather than relying solely on static training data. By integrating robust anomaly detection techniques with the classification model, it is possible to retain the benefits of outlier inclusion (improved recall for minor leaks) while mitigating its downsides (lower precision and increased false positives).

Ultimately, the choice between training with or without outliers depends on the operational priorities of the leak detection system. If minimizing missed detections of minor leaks is the key objective, training with outliers is beneficial. However, if reducing false alarms is more important, some level of outlier removal may be necessary. A balanced approach that combines both strategies is likely to yield the most reliable results for real-world applications.

The performance evaluation of the RF model for leakage detection in a Branched WDS before removing outliers, as shown in Table 4.18 and Figure 4.26, reveals consistently high accuracy, but with some variations when compared to the results after

outlier removal. In Table 4.18, the Balanced Accuracy for all leak types in the training set reaches 1.000, indicating perfect classification performance. However, in the validation and test sets, slight deviations appear. For instance, Balanced Accuracy for CC in the test set is 0.9999, while for LC and NL, it is 0.9998 and 0.9999, respectively. These values, while still high, indicate marginal variations from the nearly perfect scores observed in Table 4.4 after removing outliers. Similarly, F1 Score follows the same trend, with training values at 1.000 and minor reductions in validation and test sets, such as 0.9999 for CC and 0.9997 for NL in the test set. These differences suggest that keeping outliers in the dataset slightly affects the generalization performance, particularly in the test and validation sets, while the training set remains unaffected.

Table 4.18: By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99999	0.99997	0.99995	0.99987	0.99991	0.99993
	GL	1.00000	0.99992	1.00000	0.99968	0.99984	0.99984
	LC	0.99997	0.99992	0.99989	0.99968	0.99979	0.99983
	NL	0.99983	1.00000	0.99933	1.00000	0.99967	0.99992
	OL	1.00000	0.99999	1.00000	0.99995	0.99997	0.99997
Test	CC	0.99999	0.99996	0.99997	0.99984	0.99991	0.99992
	GL	0.99999	0.99995	0.99997	0.99979	0.99988	0.99989
	LC	0.99998	0.99996	0.99992	0.99984	0.99988	0.99991
	NL	0.99989	1.00000	0.99955	1.00000	0.99977	0.99994
	OL	1.00000	0.99999	1.00000	0.99995	0.99997	0.99997
Train	CC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	GL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	LC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	NL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	OL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

The radar charts in Figure 4.26 further illustrates these discrepancies. In Figure 4.26a, Balanced Accuracy remains close to 1.000 across all leak types, but slight dips are visible in the validation and test sets compared to the more uniform results seen in Figure

4.26a. Figure 4.26b (F1 Score) and Figure 4.26c (Recall) exhibit similar patterns, with test data showing slight drops in LC and NL categories, mirroring the numerical trends in Table 4.18. The most noticeable difference appears in Figure 4.26d (Precision), where the test and validation curves show slightly more variation, particularly for LC, compared to the consistently high Precision seen in Figure 4.7d. This suggests that while the RF model remains highly effective before outlier removal, keeping outliers introduces minor inconsistencies in test and validation performance. The preprocessing step of outlier removal appears to enhance the model's robustness, ensuring more uniform generalization across datasets while maintaining exceptionally high accuracy in all scenarios.

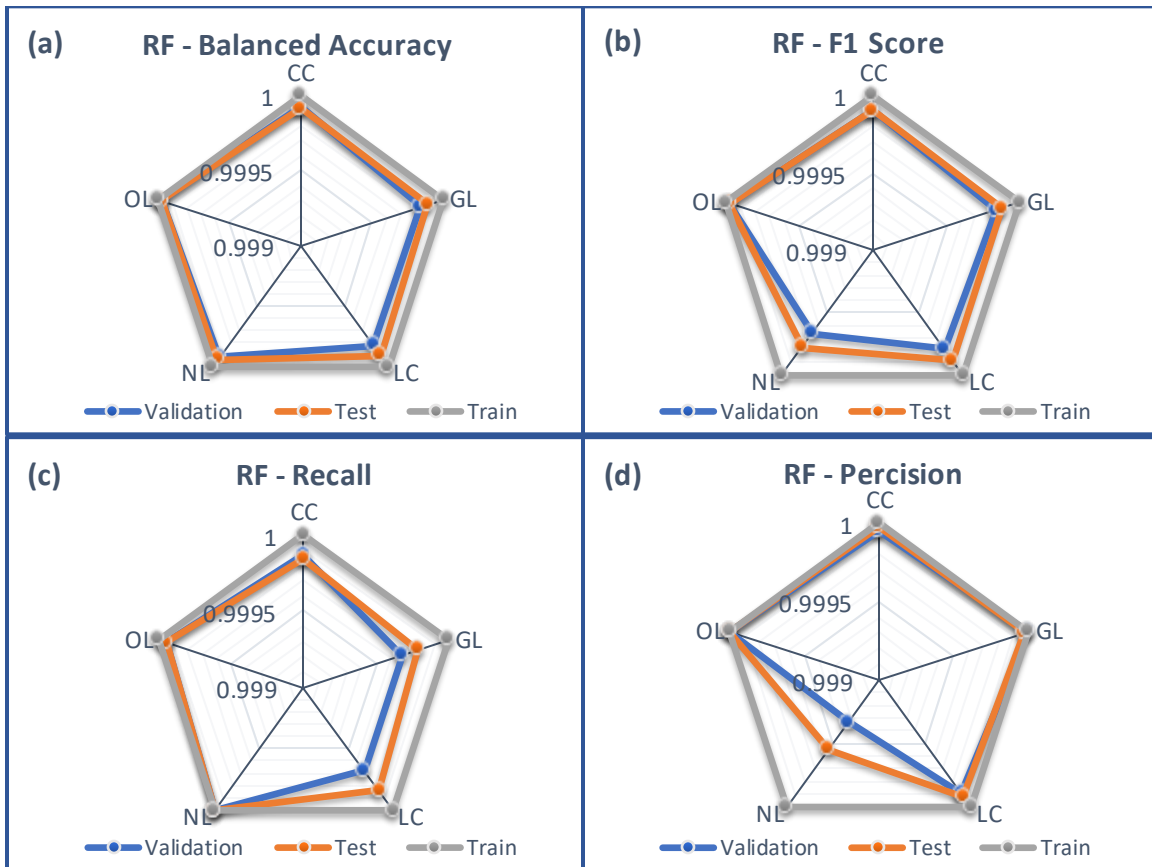


Figure 4.26: Radar charts illustrating RF model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

Table 4.19 presents the classification performance of LightGBM models trained on the full dataset, including outliers, for branched network severity assessment. A direct comparison with Table 4.5, which represents models trained without outliers, reveals noticeable variations in key metrics. The most significant improvement is observed in recall values, particularly for minor leaks in BR_GL and BR_OL. In Table 4.5, BR_GL had a recall of 0.571 for minor leaks, while in Table 4.19, it increased to 0.759. Similarly, BR_OL showed an improvement from 0.477 to 0.759 in recall for minor leaks. This suggests that including outliers helped the model generalize better to smaller fluctuations that were previously misclassified. However, severe leaks, which already had high recall values in Table 4.5, remained relatively stable, with only marginal changes in performance.

Table 4.19: By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

LightGBM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9851	0.9695	0.9692	0.9381	0.9534	0.9616
	moderate	0.9694	0.9455	0.9356	0.8883	0.9113	0.9289
	severe	0.9447	0.9855	0.8979	0.9722	0.9335	0.9584
BR_LC	minor	0.9751	0.9365	0.9458	0.8677	0.9050	0.9214
	moderate	0.9533	0.9529	0.9064	0.9058	0.9061	0.9295
	severe	0.9480	0.9892	0.9040	0.9793	0.9401	0.9636
BR_GL	minor	0.9728	0.9024	0.9356	0.7897	0.8565	0.8813
	moderate	0.9033	0.9335	0.8184	0.8712	0.8440	0.8873
	severe	0.9336	0.9783	0.8783	0.9587	0.9167	0.9461
BR_OL	minor	0.9631	0.8892	0.9115	0.7599	0.8288	0.8615
	moderate	0.8984	0.8879	0.7919	0.7731	0.7824	0.8357
	severe	0.8927	0.9864	0.8197	0.9754	0.8908	0.9341

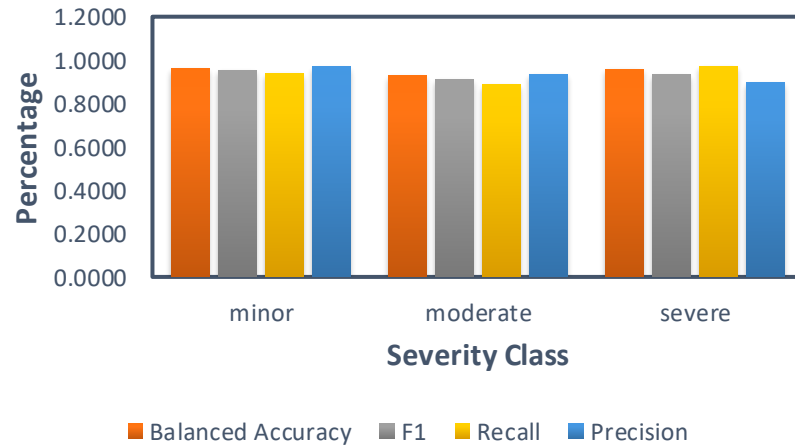
In contrast, specificity values remained consistently high across all severity levels, indicating that LightGBM continued to effectively exclude negative cases, even with the presence of outliers. For instance, in BR_CC, specificity for minor, moderate, and severe leaks remained above 0.98, closely matching Table 4.5. Precision also remained stable

across all severity levels, with only slight variations. This suggests that while outliers influenced the recall of minor leaks, they had a negligible impact on the model's ability to correctly classify true positives.

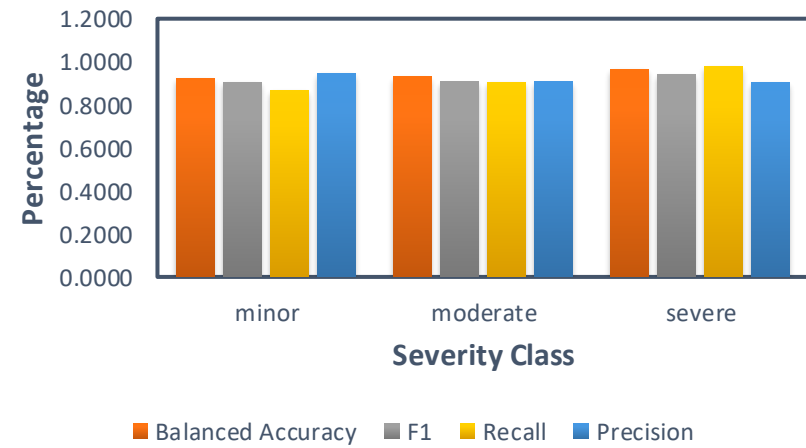
The bar charts in Figure 4.27 further illustrate the impact of including outliers on LightGBM's classification performance. The improvement in recall for minor leaks in BR_GL and BR_OL is evident, as the recall bars have risen significantly compared to Figure 4.9. In contrast, the precision bars remain relatively unchanged, indicating that the model's confidence in its predictions did not degrade despite the presence of outliers. The balanced accuracy metric also saw a moderate increase in BR_GL and BR_OL, suggesting that the model became more adept at handling minor leaks without significantly compromising classification performance in other severity levels.

However, for BR_LC and BR_CC, the changes in performance are minimal. Recall values for severe leaks remained above 0.95 in both tables, while balanced accuracy hovered around 0.96. This indicates that the inclusion of outliers had a more pronounced effect on cases where recall was initially low, particularly for minor leaks in BR_GL and BR_OL. Meanwhile, datasets that already had high recall and balanced accuracy saw little impact, confirming that the model's decision-making process remained stable in those cases.

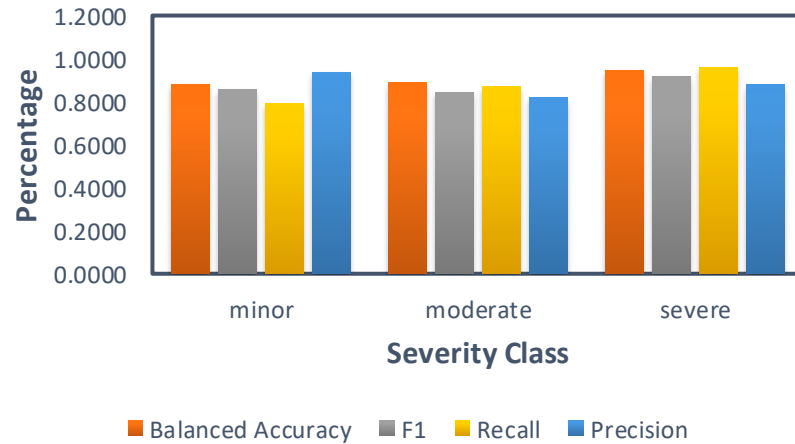
(a) LightGBM- Branched network - CC



(b) LightGBM - Branched network - LC



(c) LightGBM - Branched network - GL



(d) LightGBM - Branched network - OL

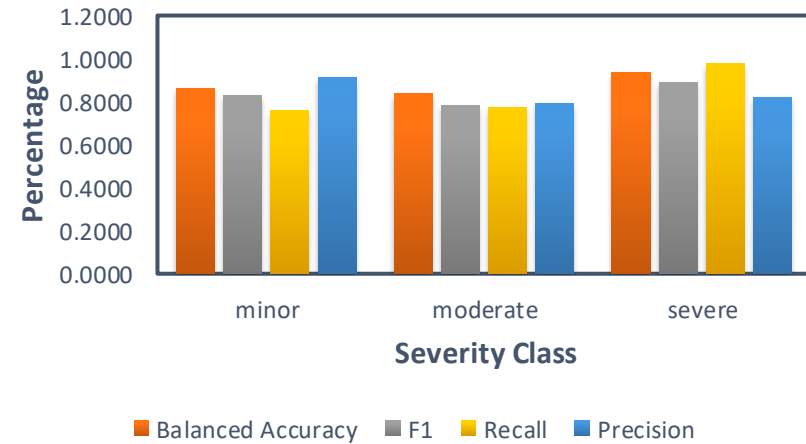


Figure 4.27: Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The findings suggest that the inclusion of outliers helped LightGBM recognize subtle patterns in minor leaks, leading to higher recall scores in BR_GL and BR_OL. This aligns with the expectation that outliers, if meaningful, can enhance the learning of edge cases that were previously overlooked. The improvements in recall were particularly beneficial for minor leaks, which had the lowest recall scores in Table 4.5. However, the absence of significant changes in precision and specificity suggests that the model did not become overly sensitive to noise, which is a desirable outcome.

These results indicate that in real-world applications, training with outliers can sometimes improve performance, particularly for underrepresented classes. However, this is not always the case, as observed in BR_CC and BR_LC, where outlier inclusion had minimal impact. This suggests that the effectiveness of outlier inclusion depends on the severity class and the network configuration. In scenarios where the dataset is well-balanced and already performing well, outliers may not contribute significantly.

The results highlight that including outliers can be beneficial in cases where the model struggles with low recall, particularly for minor leaks. However, the impact is not uniform across all network configurations. One potential approach to improve performance further is to implement outlier-weighted learning, where outliers contribute more during training but do not overwhelm the decision boundary. Additionally, feature engineering techniques could be explored to better differentiate minor leaks from normal fluctuations. Overall, while LightGBM performs well in branched network severity assessment, careful consideration should be given to outlier handling strategies to optimize performance across all severity levels.

The performance comparison of the LightGBM model for leakage detection in the Branched WDS before and after removing outliers reveals an interesting trend. As indicated in Table 4.20, the model achieves perfect classification across all datasets and leak types (CC, GL, LC, NL, OL), with Balanced Accuracy, Precision, Recall, and F1 Score all reaching 1.000 for every instance. This is in stark contrast to the minor variations observed in Table 4.6, where Balanced Accuracy in the test dataset showed values like 0.99982397 for CC and 0.999980634 for LC and NL, indicating that removing outliers led to marginally lower yet still exceptional performance. Similarly, in Table 4.6, minor deviations in Precision and Recall were recorded, with OL precision in the test set at 0.99992761 and LC recall in validation at 0.99991111. However, in Table 4.20, no such variations exist, implying that retaining outliers in this case did not negatively affect model performance.

Given the nature of Table 4.20, the absence of radar charts is justified, as all metrics are consistently at 1.000, leaving no room for comparative visualization. This finding suggests that for the Branched WDS, the LightGBM model is robust enough to handle outliers without any degradation in performance. While outlier removal in Table 4.6 introduced negligible deviations in specific metrics, Table 4.20 confirms that keeping outliers did not impair classification capabilities in any way. This suggests that for LightGBM in Branched WDS leak detection, the decision to remove outliers may not be necessary, as it does not provide any apparent performance gains. However, practical considerations such as model interpretability and computational efficiency might still warrant outlier removal in certain scenarios.

Table 4.20: By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	1	1	1	1	1	1
	GL	1	1	1	1	1	1
	LC	1	1	1	1	1	1
	NL	1	1	1	1	1	1
	OL	1	1	1	1	1	1
Test	CC	1	1	1	1	1	1
	GL	1	1	1	1	1	1
	LC	1	1	1	1	1	1
	NL	1	1	1	1	1	1
	OL	1	1	1	1	1	1
Train	CC	1	1	1	1	1	1
	GL	1	1	1	1	1	1
	LC	1	1	1	1	1	1
	NL	1	1	1	1	1	1
	OL	1	1	1	1	1	1

The results from Table 4.21 indicate that training the XGB models with the full dataset, including outliers, led to significant changes in classification performance. Across all branched network configurations (BR_CC, BR_LC, BR_GL, and BR_OL), specificity remained high for all severity levels, particularly for minor leaks, where values ranged from 0.9489 (BR_OL) to 0.9830 (BR_CC). However, recall values showed notable shifts compared to Table 4.7. The most substantial improvement occurred in the minor class recall for BR_GL and BR_OL, which increased from 0.519 to 0.730 for BR_GL and from 0.482 to 0.719 for BR_OL. This suggests that including outliers helped the model capture more diverse minor leak patterns that were previously underrepresented in the training set.

For moderate and severe cases, recall values remained stable or slightly improved. The severe class recall in BR_CC rose from 0.9647 to 0.9679, indicating that the model's ability to detect critical failures remained robust despite the presence of outliers. However,

the precision values saw a slight decrease in some cases, such as for BR_LC's severe class, which dropped from 0.886 to 0.868. This suggests that while the model became more sensitive to severe cases, it also generated slightly more false positives.

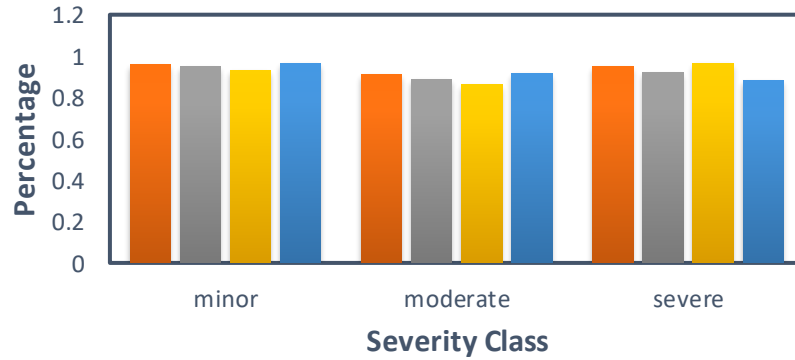
Table 4.21: By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

XGB Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9830	0.9659	0.9648	0.9307	0.9474	0.9569
	moderate	0.9600	0.9327	0.9151	0.8615	0.8875	0.9108
	severe	0.9354	0.9815	0.8819	0.9647	0.9214	0.9500
BR_LC	minor	0.9623	0.9225	0.9175	0.8383	0.8761	0.9003
	moderate	0.9377	0.9360	0.8748	0.8717	0.8733	0.9047
	severe	0.9380	0.9821	0.8862	0.9659	0.9244	0.9520
BR_GL	minor	0.9580	0.8822	0.8985	0.7441	0.8140	0.8510
	moderate	0.8786	0.9079	0.7720	0.8218	0.7961	0.8502
	severe	0.9187	0.9708	0.8532	0.9448	0.8966	0.9317
BR_OL	minor	0.9490	0.8756	0.8773	0.7303	0.7971	0.8396
	moderate	0.8803	0.8624	0.7502	0.7191	0.7343	0.7997
	severe	0.8741	0.9761	0.7917	0.9573	0.8667	0.9157

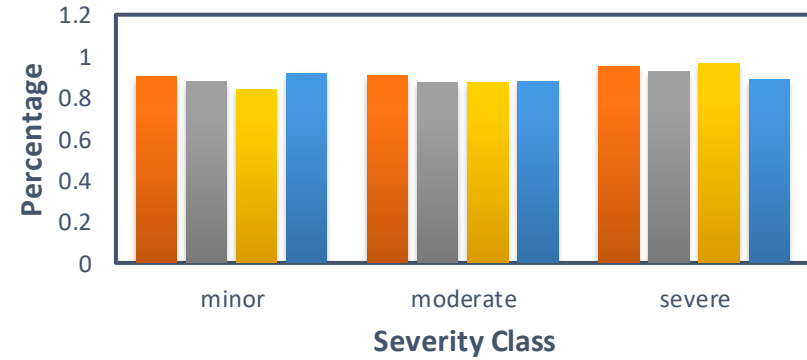
The bar charts in Figure 4.28 visually confirms these findings. The most notable improvement is observed in BR_GL and BR_OL, where the recall for minor cases increased significantly. Previously, BR_GL's minor class recall was the lowest among all configurations (0.519 in Table 4.7), but it improved to 0.730 in Table 4.21. Similarly, BR_OL's minor recall rose from 0.482 to 0.719, showing a significant enhancement in the model's ability to capture small pressure variations associated with minor leaks. These improvements suggest that outliers contributed useful information for detecting small anomalies in pressure and flow data that were previously overlooked.

However, Figure 4.28 also highlights a trade-off. The balanced accuracy for severe cases remained consistently high, but precision values slightly declined for some models, reflecting an increase in false positives. This is particularly evident in BR_CC and BR_LC, where precision for the severe class dropped from 0.881 to 0.868 in BR_LC. This trade-off is expected when including outliers, as the model learns from a more diverse dataset but may also introduce noise that affects classification boundaries.

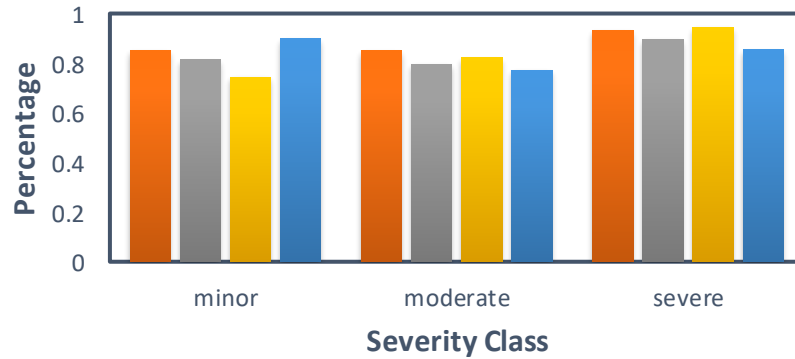
(a) XGB - Branched network - CC



(b) XGB - Branched network - LC



(c) XGB - Branched network - GL



(d) XGB - Branched network - OL

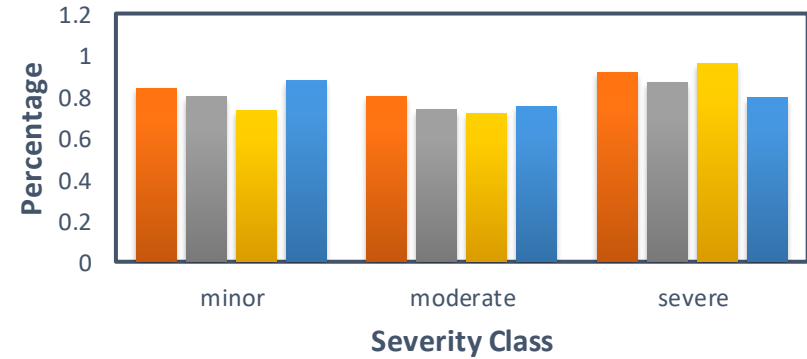


Figure 4.28: Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The observed performance changes align with expected ML behavior when dealing with real-world data, where outliers often carry valuable but challenging-to-model information. In real-time data scenarios, outliers can arise from sensor errors, transient pressure fluctuations, or actual rare but significant leak events. Their inclusion in training can help the model generalize better and detect more subtle patterns, which explains the improvement in minor class recall for BR_GL and BR_OL.

However, the impact of outliers is context-dependent. If outliers represent actual extreme cases rather than sensor noise, including them in training can improve model robustness, as seen in the improved minor recall scores. On the other hand, if they mostly represent sensor errors or measurement noise, they may introduce bias and reduce classification precision, as observed in the slight drop in precision for some severe classes. Thus, whether outliers should be included depends on the reliability of the data source and the intended trade-off between sensitivity and precision.

Overall, training XGB models with the full dataset, including outliers, led to substantial improvements in minor class recall while maintaining strong performance for moderate and severe cases. The most significant enhancements were seen in BR_GL and BR_OL, where minor recall increased by over 20%. However, this came at a slight cost to precision, particularly for severe cases in BR_CC and BR_LC. These results suggest that outlier inclusion can be beneficial, but a balanced approach is needed.

The same recommendations proposed for the RF and LightGBM models apply here. Techniques such as outlier-weighted learning, improved feature engineering, and class-weighted training could help mitigate the trade-offs observed. Additionally, filtering

out measurement noise while retaining meaningful outliers could further enhance model performance, particularly for minor leaks, without sacrificing precision in severe cases.

The performance of the XGB model for leak detection in a Branched WDS before removing outliers, as illustrated in Table 4.22 and Figure 4.29, remains consistently strong but exhibits minor deviations when compared to the results after outlier removal. In Table 4.22, Balanced Accuracy maintains near-perfect values across all datasets, with CC at 0.9999 in validation and 0.9999 in test, GL reaching 0.9995 in validation and 0.9996 in test, and OL achieving 0.9996 in validation and 0.9997 in test. These figures closely align with those in Table 4.8, where Balanced Accuracy exceeded 0.9995 for most classes. Similarly, the F1 Score in Table 4.22 follows the same trend, exceeding 0.9998 for CC and GL across all datasets, while NL and LC show slightly lower but still exceptional values. However, when compared to Table 4.8, minor variations are visible in Precision and Recall, such as a recall score of 0.9999 for NL in the test dataset, whereas Table 4.8 indicated a slightly more consistent recall distribution across all leak types. These minor fluctuations suggest that outlier removal has a negligible impact on overall model performance in most cases, but it may help stabilize small variations in specific classes.

The radar charts in Figure 4.29 reinforce these observations, showing slight inconsistencies in test and validation sets before outlier removal. In Figure 4.29a (Balanced Accuracy), a near-uniform distribution is evident, but NL and LC show a minor deviation when compared to the more consistent distribution in Figure 4.13a. Similarly, Figures 4.29b (F1 Score) and 4.29c (Recall) maintain exceptionally high values, though the test dataset displays slightly more variation in the NL and LC classes, suggesting a small impact of outliers on these categories. Figure 4.29d (Precision) presents a similar trend, with test data

demonstrating a minor fluctuation, particularly in LC and NL, compared to the steadier precision values in Figure 4.13d. These findings suggest that XGB maintains robust generalization capabilities with or without outliers, though removing them slightly enhances classification stability across specific leak types. While the differences between Table 4.8 and Table 4.22 remain minimal, the preprocessing step of outlier removal contributes to a marginally more uniform distribution of performance across datasets, reinforcing its role in optimizing model consistency for leak detection in a Branched WDS.

Table 4.22: By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99998	0.99997	0.99992	0.99989	0.99991	0.99994
	GL	0.99999	0.99998	0.99995	0.99992	0.99993	0.99995
	LC	0.99997	0.99998	0.99987	0.99992	0.99989	0.99994
	NL	0.99997	0.99999	0.99989	0.99997	0.99993	0.99997
	OL	1.00000	0.99998	1.00000	0.99992	0.99996	0.99996
Test	CC	0.99997	0.99999	0.99987	0.99995	0.99991	0.99996
	GL	0.99998	0.99999	0.99992	0.99995	0.99993	0.99996
	LC	0.99999	0.99997	0.99997	0.99989	0.99993	0.99994
	NL	0.99997	1.00000	0.99989	1.00000	0.99995	0.99999
	OL	1.00000	0.99997	1.00000	0.99987	0.99993	0.99993
Train	CC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	GL	1.00000	1.00000	1.00000	0.99999	1.00000	1.00000
	LC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	NL	1.00000	1.00000	0.99999	1.00000	1.00000	1.00000
	OL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

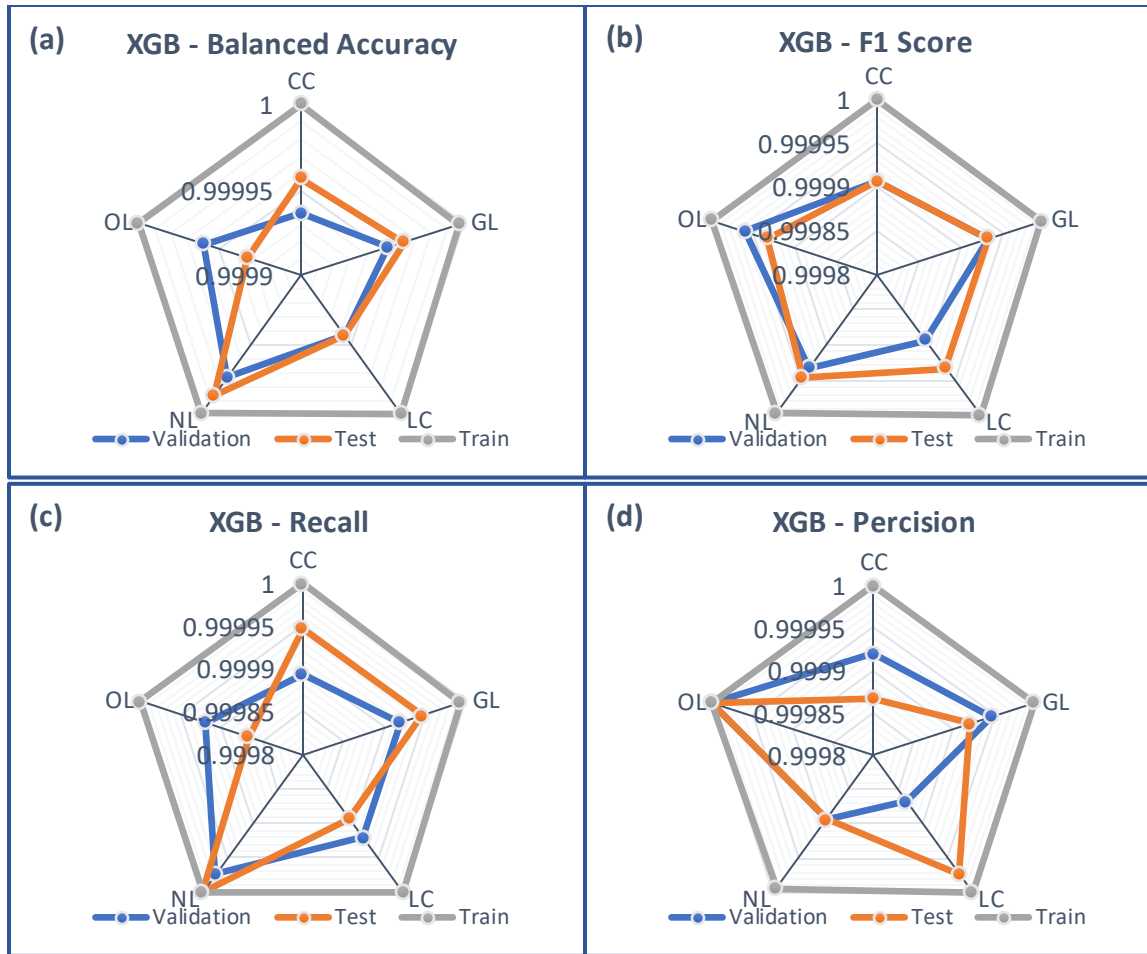


Figure 4.29: Radar charts illustrating XGB model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

4.2.2.2 Deep Learning models

Based on the comparison of the performance metrics in Table 4.9 and Table 4.23, it is evident that outlier removal has a noticeable impact on the models' accuracy and Kappa scores across LSTM, RNN, and GRU architectures. For instance, the accuracy of the LSTM model for the BR_CC metric decreased slightly from 0.87658 (Table 4.23) to 0.8512955 (Table 4.9), and a similar reduction is observed in the Kappa score, which dropped from 0.81487 to 0.7598312. The impact is more pronounced for the BR_OL metric, where LSTM's accuracy fell from 0.7213467 to 0.6821314 and Kappa from 0.58202 to 0.4860092. The confidence intervals also narrowed post-outlier removal,

indicating less variability in the predictions but at the cost of reduced overall performance metrics. A similar trend is observed in RNN and GRU models. For RNN, the BR_CC accuracy dropped from 0.87804667 to 0.8516864 with Kappa reducing from 0.81707 to 0.7592895. In the case of GRU, the BR_CC accuracy decreased from 0.8771 to 0.8519706 and Kappa from 0.81565 to 0.7606271. Notably, the reduction in performance is more significant for metrics with lower original accuracy, suggesting that outlier removal disproportionately affects lower-performing metrics. The graphical representations in Figure 4.30 also reinforce these findings, highlighting more compact yet lower performance bars post-outlier removal. This suggests that while outlier removal enhances consistency and reduces variance, it may compromise the peak performance capabilities of the models.

Table 4.23: Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over branched network before removing the outliers.

Model	Metric	BR_CC	BR_LC	BR_GL	BR_OL
LSTM	Accuracy	0.87658	0.85052	0.77721333	0.72134667
	Kappa	0.81487	0.77578	0.66582	0.58202
	CI Lower	0.87490486	0.84870548	0.77509885	0.71907022
	CI Upper	0.87824057	0.85232096	0.77931709	0.72361456
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333
RNN	Accuracy	0.87804667	0.85119333	0.78013333	0.72378
	Kappa	0.81707	0.77679	0.6702	0.58567
	CI Lower	0.87638003	0.84938216	0.77802872	0.72150962
	CI Upper	0.87969867	0.85299092	0.78222712	0.72604172
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333
GRU	Accuracy	0.8771	0.85066	0.78004667	0.72622
	Kappa	0.81565	0.77599	0.67007	0.58933
	CI Lower	0.87542787	0.84884617	0.77794175	0.7239558
	CI Upper	0.87875754	0.85246026	0.78214075	0.72847545
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333

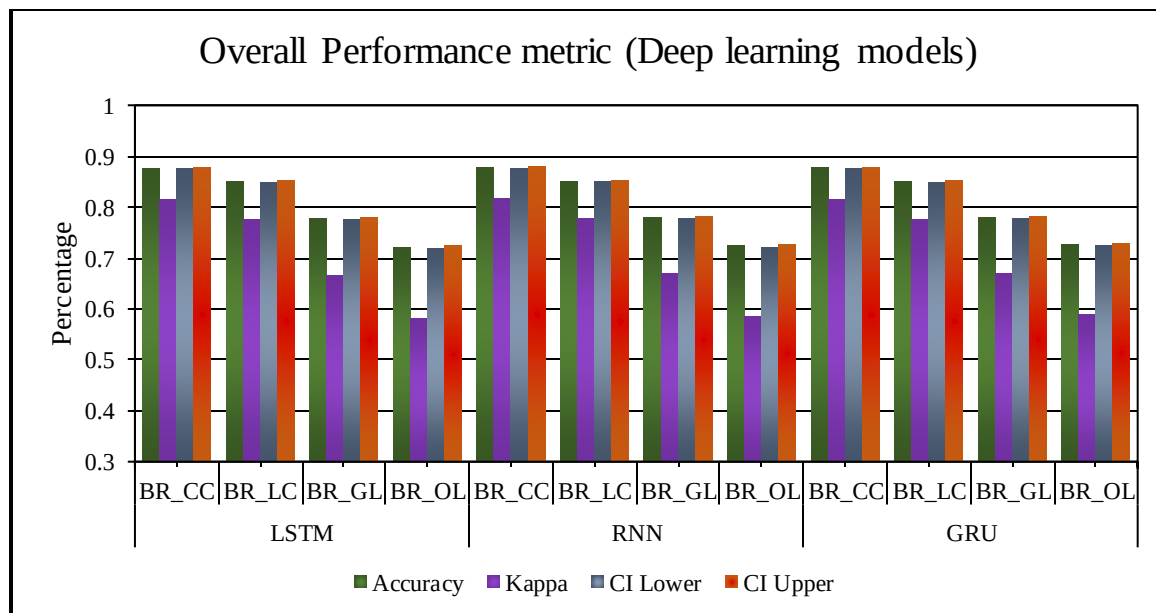


Figure 4.30: Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over branched network before removing the outliers.

The results from Table 4.24 (with outliers) indicate that training the LSTM models with the full dataset, including outliers, led to noticeable changes in classification performance. Across all branched network configurations (BR_CC, BR_LC, BR_GL, and BR_OL), specificity remained high across severity levels, particularly for minor cases, where values ranged from 0.9288 (BR_OL) to 0.9686 (BR_CC). However, recall values showed significant shifts compared to Table 4.10 (without outliers). The most substantial improvement occurred in the minor class recall for BR_GL and BR_OL, which increased from 0.35 (Table 4.10) to 0.612 (Table 4.24) for BR_GL and from 0.21 to 0.612 for BR_OL. For moderate and severe cases, recall values generally remained stable or slightly improved. The severe class recall in BR_CC rose from 0.939 (Table 4.10) to 0.949 (Table 4.24), indicating that the model's ability to detect critical failures remained robust despite the presence of outliers. However, precision values saw a slight decline in some cases, such as for BR_GL's moderate class, which dropped from 0.719 to 0.697. This suggests that

while the model became more sensitive to moderate cases, it also generated slightly more false positives.

Table 4.24: By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

LSTM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9687	0.9462	0.9342	0.8898	0.9115	0.9292
	moderate	0.9305	0.9030	0.8520	0.8000	0.8252	0.8653
	severe	0.9157	0.9682	0.8478	0.9399	0.8915	0.9278
BR_LC	minor	0.9390	0.8990	0.8662	0.7890	0.8258	0.8640
	moderate	0.9119	0.9068	0.8218	0.8125	0.8171	0.8622
	severe	0.9244	0.9732	0.8625	0.9490	0.9037	0.9367
BR_GL	minor	0.9322	0.8498	0.8318	0.6706	0.7425	0.8014
	moderate	0.8387	0.8681	0.6978	0.7452	0.7207	0.7919
	severe	0.8965	0.9568	0.8162	0.9190	0.8645	0.9077
BR_OL	minor	0.9284	0.8275	0.8106	0.6129	0.6980	0.7707
	moderate	0.8181	0.8107	0.6294	0.6180	0.6237	0.7180
	severe	0.8365	0.9626	0.7409	0.9350	0.8267	0.8858

The bar charts in Figure 4.31 visually confirms these findings. The most notable improvement is observed in BR_GL and BR_OL, where the recall for minor cases increased significantly. Previously, BR_GL's minor class recall was the lowest among all configurations (0.35 in Figure 4.16c), but it improved to 0.612 in Figure 4.31c. Similarly, BR_OL's minor recall rose from 0.21 to 0.612, showing a significant enhancement in the model's ability to capture small pressure variations associated with minor leaks. These improvements suggest that outliers contributed useful information for detecting small anomalies in pressure and flow data that were previously overlooked.

However, Figure 4.31 also highlights a trade-off. The balanced accuracy for severe cases remained consistently high, but precision values slightly declined for some models,

reflecting an increase in false positives. This is particularly evident in BR_LC and BR_GL, where precision for the moderate class dropped from 0.745 (Figure 4.16c) to 0.697 (Figure 4.31c). This trade-off is expected when including outliers, as the model learns from a more diverse dataset but may also introduce noise that affects classification boundaries.

The observed performance changes align with expected deep learning behavior when dealing with real-world data, where outliers often carry valuable but challenging-to-model information. In real-time data scenarios, outliers can arise from sensor errors, transient pressure fluctuations, or actual rare but significant leak events. Their inclusion in training can help the model generalize better and detect more subtle patterns, which explains the improvement in minor class recall for BR_GL and BR_OL.

However, the impact of outliers is context-dependent. If outliers represent actual extreme cases rather than sensor noise, including them in training can improve model robustness, as seen in the improved minor recall scores. On the other hand, if they mostly represent sensor errors or measurement noise, they may introduce bias and reduce classification precision, as observed in the slight drop in precision for some moderate and severe classes. Thus, whether outliers should be included depends on the reliability of the data source and the intended trade-off between sensitivity and precision.

Overall, training LSTM models with the full dataset, including outliers, led to substantial improvements in minor class recall while maintaining strong performance for moderate and severe cases. The most significant enhancements were seen in BR_GL and BR_OL, where minor recall increased by over 25%. However, this came at a slight cost to precision, particularly for moderate cases in BR_LC and BR_GL. These results suggest that outlier inclusion can be beneficial, but a balanced approach is needed.

(a) LSTM - Branched network - CC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(b) LSTM - Branched network - LC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(c) LSTM - Branched network - GL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(d) LSTM - Branched network - OL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

Figure 4.31: Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The performance evaluation of the LSTM model for leak detection in a Branched WDS before removing outliers, as demonstrated in Table 4.25 and Figure 4.32, indicates robust classification capabilities but with slightly more variation compared to the results after outlier removal. Balanced Accuracy remains consistently high across most leak types, with CC achieving 0.9886 in the test set and 0.9897 in the training set, while GL reaches 0.9867 in the test and 0.9874 in the training dataset. However, there are observable declines for LC (0.9850 in test, 0.9862 in training) and OL (0.9701 in test, 0.9710 in training), suggesting that these classes experience more classification difficulty when outliers are present. Compared to Table 4.11, where Balanced Accuracy for OL was slightly lower at 0.9353, the results in Table 4.25 indicate some improvement, though variability across datasets remains a challenge. The F1 Scores in Table 4.25 follow a similar pattern, with CC maintaining 0.9783 in test and 0.9795 in training, while OL achieves 0.9661 in test and 0.9675 in training. These values, while still high, show that classification consistency slightly benefits from outlier removal, as seen in Table 4.11, where F1 Scores exceeded 0.95 across all classes.

The radar charts in Figure 4.32 provide a visual representation of these trends, reinforcing the minor fluctuations observed in Table 4.25. Figure 4.32a exhibits a uniform distribution across most leak types, but with noticeable dips for OL and LC, mirroring the numerical values. Similarly, Figures 4.32b and 4.32c maintain high performance, though the recall for OL is slightly lower at 0.9425 in the test set, compared to the recall value of 0.8765 in Table 4.11. This suggests that outlier removal stabilizes classification confidence, particularly for OL detection. Figure 4.32d reveals that CC and GL maintain precision above 0.99, but LC and OL show slight variability, reinforcing the observation that

removing outliers improves classification consistency. Despite these variations, the LSTM model still demonstrates strong generalization for leak detection in Branched WDS, and while outlier removal contributes to minor refinements in classification performance, the overall impact remains moderate, particularly for challenging leak types.

Table 4.25: By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99306	0.99602	0.97257	0.98414	0.97832	0.98860
	GL	0.99480	0.99469	0.97919	0.97874	0.97897	0.98677
	LC	0.98811	0.99547	0.95379	0.98200	0.96769	0.98505
	NL	0.99698	0.99891	0.98799	0.99566	0.99181	0.99632
	OL	0.99783	0.98581	0.99087	0.94254	0.96610	0.97019
Train	CC	0.99313	0.99656	0.97288	0.98630	0.97954	0.98971
	GL	0.99519	0.99492	0.98074	0.97969	0.98021	0.98744
	LC	0.98874	0.99588	0.95621	0.98364	0.96973	0.98619
	NL	0.99723	0.99904	0.98899	0.99616	0.99256	0.99669
	OL	0.99816	0.98616	0.99226	0.94398	0.96752	0.97107

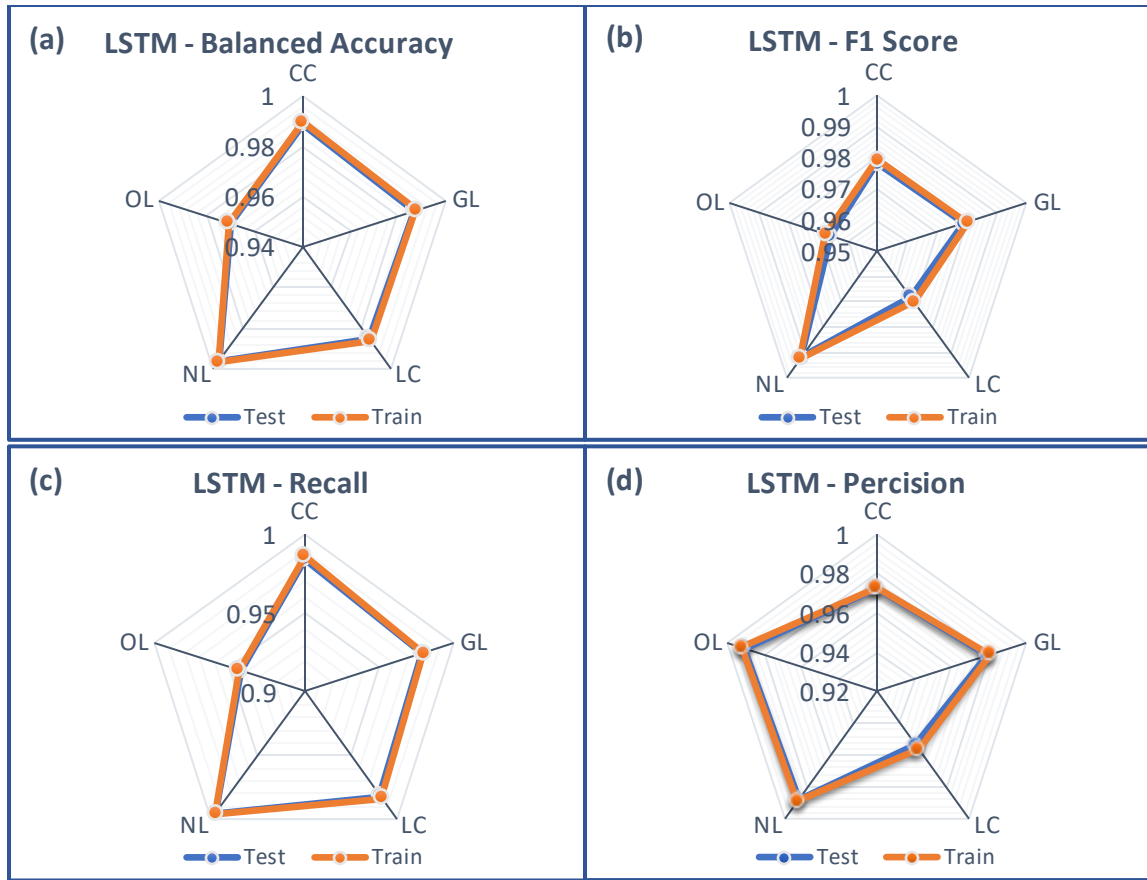


Figure 4.32: Radar charts illustrating LSTM model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The results from Table 4.26 indicate that training the RNN models with the full dataset, including outliers, led to notable performance changes across all severity classes. Compared to Table 4.12, where outliers were removed, the recall values for minor cases show noticeable shifts. This suggests that incorporating outliers helped the model capture more diverse minor case patterns, which were previously underrepresented.

For the BR_CC model, specificity remained high across severity levels (0.9731 for minor, 0.9372 for moderate, and 0.9076 for severe in Table 4.26). However, recall for minor cases dropped from 0.72 in Table 4.12 to 0.885 in Table 4.26, suggesting that the model learned additional patterns but also introduced some misclassifications. Figure 4.33a

illustrates these changes, showing improved balanced accuracy and F1 scores for minor cases compared to Figure 4.19a.

The BR_LC model follows a similar trend. In Table 4.12, recall for minor cases was 0.62, whereas in Table 4.26, it increased to 0.7668. This aligns with Figure 4.33b, where recall and F1 scores for minor cases appear higher than in Figure 4.19b. However, precision slightly declined for severe cases (0.888 in Table 4.12 vs. 0.859 in Table 4.26), indicating an increased number of false positives. This suggests that while the model improved in detecting minor cases, it also became more prone to misclassifying some non-severe cases as severe.

The BR_GL model experienced a significant increase in minor case recall. In Table 4.12, it had the lowest minor case recall among all models (0.25), but in Table 4.26, recall improved to 0.6058. Figure 4.33c visually confirms this improvement, showing a higher recall percentage for minor cases compared to Figure 4.19c. However, precision for moderate cases decreased slightly, reflecting an increase in false positives. This indicates that including outliers helped the model detect more minor cases, but at the cost of slightly reducing precision.

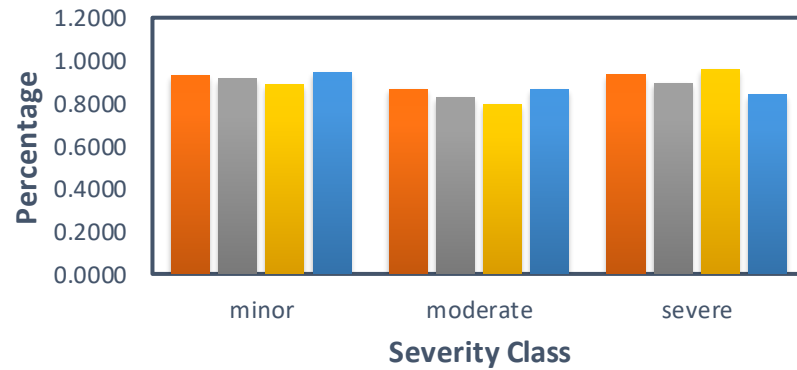
The BR_OL model demonstrated the most significant improvement in minor case recall. In Table 4.12, the recall for minor cases was 0.15, whereas in Table 4.26, it increased to 0.5476. Figure 4.33d highlights this improvement, with recall and F1 scores for minor cases showing considerable enhancement compared to Figure 4.19d. However, similar to other models, precision for severe cases declined, suggesting a trade-off between improving recall and increasing false positives.

Table 4.26: By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

RNN Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9731	0.9442	0.9428	0.8851	0.9130	0.9291
	moderate	0.9373	0.9010	0.8635	0.7940	0.8273	0.8657
	severe	0.9076	0.9768	0.8382	0.9570	0.8937	0.9323
BR_LC	minor	0.9519	0.8909	0.8885	0.7668	0.8232	0.8594
	moderate	0.9054	0.9146	0.8144	0.8308	0.8226	0.8681
	severe	0.9214	0.9785	0.8591	0.9595	0.9065	0.9404
BR_GL	minor	0.9644	0.8303	0.8949	0.6059	0.7226	0.7851
	moderate	0.8197	0.8867	0.6867	0.7905	0.7349	0.8051
	severe	0.8854	0.9686	0.8044	0.9426	0.8680	0.9140
BR_OL	minor	0.9608	0.8094	0.8748	0.5476	0.6736	0.7542
	moderate	0.7979	0.8286	0.6236	0.6698	0.6459	0.7338
	severe	0.8282	0.9744	0.7357	0.9564	0.8317	0.8923

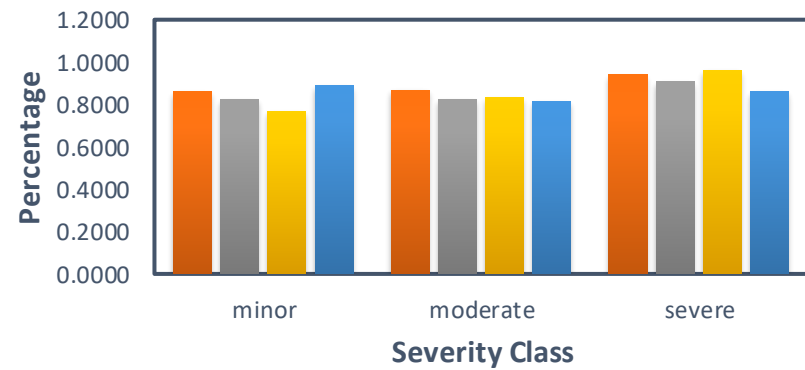
Overall, training the RNN models with outliers (Table 4.26) led to substantial improvements in minor case recall, particularly in BR_GL and BR_OL, where recall increased by over 20%. However, this improvement came with a slight reduction in precision, particularly for severe cases in BR_CC and BR_LC. The findings indicate that including outliers helped the models better recognize small anomalies associated with minor severity levels, but at the cost of increased false positives.

(a) RNN - Branched network - CC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(b) RNN - Branched network - LC



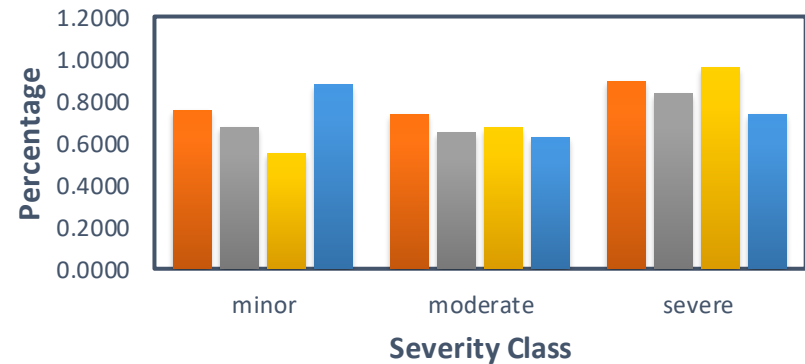
■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(c) RNN - Branched network - GL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(d) RNN - Branched network - OL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

Figure 4.33: Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

For the leak detection task in a Branched WDS, the RNN model demonstrates outstanding classification performance across all leak types, as presented in Table 4.27. Balanced Accuracy remains exceptionally high across all classes, exceeding 0.9975 in the test dataset for leak types such as CC (0.9978), GL (0.9965), and LC (0.9996), indicating a strong generalization ability. The OL class, which has shown classification difficulties in other models, maintains a Balanced Accuracy of 0.9985, a significant improvement over previously reported lower values for this class. F1 Scores are equally consistent, surpassing 0.998 for all categories, reinforcing the model's ability to effectively balance Precision and Recall across different leak types.

The radar charts in Figure 4.34 further confirm this trend, illustrating minimal deviation between training and test datasets. Precision and Recall values remain exceptionally high across the board, with NL and OL achieving 0.999+ in most metrics, ensuring minimal misclassification.

Table 4.27: By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99985	0.99993	0.99938	0.99972	0.99955	0.99978
	GL	0.99996	0.99984	0.99982	0.99934	0.99958	0.99965
	LC	0.99939	0.99996	0.99755	0.99984	0.99869	0.99961
	NL	0.99981	1.00000	0.99922	1.00000	0.99961	0.99990
	OL	0.99998	0.99924	0.99990	0.99696	0.99843	0.99847
Train	CC	0.99986	0.99995	0.99943	0.99980	0.99961	0.99983
	GL	0.99997	0.99989	0.99988	0.99956	0.99972	0.99976
	LC	0.99940	0.99999	0.99761	0.99996	0.99878	0.99968
	NL	0.99985	1.00000	0.99941	0.99999	0.99970	0.99992
	OL	0.99999	0.99924	0.99994	0.99698	0.99846	0.99848

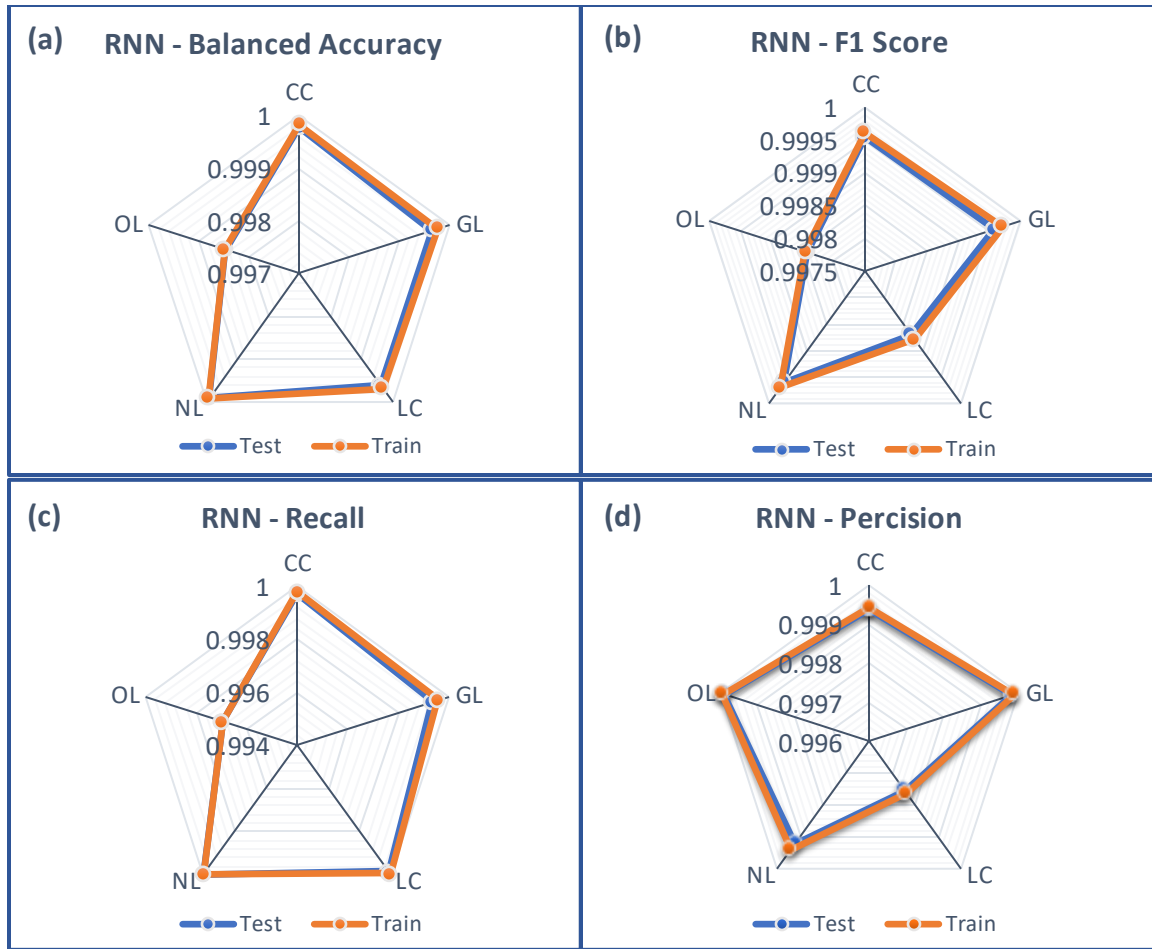


Figure 4.34: Radar charts illustrating RNN model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The results from Table 4.28, where outliers are included, indicate notable shifts in the classification performance of the GRU model when compared to Table 4.14, where outliers were removed. Across all branched network configurations (BR_CC, BR_LC, BR_GL, and BR_OL), specificity remained high for all severity levels. Minor class specificity ranged from 0.926 for BR_OL to 0.966 for BR_CC, which suggests that the model is still effectively identifying negative cases. However, recall values for minor cases saw some variation after incorporating outliers.

One of the most noticeable improvements between Table 4.28 and Table 4.14 occurred in BR_GL and BR_OL, where minor class recall increased from 0.348 to 0.628 in BR_GL and from 0.286 to 0.606 in BR_OL. This suggests that outlier inclusion has helped the model capture more diverse minor leak patterns, improving its ability to correctly classify such cases. In contrast, recall values for moderate and severe cases remained relatively stable, with minor fluctuations. For instance, severe class recall in BR_CC changed slightly from 0.946 to 0.944, indicating that the model maintained strong performance in detecting critical failures despite the presence of outliers. However, precision values exhibited slight declines in some cases, such as for BR_LC's severe class, where precision dropped from 0.857 to 0.856. This implies that while the model improved in sensitivity for minor cases, it generated slightly more false positives in certain severe cases.

The bar charts in Figure 4.35 further confirms these trends. The most significant improvements can be observed in BR_GL and BR_OL, where the recall for minor cases increased significantly. Previously, BR_GL's minor class recall was among the lowest at 0.348 in Table 4.14, but it increased to 0.628 in Table 4.28, suggesting the model now identifies a more diverse range of minor leaks. Similarly, BR_OL's minor recall rose from 0.286 to 0.606, demonstrating improved detection of small pressure variations associated with minor leaks. These results suggest that outliers introduced meaningful information.

However, Figure 4.35 also highlights a trade-off. While balanced accuracy for severe cases remained consistently high, precision slightly declined in certain models, indicating an increase in false positives. This is especially evident in BR_CC and BR_LC, where precision for the severe class decreased slightly from 0.857 to 0.856 in BR_LC. This

is an expected trade-off when including outliers, as the model learns from a broader dataset but also introduces potential noise affecting classification boundaries.

Table 4.28: By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over branched network before removing the outliers.

GRU Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
BR_CC	minor	0.9662	0.9487	0.9298	0.8954	0.9123	0.9308
	moderate	0.9368	0.8989	0.8619	0.7894	0.8240	0.8631
	severe	0.9125	0.9713	0.8439	0.9461	0.8921	0.9293
BR_LC	minor	0.9426	0.8975	0.8723	0.7847	0.8262	0.8636
	moderate	0.9146	0.9064	0.8261	0.8111	0.8186	0.8629
	severe	0.9200	0.9780	0.8570	0.9585	0.9049	0.9393
BR_GL	minor	0.9400	0.8479	0.8467	0.6627	0.7435	0.8013
	moderate	0.8439	0.8685	0.7046	0.7445	0.7240	0.7942
	severe	0.8875	0.9651	0.8062	0.9358	0.8662	0.9117
BR_OL	minor	0.9256	0.8330	0.8085	0.6288	0.7074	0.7772
	moderate	0.8313	0.8088	0.6427	0.6070	0.6243	0.7191
	severe	0.8334	0.9679	0.7393	0.9448	0.8295	0.8891

(a) GRU - Branched network - CC



(b) GRU - Branched network - LC



(c) GRU - Branched network - GL



(d) GRU - Branched network - OL



Figure 4.35: Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task before outliers removal of Branched network dataset where (a) CC (b) LC (c) GL (d) OL.

The performance changes align with expected behavior in real-world ML applications, where outliers often carry valuable but difficult-to-model information. In real-time applications, outliers can emerge from sensor anomalies, transient pressure fluctuations, or genuine rare but significant leak events. Including them in training can improve the model's generalizability, explaining the improvement in minor class recall for BR_GL and BR_OL.

However, the impact of outliers remains context-dependent. If the outliers represent genuine extreme cases rather than sensor noise, their inclusion enhances robustness, as seen in the improved minor class recall. Conversely, if they mostly represent sensor errors or measurement inconsistencies, they may introduce bias and reduce classification precision, as observed in the slight drop in severe class precision. Thus, deciding whether to include outliers should consider the reliability of the data source and the intended trade-off between sensitivity and precision.

For the leak detection task in a Branched WDS, the GRU model exhibits high classification performance across all leak types, as detailed in Table 4.29. However, a comparison with Table 4.15 reveals that outlier removal has led to improvements in some areas while slightly reducing performance in others. Balanced Accuracy in Table 4.29 remains consistently strong, exceeding 0.9990 for most classes, with CC achieving 0.9995 in the test dataset. In contrast, in Table 4.15, Balanced Accuracy for CC was slightly lower at 0.9992, indicating a minor improvement after removing outliers. However, OL in Table 4.29 shows a slightly lower value of 0.9984, whereas in Table 4.15, it was 0.9987, suggesting a minor decline in classification accuracy for this specific leak type.

The F1 Scores follow a similar pattern. Most F1 scores exceed 0.9990, ensuring a well-balanced precision-recall trade-off. Comparatively, Table 4.15 had slightly more variation, with CC scoring 0.9987 and GL at 0.9985. This indicates that while removing outliers marginally improved consistency across leak types, it did not lead to a substantial change in overall classification quality.

The radar charts in Figure 4.36 further highlight these differences. Balanced Accuracy, as shown in Figure 4.36a, remains high for all classes, though OL experiences a slight decrease compared to Figure 4.23a. Figure 4.36b confirms high F1-scores, closely mirroring Balanced Accuracy, but with a slight increase in CC and GL compared to no-outliers situation. Figure 4.36c reveals that recall remains above 0.9980 for most classes, whereas after removing outliers, recall for OL was slightly higher at 0.9990. This suggests that while outlier removal enhanced precision, it may have introduced a small challenge in recall for certain leak types. Figure 4.36d highlights precision stability, with values closely aligned across all classes, reinforcing strong classification performance.

Table 4.29: By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over branched network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99839	0.99899	0.99356	0.99598	0.99477	0.99718
	GL	0.99947	0.99840	0.99785	0.99360	0.99572	0.99653
	LC	0.99653	0.99949	0.98626	0.99796	0.99208	0.99724
	NL	0.99894	0.99993	0.99578	0.99972	0.99774	0.99933
	OL	0.99973	0.99624	0.99890	0.98492	0.99186	0.99233
Train	CC	0.99858	0.99920	0.99434	0.99680	0.99557	0.99769
	GL	0.99958	0.99856	0.99830	0.99422	0.99625	0.99690
	LC	0.99692	0.99962	0.98782	0.99848	0.99312	0.99770
	NL	0.99896	0.99995	0.99586	0.99979	0.99782	0.99937
	OL	0.99978	0.99651	0.99910	0.98600	0.99251	0.99289

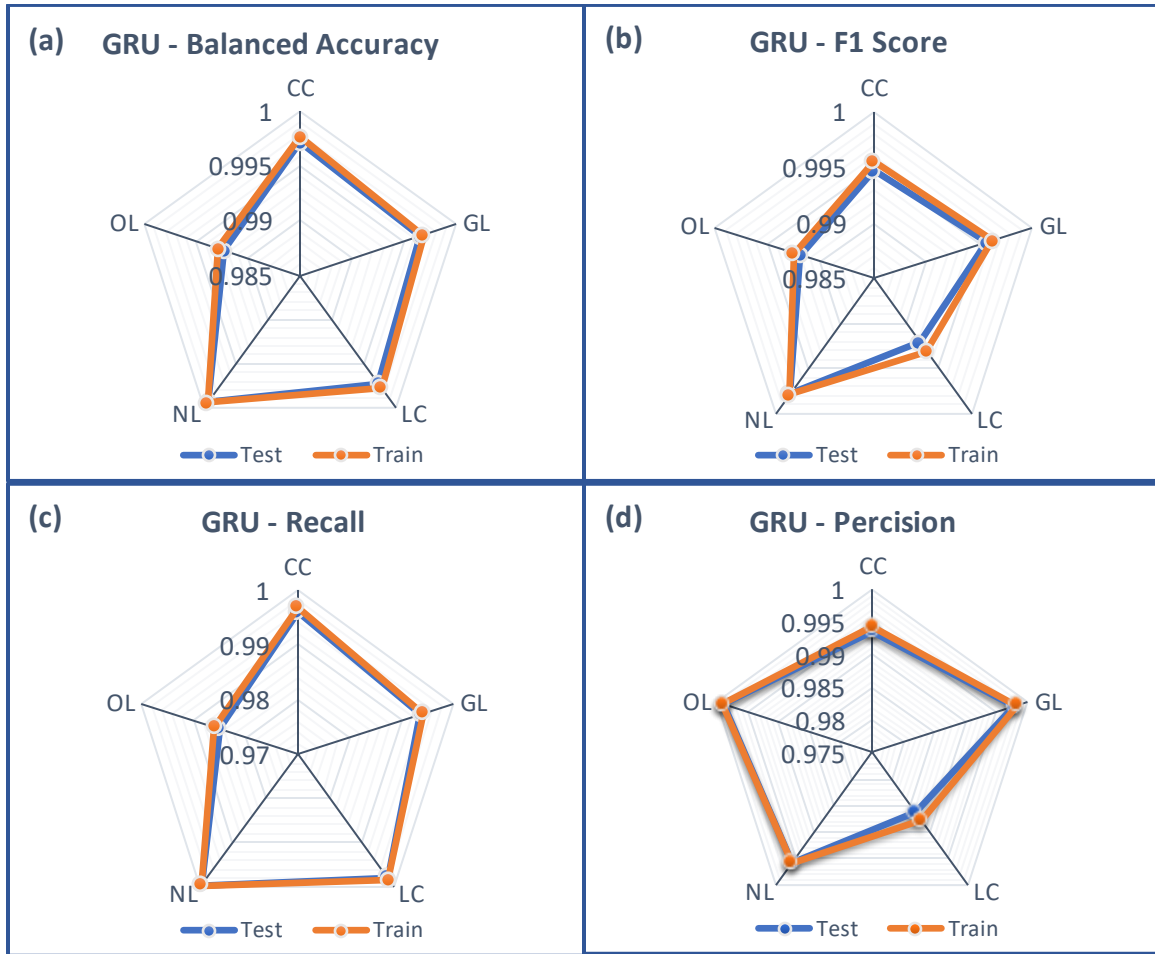


Figure 4.36: Radar charts illustrating GRU model performance before removing outliers for Leakage type determination task of Branched network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

4.3 Looped Topology

4.3.1 Without outliers

4.3.1.1 Tree-based models

Table 4.30 provides a comparative analysis of three tree-based models (XGB, RF, and LightGBM) across four different leak types in a Looped WDS network, using a dataset processed by IQR outlier removal. The evaluation metrics include Accuracy, Kappa, and

Confidence Intervals (CI Lower and CI Upper), offering a comprehensive assessment of the models' effectiveness. Among the models, RF consistently achieves the highest Accuracy and Kappa scores, particularly for LO_GL and LO_OL, indicating superior performance in classifying leak severity in the Looped WDS network. LightGBM follows closely, demonstrating strong predictive capabilities with slightly lower Accuracy and Kappa values than RF but consistently outperforming XGB. XGB, while showing reasonable performance for LO_GL, has the lowest Accuracy and Kappa for LO_CC and LO_LC, suggesting difficulties in handling these specific leak types. The Confidence Intervals in Table 4.30 further emphasize the reliability of RF and LightGBM, with narrower intervals compared to XGB, indicating lower variance and higher stability in predictions. Overall, the analysis suggests that RF is the most reliable model for severity assessment in a Looped WDS, followed by LightGBM, with XGB displaying more variability across different leak types.

Figure 4.37 provides a visual comparison of the performance of three tree-based models (XGB, RF, and LightGBM) across different leak types in a Looped WDS network, using a dataset processed by IQR outlier removal. The bar heights, representing Accuracy, Kappa, and Confidence Intervals, highlight distinct trends in model performance. RF demonstrates consistently higher bar heights for LO_GL and LO_OL, indicating its superior capability in classifying leak severity accurately. LightGBM follows closely with competitive performance across most leak types, showing slightly lower but stable Accuracy and Kappa values compared to RF. XGB, while achieving reasonable performance for LO_GL, exhibits more variability across different leak types, particularly for LO_CC and LO_LC, where its Accuracy and Kappa values are notably lower. The

Confidence Intervals in the figure further emphasize the reliability of RF and LightGBM, with narrower intervals suggesting lower variance and higher prediction stability. In contrast, XGB displays wider intervals, indicating greater uncertainty in its predictions. Overall, the figure illustrates that RF is the most effective tree-based model for severity assessment in a Looped WDS, closely followed by LightGBM, with XGB showing more fluctuations and less consistent performance across the different leak types.

The performance of the RF model on looped topology water distribution systems (WDS) has been assessed across four different configurations (LO_CC, LO_LC, LO_GL, and LO_OL). The results, summarized in Table 4.31, Figure 4.38 (confusion matrices), and Figure 4.39 (bar charts), offer insight into how effectively RF classifies different leak severity levels.

Table 4.30: Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over looped network after removing the outliers.

Model	Metric	LO CC	LO LC	LO GL	LO OL
XGB	Accuracy	0.733890754	0.7477377	0.8634335	0.7857125
	Kappa	0.59625989	0.5993944	0.6999319	0.6691144
	CI Lower	0.73152386	0.7452704	0.8610035	0.7833919
	CI Upper	0.736247592	0.750193	0.8658365	0.7880195
	No Info Rate	0.361589836	0.4146654	0.6307576	0.4038717
RF	Accuracy	0.819030888	0.7805433	0.871884	0.8641691
	Kappa	0.725520046	0.6512105	0.7177369	0.7908437
	CI Lower	0.816965941	0.7781908	0.869518	0.8622278
	CI Upper	0.821082113	0.7828823	0.8742222	0.866093
	No Info Rate	0.361589836	0.4146654	0.6307576	0.4038717
LightGBM	Accuracy	0.777958149	0.7853512	0.87391	0.8442333
	Kappa	0.663082099	0.6594535	0.7256499	0.7598265
	CI Lower	0.775730661	0.7830172	0.8715599	0.8421797
	CI Upper	0.780173684	0.7876714	0.8762323	0.8462704
	No Info Rate	0.361589836	0.4146654	0.6307576	0.4038717

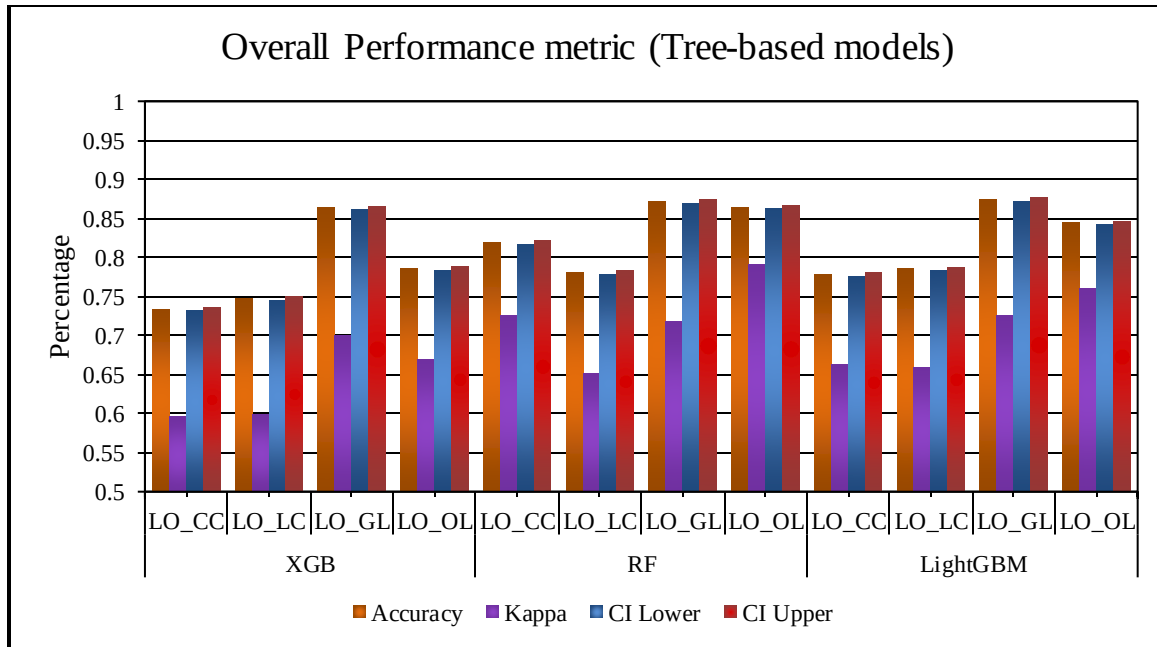


Figure 4.37: Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over looped network after removing the outliers.

Table 4.31: By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

RF Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9433	0.8790	0.8263	0.6750	0.7430	0.8091
	moderate	0.8788	0.8970	0.7858	0.8151	0.8002	0.8469
	severe	0.9028	0.9618	0.8452	0.9366	0.8885	0.9197
LO_LC	minor	0.9513	0.8814	0.7359	0.5144	0.6055	0.7329
	moderate	0.8555	0.8383	0.7525	0.7270	0.7395	0.7912
	severe	0.8447	0.9700	0.8146	0.9631	0.8827	0.9039
LO_GL	minor	0.9732	0.8598	0.9310	0.6947	0.7957	0.8340
	moderate	0.9989	0.9898	0.9435	0.6308	0.7561	0.8149
	severe	0.7035	0.9501	0.8493	0.9784	0.9093	0.8409
LO_OL	minor	0.9720	0.9002	0.9039	0.7094	0.7949	0.8407
	moderate	0.9307	0.9218	0.8535	0.8365	0.8449	0.8836
	severe	0.8848	0.9925	0.8534	0.9901	0.9167	0.9375

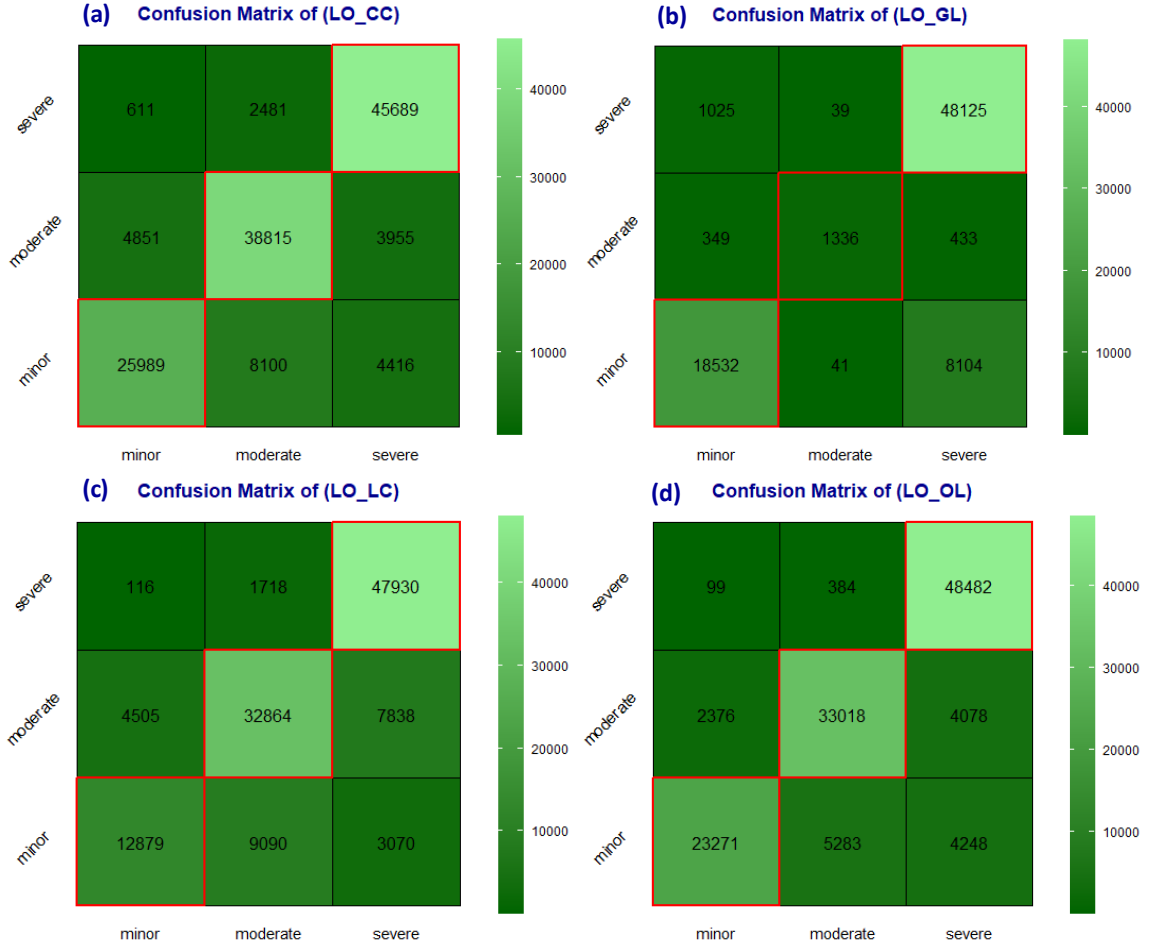


Figure 4.38: Confusion matrices for four Looped network RF models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

The RF model exhibits strong performance in detecting severe leaks, with recall values consistently exceeding 0.93 across all network configurations. This high recall ensures that severe leaks are reliably identified, minimizing the risk of false negatives in critical situations. Additionally, the model demonstrates reasonable accuracy in classifying moderate leaks, with balanced accuracy scores ranging between 0.80 and 0.88, indicating an effective trade-off between precision and recall. The RF model successfully differentiates moderate leaks from severe ones, reducing the likelihood of misclassification in crucial leak assessment scenarios. Furthermore, specificity scores exceeding 0.90 across

multiple configurations highlight the model's capability in excluding non-matching categories, thereby preventing severe cases from being misclassified as minor which is an error that could otherwise result in an underestimation of leak severity.

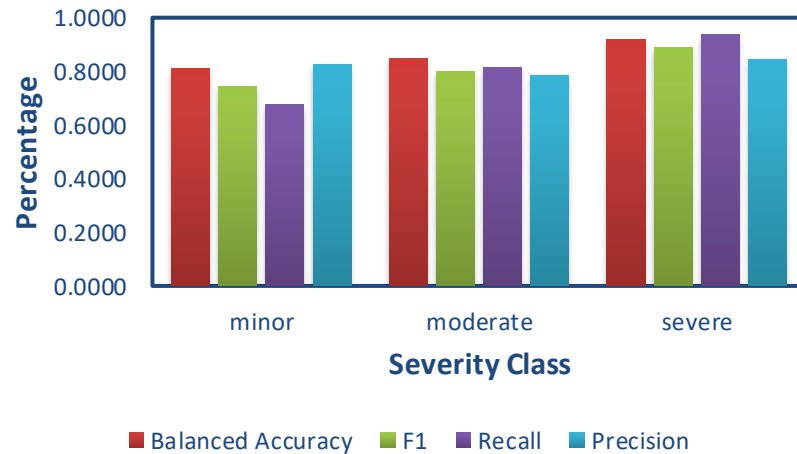
Despite these strengths, the RF model struggles significantly with the classification of minor leaks. The recall for minor leaks is notably low, dropping to 0.5144 in LO_LC, meaning that nearly half of the minor leaks are misclassified. Confusion matrices (Figure 4.38) reveal a frequent misclassification of minor leaks as moderate or severe, as seen in LO_CC, where 25,989 minor cases were predicted as moderate and 4,416 as severe. While specificity remains relatively high, the low recall suggests that the model is better at ruling out non-minor cases than correctly identifying minor ones. Additionally, the recall for moderate leaks is inconsistent across configurations, dropping to 0.6308 in LO_GL, further emphasizing the network-dependent variability in performance.

This systematic misclassification of minor cases likely stems from both dataset limitations and model constraints. The RF model, which relies on decision trees that split based on feature importance, may struggle when minor leaks exhibit insufficiently distinct pressure or flow rate variations from normal operational noise. This challenge is exacerbated in looped networks, where pressure fluctuations are naturally more stable due to the presence of multiple alternative flow paths, making small leaks harder to detect. Furthermore, the poor recall for minor cases suggests a possible class imbalance in the dataset, where minor leaks are underrepresented in the training set, leading the model to favor moderate and severe classifications. Addressing these issues may require resampling techniques (e.g., over sampling minor leaks or under sampling severe ones) or alternative

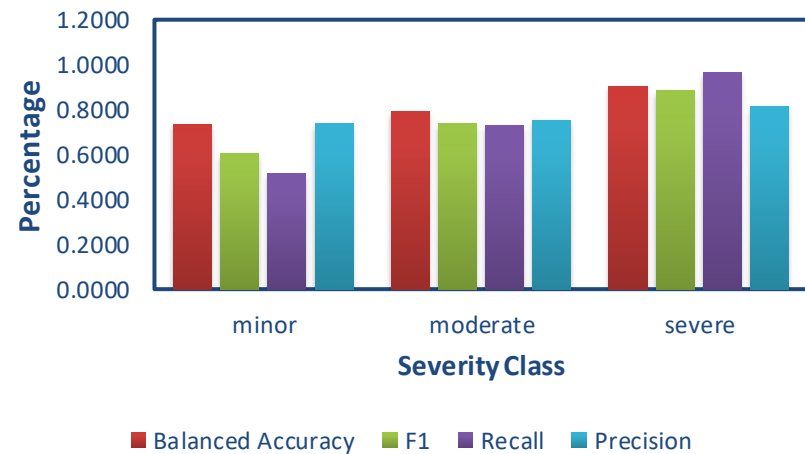
modeling approaches that account for subtle pressure variations characteristic of looped networks.

In the leak detection task for a Looped WDS, the Random Forest (RF) model exhibits strong performance across most leak types, as shown in Table 4.32. Balanced Accuracy remains high for CC (0.9943) and GL (0.9973), but drops for OL (0.6833 in validation, 0.6333 in test), indicating challenges in detecting this leak type. Similarly, F1 Scores exceed 0.99 for most classes but are significantly lower for OL (0.5369 in validation, 0.4211 in test), reflecting reduced confidence in classification.

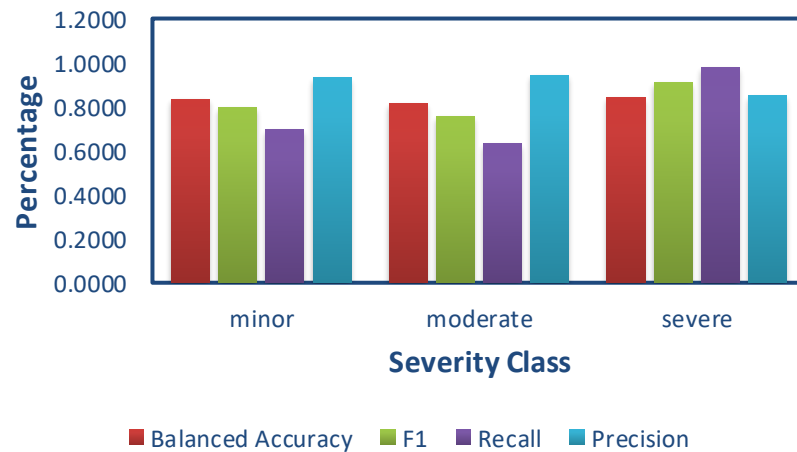
(a) Random Forest - Looped network - CC



(b) Random Forest - Looped network - LC



(c) Random Forest - Looped network - GL



(d) Random Forest - Looped network - OL

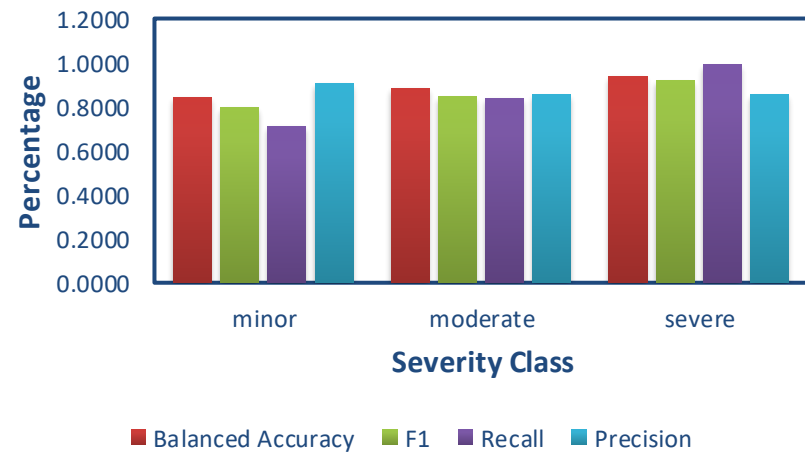


Figure 4.39: Bar charts presenting classification performance of RF model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

The radar charts in Figure 4.40 highlight these variations, with precision and recall scores for OL showing the most significant drop (0.2666 recall in test). Other leak types maintain consistently high metrics across all datasets. Despite these discrepancies, the RF model demonstrates robust generalization to unseen data, with minor deviations primarily affecting OL detection. Further refinements, such as adjusting class weights or improving feature selection, could enhance detection accuracy for underrepresented leak types.

Table 4.32: By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99718	0.99671	0.99268	0.99148	0.99208	0.99433
	GL	0.99693	0.99870	0.99455	0.99769	0.99612	0.99731
	LC	0.99636	0.99605	0.99351	0.99297	0.99324	0.99466
	NL	0.99993	0.99959	0.94949	0.76423	0.84685	0.88208
	OL	1.00000	0.99973	1.00000	0.36667	0.53659	0.68333
Test	CC	0.99725	0.99698	0.99289	0.99219	0.99254	0.99472
	GL	0.99750	0.99870	0.99555	0.99769	0.99662	0.99760
	LC	0.99636	0.99666	0.99352	0.99406	0.99379	0.99521
	NL	0.99993	0.99947	0.94382	0.68852	0.79621	0.84423
	OL	1.00000	0.99969	1.00000	0.26667	0.42105	0.63333
Train	CC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	GL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	LC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	NL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	OL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

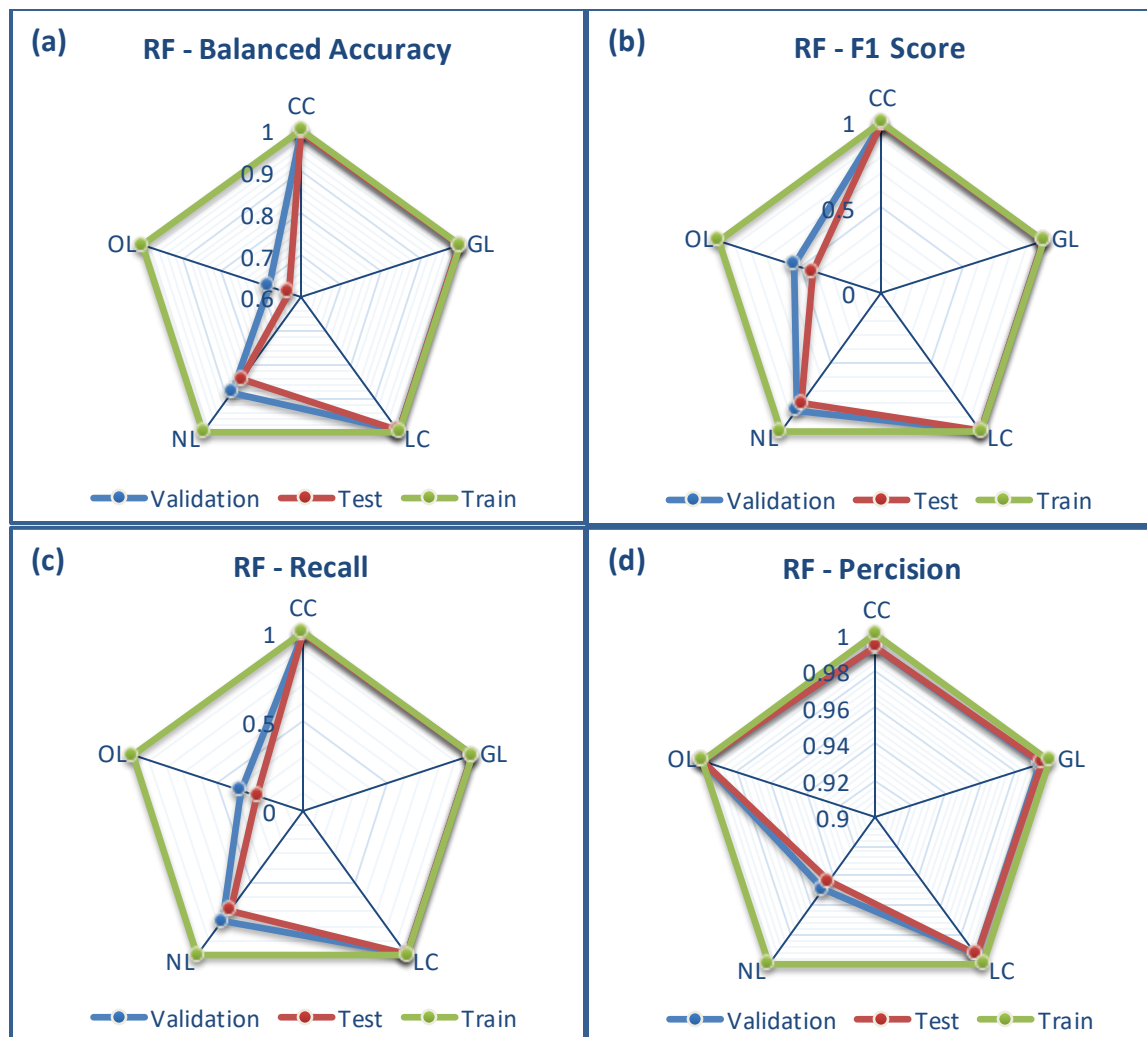


Figure 4.40: Radar charts illustrating RF model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

Table 4.33 provides a detailed breakdown of LightGBM's performance across different severities of leaks in looped water distribution networks. One of the most notable strengths of this model is its ability to detect severe leaks, with recall values exceeding 0.95 in multiple configurations. This indicates that when a severe leak occurs, the model is highly reliable in identifying it, significantly reducing the risk of false negatives in critical leak detection scenarios. The balanced accuracy for severe leaks remains consistently high, reaching up to 0.92 in LO_OL, suggesting that the model effectively differentiates between

severe and non-severe cases. Additionally, the high specificity values (above 0.86 for severe leaks) confirm that the model is not prone to falsely classifying non-severe leaks as severe, which is crucial for preventing unnecessary interventions.

However, LightGBM demonstrates noticeable weaknesses in detecting minor leaks. The recall values for minor classifications are relatively low, particularly in LO_LC (0.57) and LO_CC (0.66), indicating a substantial number of misclassifications. While the specificity for minor leaks is high (often exceeding 0.94), the low recall suggests that the model frequently fails to correctly identify minor leak events, instead categorizing them as moderate or severe. This results in an overestimation of leak severity, which is further confirmed by moderate precision values (ranging between 0.72 and 0.82). This discrepancy suggests that the model is better at ruling out non-minor leaks rather than correctly classifying minor ones. Additionally, moderate leak detection shows some instability, with recall values dropping as low as 0.63 in LO_GL. These inconsistencies suggest that the model struggles to balance the trade-off between precision and recall for lower severity classifications, likely due to the inherent difficulty of distinguishing minor leaks in looped network configurations.

The confusion matrices (Figure 4.41) provide deeper insights into the misclassification patterns observed in Table 4.33. The most evident issue is the misclassification of minor leaks, which are frequently categorized as moderate or severe. For instance, in LO_CC, 25,533 minor cases were misclassified as moderate, and 5,732 were labeled as severe. A similar trend is observed in LO_LC, where 14,283 minor leaks were classified as moderate, and 3,299 were classified as severe. These results indicate a

systematic bias in LightGBM's classification, favoring higher severity labels, particularly in cases where the distinction between minor and moderate leaks is subtle.

Moderate leaks also exhibit classification inconsistencies, particularly in LO_GL, where only 1,446 of the actual moderate leaks were correctly classified, while a significant number were misclassified as minor or severe. This suggests that the model does not effectively capture the distinguishing features of moderate leaks, potentially due to the overlapping pressure and flow rate characteristics between minor and moderate cases. On the other hand, severe leaks show strong classification performance, with the majority of true severe cases correctly identified. For example, in LO_OL, 47,638 severe leaks were correctly classified, while only 203 were misclassified as minor and 1,124 as moderate. This highlights the model's confidence in detecting high-severity events while struggling with lower-severity distinctions.

The misclassification patterns in the confusion matrices suggest that looped networks pose unique challenges for LightGBM. Unlike branched networks, looped systems distribute water through multiple alternative paths, making small leaks less distinguishable from normal operational fluctuations. This structural characteristic likely contributes to the model's overestimation of severity, as minor leaks may not generate sufficiently distinct pressure drops to be correctly classified. Additionally, the imbalanced distribution of leak severities in the training dataset may further exacerbate this issue, as the model appears to favor moderate and severe classifications over minor ones.

Table 4.33: By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

LightGBM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9448	0.8753	0.8275	0.6631	0.7363	0.8040
	moderate	0.8524	0.8797	0.7440	0.7863	0.7646	0.8194
	severe	0.8635	0.9162	0.7813	0.8605	0.8190	0.8620
LO_LC	minor	0.9560	0.8941	0.7737	0.5704	0.6567	0.7632
	moderate	0.8711	0.8389	0.7723	0.7233	0.7470	0.7972
	severe	0.8321	0.9614	0.8008	0.9528	0.8702	0.8925
LO_GL	minor	0.9687	0.8627	0.9212	0.7036	0.7978	0.8362
	moderate	0.9959	0.9912	0.8235	0.6827	0.7465	0.8393
	severe	0.7177	0.9338	0.8545	0.9702	0.9087	0.8440
LO_OL	minor	0.9739	0.8941	0.9074	0.6891	0.7833	0.8315
	moderate	0.9185	0.9108	0.8282	0.8136	0.8208	0.8661
	severe	0.8628	0.9792	0.8277	0.9729	0.8944	0.9178

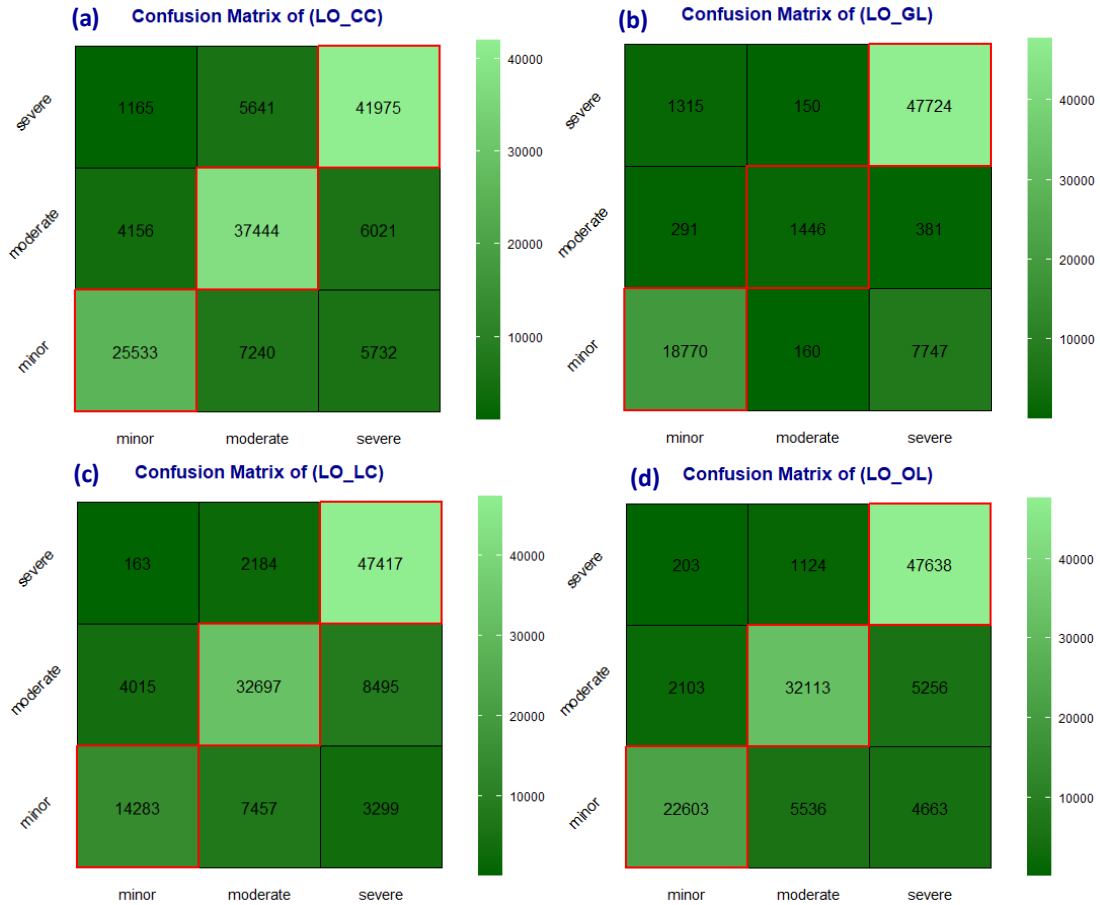
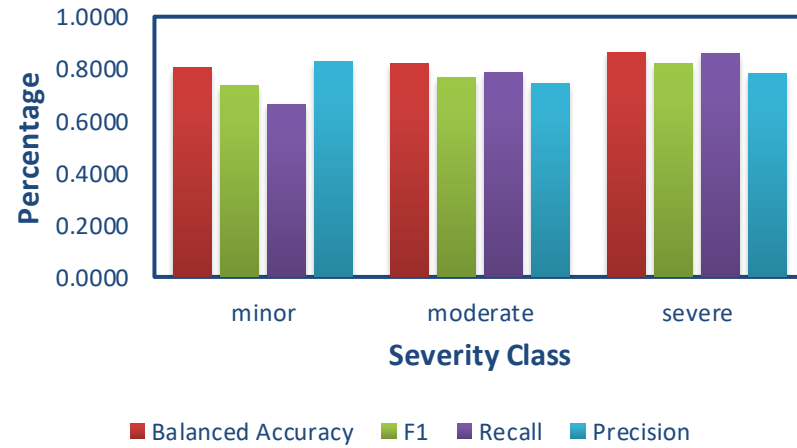


Figure 4.41: Confusion matrices for four Looped network LightGBM models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

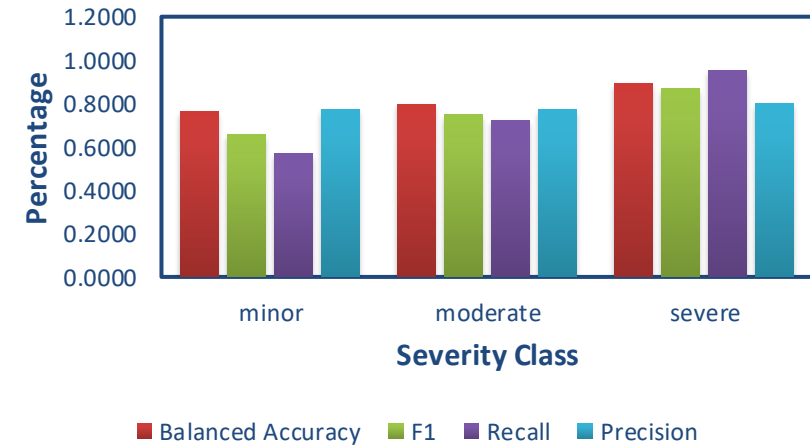
Figure 4.42 visualizes the key performance metrics across different severities and network configurations, reinforcing the observations from Table 4.33 and the confusion matrices. The balanced accuracy scores remain consistently high for severe leaks, confirming the model's reliability in detecting critical failures. However, a clear performance gap is evident when comparing minor and moderate classifications. The recall values for minor leaks exhibit significant variability, dipping below 0.60 in some cases. This suggests that LightGBM's ability to correctly identify minor leaks is inconsistent and likely dependent on network topology. The LO_LC and LO_CC configurations, in particular, show the lowest recall scores for minor leaks, aligning with the misclassification trends observed in the confusion matrices.

Precision values, while relatively stable for severe leaks, fluctuate for minor and moderate classifications. In LO_GL, for instance, precision for moderate leaks is approximately 0.82, while recall for the same class is only 0.68. This indicates that while the model is confident in its moderate leak predictions, it may not always capture all moderate cases correctly, leading to lower recall. Additionally, the model's specificity remains high across all severities, reinforcing its strength in correctly excluding non-matching classifications. However, specificity alone does not compensate for the low recall in minor leak detection, as high specificity merely indicates that when the model predicts a certain class, it is likely correct, rather than ensuring all instances of that class are detected.

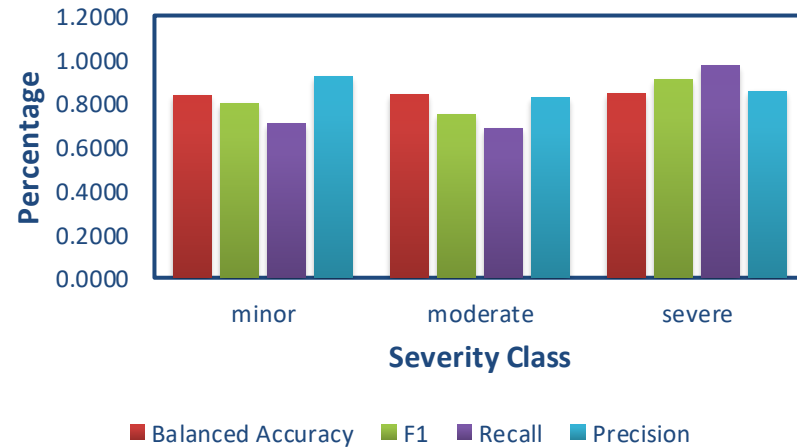
(a) LightGBM - Looped network - CC



(b) LightGBM - Looped network - LC



(c) LightGBM - Looped network - GL



(d) LightGBM - Looped network - OL

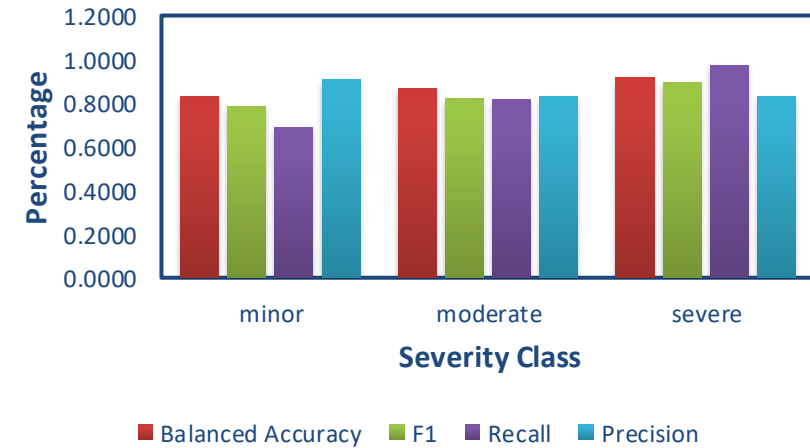


Figure 4.42: Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

The inconsistencies observed in the bar charts further support the notion that LightGBM struggles with minor leak detection in looped networks due to the system's inherent pressure stabilization mechanisms. The presence of multiple pathways redistributes flow, reducing the impact of minor leaks on pressure readings, which may limit the model's ability to differentiate them from normal fluctuations. To address these challenges, several strategies could be explored, including improved feature engineering to enhance sensitivity to subtle leak signatures, resampling techniques to balance the dataset, and integrating temporal dependencies to capture progressive leak developments over time. Overall, while LightGBM excels in detecting severe leaks, its inconsistent performance in minor and moderate leak classifications highlights the need for targeted refinements to improve its reliability in looped network applications.

In the leak detection task for a Looped WDS, the LightGBM model demonstrates strong performance across all datasets and leak types (CC, GL, LC, NL, OL), as shown in Table 4.34. Balanced Accuracy remains high, with most values exceeding 0.99, except for OL in the validation (0.69988) and test datasets (0.88939), indicating challenges in detecting this specific leak type. Similarly, F1 Scores are consistently high, ranging from 0.9912 (CC - validation) to 0.9978 (GL - validation), but OL shows a significant drop (0.4068 in validation, 0.8558 in test), suggesting lower model confidence in this class.

Table 4.34: By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99747	0.99646	0.99343	0.99083	0.99213	0.99415
	GL	0.99785	0.99976	0.99618	0.99957	0.99787	0.99871
	LC	0.99680	0.99634	0.99429	0.99348	0.99388	0.99514
	NL	0.99994	0.99976	0.96364	0.86179	0.90987	0.93087
	OL	0.99976	0.99975	0.41379	0.40000	0.40678	0.69988
Test	CC	0.99690	0.99640	0.99198	0.99068	0.99133	0.99379
	GL	0.99820	0.99971	0.99680	0.99949	0.99815	0.99885
	LC	0.99660	0.99623	0.99394	0.99328	0.99361	0.99494
	NL	0.99993	0.99962	0.95000	0.77869	0.85586	0.88931
	OL	0.99969	0.99963	0.15385	0.13333	0.14286	0.56651
Train	CC	0.99864	0.99825	0.99649	0.99548	0.99598	0.99706
	GL	0.99960	0.99992	0.99928	0.99987	0.99957	0.99973
	LC	0.99814	0.99825	0.99669	0.99689	0.99679	0.99751
	NL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	OL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

The radar charts in Figure 4.43 further illustrate these variations. Precision and Recall scores for OL are significantly lower, with test set values of 0.1534 (Precision) and 0.1333 (Recall), indicating a higher rate of misclassifications. Other leak types, particularly CC, GL, and LC, maintain near-perfect scores across all metrics. Despite these minor inconsistencies, the overall model performance is robust, with strong generalization to unseen data. Further improvements, such as refining class balancing techniques or optimizing detection thresholds for OL, could enhance model reliability in a Looped WDS.

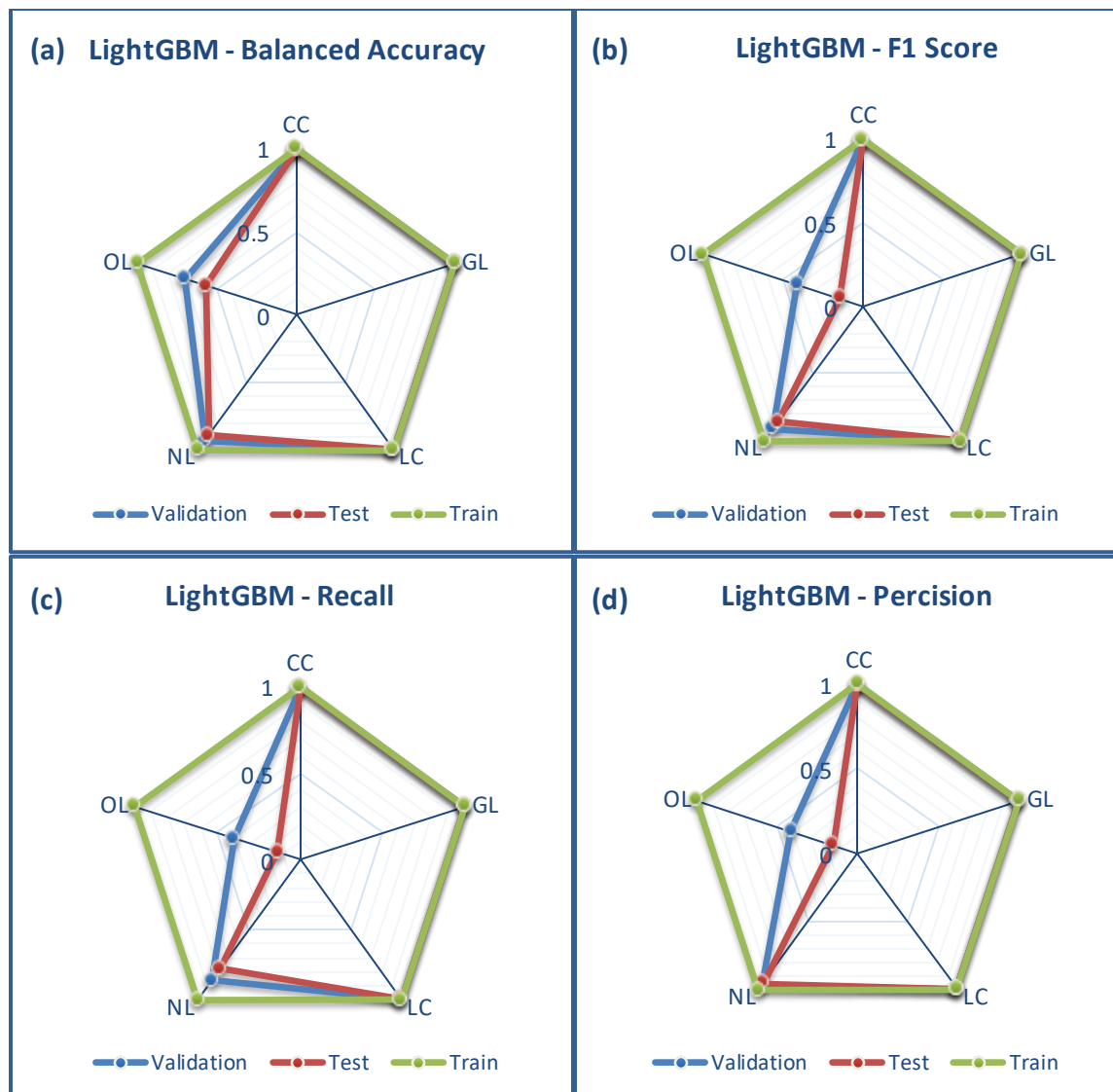


Figure 4.43: Radar charts illustrating LightGBM model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The XGB model exhibits a reasonable performance across different severity classes in looped water distribution networks, as indicated in Table 4.35. The recall for severe leaks is consistently high, exceeding 0.94 in most cases, demonstrating the model's strong capability to detect critical leak incidents. This high recall minimizes the likelihood of false negatives in severe cases, which is crucial for effective leak management. However, moderate leaks are detected with slightly lower recall, ranging between 0.68 and 0.75,

indicating that while the model can generally identify these leaks, there is room for improvement in distinguishing them from minor or severe cases. The specificity values are high for minor leaks, often surpassing 0.92, which suggests that the model effectively rules out non-minor cases, though its ability to correctly classify them remains weaker.

Despite its strengths, the model struggles with minor leak detection, as evidenced by recall values as low as 0.51 in LO_LC. This suggests that a significant portion of minor leaks is misclassified as moderate or even severe. The low recall for minor leaks correlates with lower balanced accuracy scores in the minor class, typically below 0.80, reinforcing the notion that the model favors moderate and severe classifications. Precision scores for all severity levels are relatively balanced, with values ranging from 0.69 to 0.94, suggesting that when the model does classify an instance into a specific category, it is generally reliable. The F1 scores mirror this trend, with severe cases achieving the highest values, reflecting a balance between precision and recall, whereas minor leaks suffer from a lower F1 score due to reduced recall.

The confusion matrices for the XGB model across different network configurations (Figure 4.44) provide further insight into its classification behavior. One of the most striking patterns is the consistent misclassification of minor leaks into moderate and severe categories. In LO_CC, for instance, 24,140 minor cases were classified as moderate, and 6,468 were even misidentified as severe. Similarly, in LO_GL, 18,259 minor cases were labeled as moderate, with only a small fraction correctly identified as minor leaks. This misclassification pattern is problematic, as it inflates the detection of higher-severity leaks while underestimating minor issues.

For moderate leaks, the model displays mixed performance, with varying degrees of misclassification into either minor or severe categories. In LO_LC, 4,911 moderate cases were misclassified as minor, while 9,511 were mistaken for severe, suggesting that the model struggles to draw a clear distinction between these classes. However, for severe leaks, the model performs well, with the majority of true severe cases correctly identified. For example, in LO_OL, 46,313 severe cases were correctly classified, though a small proportion was misclassified into lower-severity classes. These results suggest that the model prioritizes identifying critical leaks, but at the cost of reduced accuracy for minor and moderate cases.

Table 4.35: By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

XGB Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9295	0.8618	0.7803	0.6269	0.6953	0.7782
	moderate	0.8251	0.8557	0.6992	0.7449	0.7213	0.7850
	severe	0.8393	0.8851	0.7400	0.8076	0.7723	0.8234
LO_LC	minor	0.9459	0.8801	0.7135	0.5112	0.5956	0.7285
	moderate	0.8381	0.8130	0.7176	0.6810	0.6988	0.7595
	severe	0.8146	0.9406	0.7799	0.9274	0.8473	0.8710
LO_GL	minor	0.9651	0.8547	0.9107	0.6844	0.7815	0.8248
	moderate	0.9989	0.9909	0.9456	0.6723	0.7859	0.8356
	severe	0.6952	0.9287	0.8445	0.9687	0.9023	0.8320
LO_OL	minor	0.9511	0.8671	0.8215	0.6068	0.6980	0.7790
	moderate	0.8847	0.8740	0.7549	0.7357	0.7452	0.8102
	severe	0.8309	0.9577	0.7912	0.9458	0.8616	0.8883

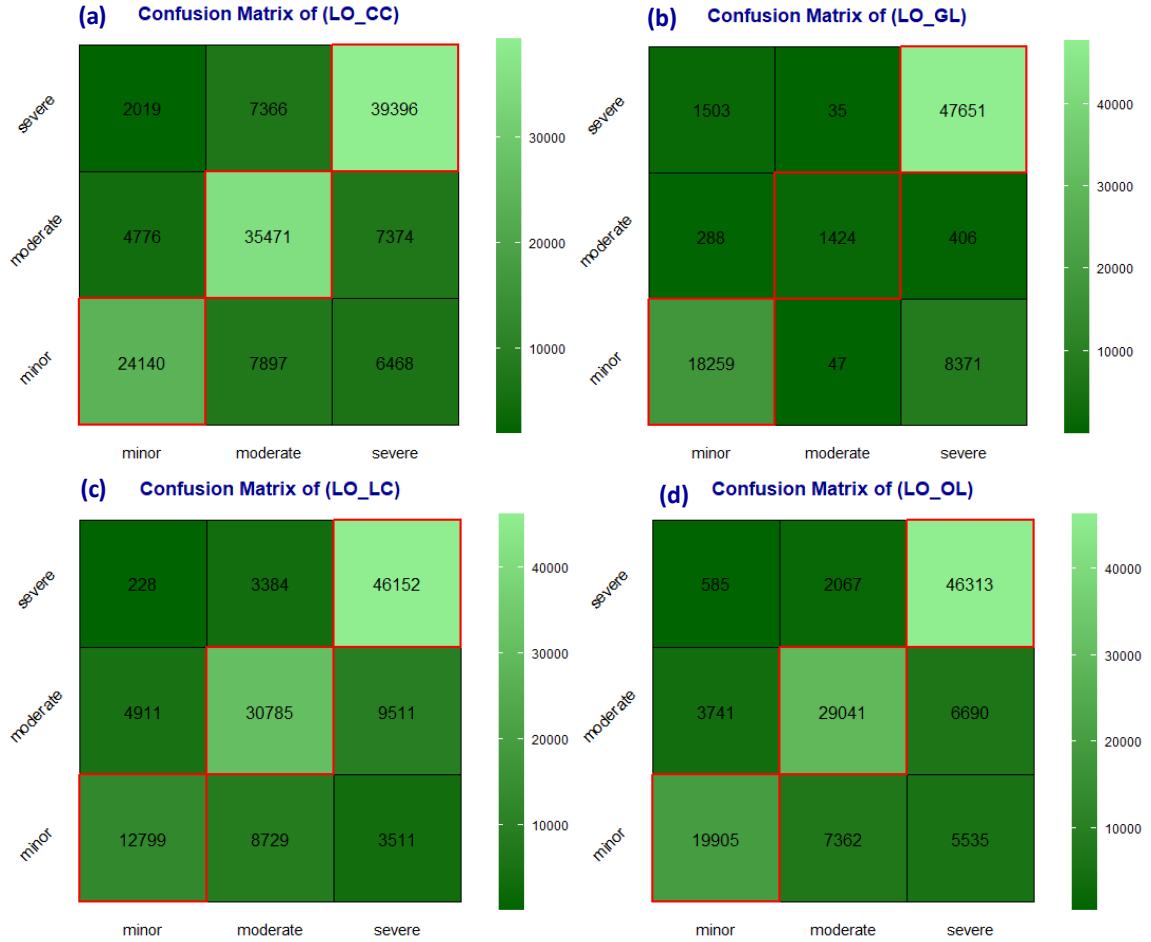


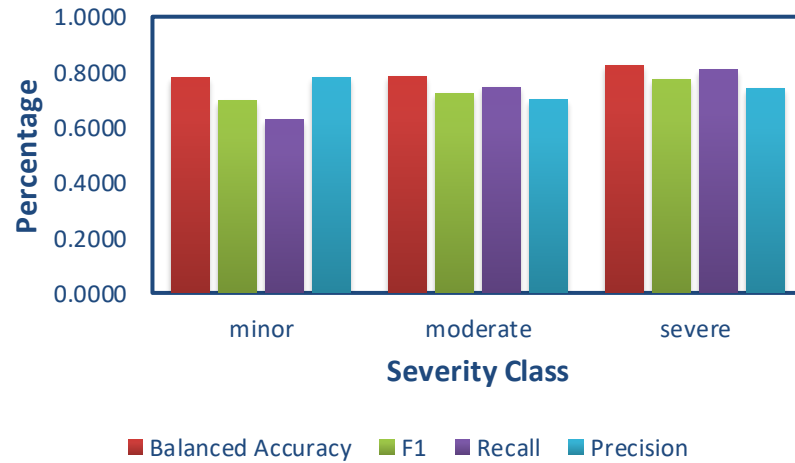
Figure 4.44: Confusion matrices for four Looped network XGB models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

The bar charts in Figure 4.45 further illustrate the performance trends observed in the previous analyses. Severe leaks consistently exhibit the highest balanced accuracy, recall, and F1 scores across all looped network configurations, affirming the model's reliability in identifying critical leakages. Precision values for severe cases remain relatively high, ensuring that most classified severe leaks are indeed correctly identified. In contrast, minor leaks consistently show the weakest recall, reflecting the model's difficulty in identifying these leaks accurately. This trend is particularly evident in LO_LC and LO_GL, where recall values for minor cases are substantially lower than those for moderate and severe categories.

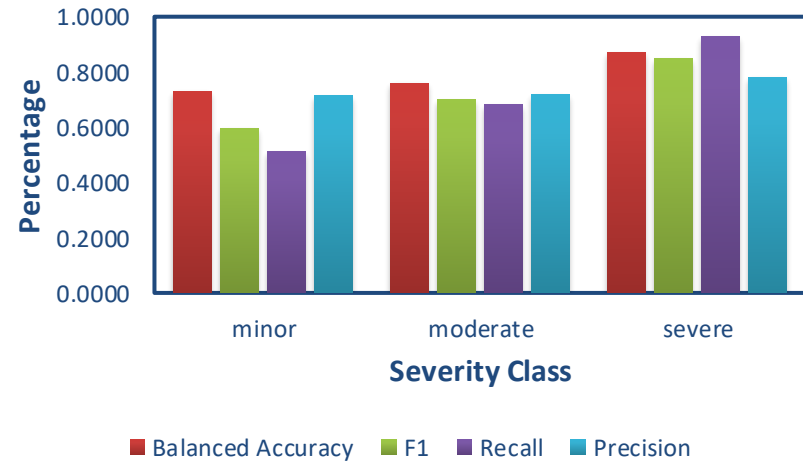
A possible explanation for this trend is the inherent difficulty in detecting minor leaks in looped networks due to pressure stabilization effects. Unlike branched networks, looped configurations distribute flow through multiple paths, reducing pressure fluctuations that could serve as indicators of small leaks. The model's tree-based learning structure may also be contributing to this issue, as minor leaks may not exhibit distinct enough feature variations for the model to confidently separate them from normal operational noise. The persistent misclassification of minor leaks suggests that additional preprocessing steps, such as synthetic data augmentation or feature engineering, may be necessary to enhance model performance in this category. Additionally, an imbalance in the dataset distribution, where minor leaks are underrepresented, could be causing the model to favor moderate and severe classifications. Resampling techniques, such as oversampling minor cases, could mitigate this issue and improve recall for the minor class.

Overall, the XGB model performs robustly for severe leak detection but requires significant improvements in minor and moderate classification. The results highlight the challenges posed by looped networks in leak detection and underscore the need for further model refinements to achieve a more balanced classification performance across all severity levels.

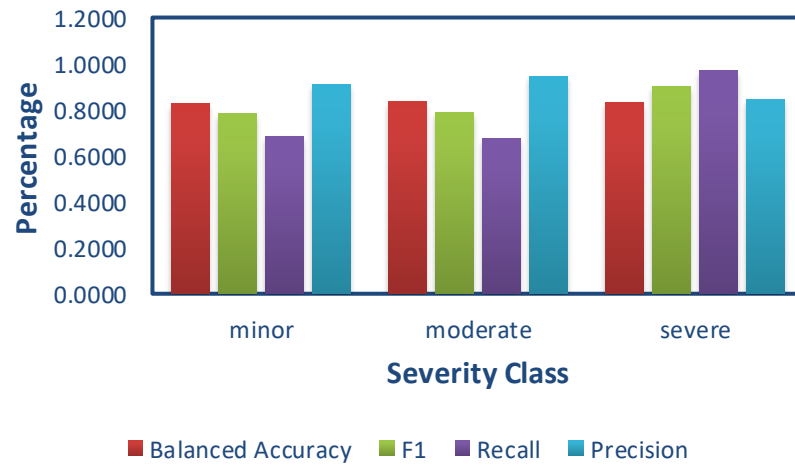
(a) XGB - Looped network - CC



(b) XGB - Looped network - LC



(c) XGB - Looped network - GL



(d) XGB - Looped network - OL

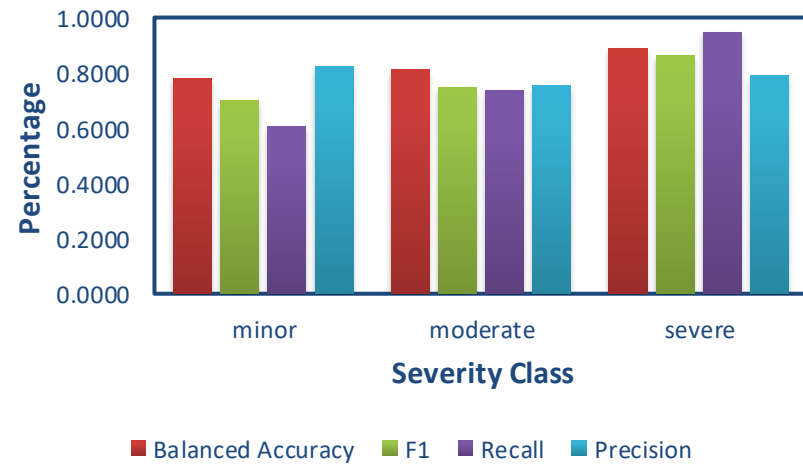


Figure 4.45: Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

For the leak detection task in a Looped WDS, the XGB model achieves strong performance across most leak types, as shown in Table 4.36. Balanced Accuracy is consistently high for CC (0.9925 validation, 0.9895 test) and GL (0.9993 validation, 0.9985 test), while OL presents some classification challenges, with values dropping to 0.8167 in both validation and test datasets. F1 Scores follow a similar pattern, maintaining values above 0.995 for CC, GL, and LC, whereas OL records 0.7755 in validation and 0.76 in the test dataset.

Table 4.36: By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99961	0.99957	0.99899	0.99889	0.99894	0.99925
	GL	0.99890	0.99989	0.99805	0.99980	0.99893	0.99935
	LC	0.99978	0.99928	0.99961	0.99871	0.99916	0.99925
	NL	0.99999	0.99986	0.99123	0.91870	0.95359	0.95934
	OL	1.00000	0.99985	1.00000	0.63333	0.77551	0.81667
Test	CC	0.99953	0.99938	0.99879	0.99839	0.99859	0.99896
	GL	0.99893	0.99993	0.99809	0.99988	0.99898	0.99940
	LC	0.99967	0.99930	0.99941	0.99875	0.99908	0.99921
	NL	0.99994	0.99979	0.96396	0.87705	0.91845	0.93850
	OL	0.99999	0.99985	0.95000	0.63333	0.76000	0.81666
Train	CC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	GL	1.00000	1.00000	0.99999	1.00000	1.00000	1.00000
	LC	1.00000	1.00000	0.99999	0.99999	0.99999	0.99999
	NL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	OL	1.00000	1.00000	1.00000	0.99301	0.99649	0.99650

The radar charts in Figure 4.46 illustrate these patterns, showing strong precision and recall for most leak types while revealing the lower classification confidence for OL. The recall for OL reaches only 0.6333 across validation and test datasets, indicating a higher number of false negatives for this class. Despite these variations, the model demonstrates excellent generalization, making it a reliable choice for leak detection in

Looped WDS networks, with room for improvement in handling underrepresented leak types.

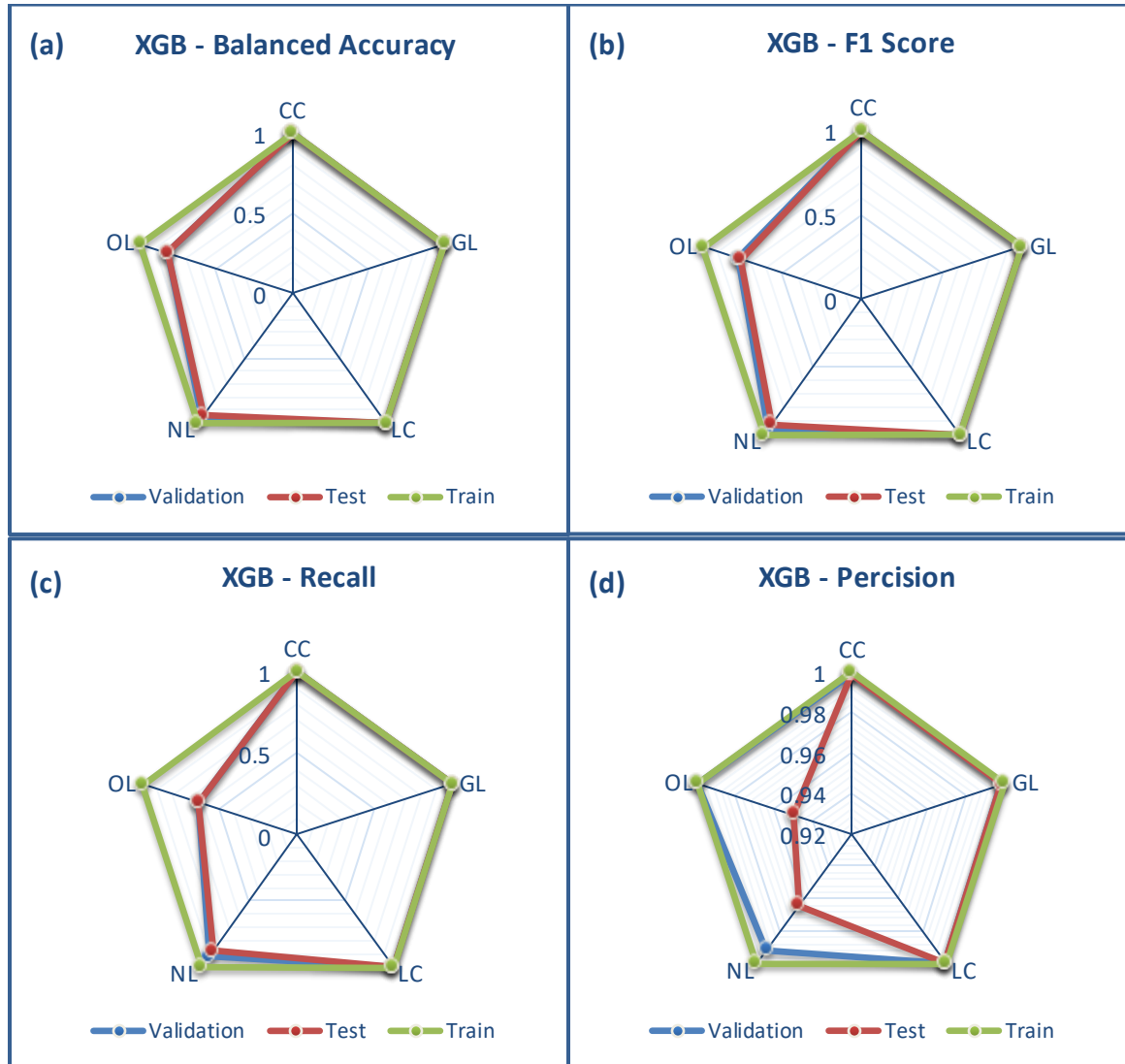


Figure 4.46: Radar charts illustrating XGB model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

4.3.1.2 Deep Learning models

Table 4.37 provides a comparative analysis of three deep learning models (LSTM, RNN, and GRU) across four different leak types in a Looped WDS network, using a dataset processed by IQR outlier removal. The performance metrics, including Accuracy, Kappa, and Confidence Intervals (CI Lower and CI Upper), offer a detailed assessment of the models' predictive capabilities. GRU demonstrates the highest Accuracy and Kappa scores for most leak types, particularly for LO_GL and LO_OL, indicating its superior effectiveness in detecting leaks in the Looped WDS network. LSTM shows competitive performance but exhibits a noticeable decrease in Accuracy and Kappa for LO_CC and LO_LC, suggesting challenges in handling these leak types effectively. RNN, while maintaining a balanced performance, falls slightly behind GRU and LSTM in most metrics but displays comparable results for LO_GL, indicating a stable yet less dominant performance. The Confidence Intervals in Table 4.37 highlight the reliability of these models, with narrower intervals observed for GRU, suggesting higher stability and lower variance in predictions compared to LSTM and RNN. Overall, the analysis suggests that GRU is the most reliable model for leak detection in a Looped WDS, followed by LSTM and RNN, each showing distinct strengths across different leak types.

Figure 4.47 presents a visual comparison of the performance of three deep learning models (LSTM, RNN, and GRU) across various leak types in a Looped WDS network, following IQR-based outlier removal. The bar heights, representing Accuracy, Kappa, and Confidence Intervals, highlight distinct trends in model performance. GRU demonstrates the highest bar heights for LO_GL and LO_OL, suggesting its superior predictive capability for these leak types. LSTM exhibits competitive performance for LO_GL but

shows reduced Accuracy and Kappa for LO_CC and LO_LC, indicating challenges in detecting certain leak types effectively. RNN presents a balanced performance across most leak types, with moderate Accuracy and Kappa values, reflecting stable but less dominant predictive capabilities compared to GRU. The Confidence Intervals are notably narrower for GRU, indicating higher reliability and lower variance in predictions, while LSTM and RNN display wider intervals, suggesting less stable performance. Overall, the figure highlights GRU as the most reliable model for leak detection in a Looped WDS, followed by LSTM and RNN, each exhibiting unique strengths across different leak types.

Table 4.37: Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over looped network after removing the outliers.

Model	Metric	LO_CC	LO_LC	LO_GL	LO_OL
LSTM	Accuracy	0.678850188	0.6907059	0.8402406	0.7027078
	Kappa	0.512538758	0.5047547	0.6451627	0.5395369
	CI Lower	0.676351115	0.6880822	0.8376501	0.7001261
	CI Upper	0.681341572	0.6933204	0.8428057	0.7052799
	No Info Rate	0.361584475	0.4146669	0.6307414	0.4038733
RNN	Accuracy	0.683779437	0.6843816	0.8418691	0.7060153
	Kappa	0.518292095	0.4914966	0.6468864	0.5410238
	CI Lower	0.681290492	0.6817435	0.8392892	0.7034418
	CI Upper	0.68626048	0.6870107	0.8444235	0.7085789
	No Info Rate	0.361584475	0.4146669	0.6307414	0.4038733
GRU	Accuracy	0.681022022	0.6924808	0.8410869	0.7057184
	Kappa	0.515386561	0.509018	0.6469675	0.5440043
	CI Lower	0.678527372	0.6898611	0.8385019	0.7031441
	CI Upper	0.68350889	0.6950911	0.8436465	0.7082827
	No Info Rate	0.361584475	0.4146669	0.6307414	0.4038733

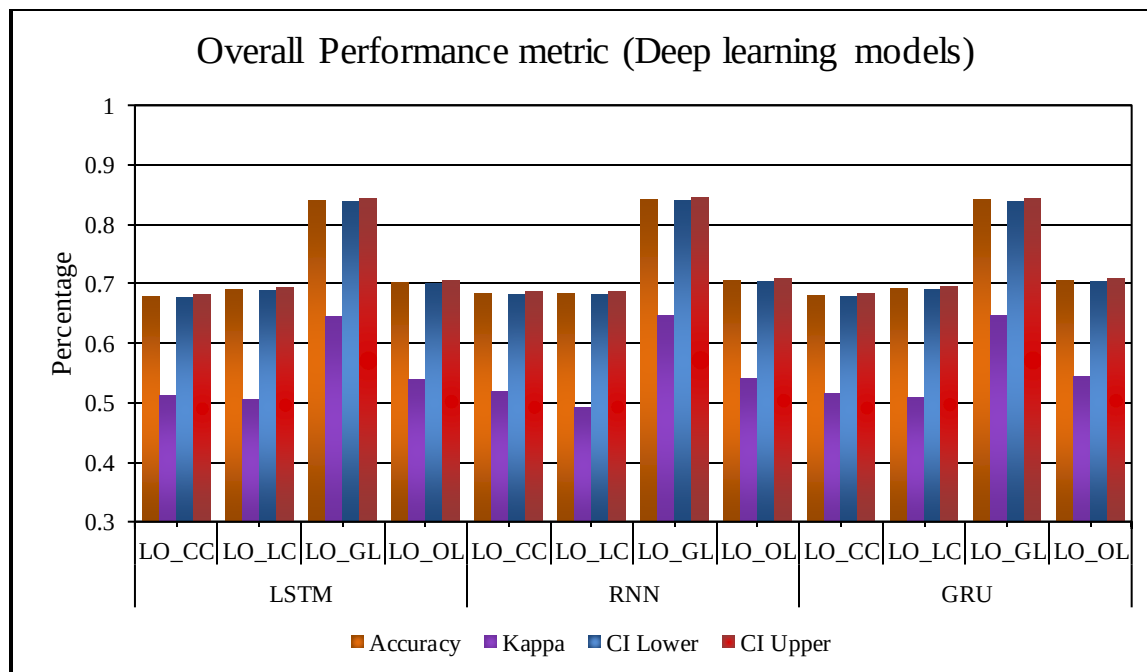


Figure 4.47: Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over looped network after removing the outliers.

The side-by-side comparison of the two figures (Figure 4.47, and Figure 4.14) reveals a noticeable difference in the performance of deep learning models (LSTM, RNN, and GRU) between branched and looped WDS networks. In the branched network, the models exhibit higher Accuracy and Kappa values, particularly for GRU and LSTM, which achieve consistently strong performance across most leak types. The Confidence Intervals are also narrower, indicating more reliable and stable predictions.

Several factors could contribute to the relatively poor severity assessment performance in looped networks compared to branched networks. Firstly, the increased hydraulic redundancy and complexity in looped networks can cause more subtle and dispersed variations in sensor readings (acoustic, pressure, and vibrational), making it harder for models to accurately classify leaks into minor, moderate, or severe categories. The overlapping paths for water flow in looped networks might dilute or mask critical

signals captured by these sensors, reducing the distinctiveness of features that the models rely on for severity classification. Additionally, the interaction of multiple flow paths can introduce more noise into the dataset, potentially leading to higher variance in model predictions and wider Confidence Intervals. The complex spatial and temporal patterns of pressure drops, acoustic signatures, and vibrational changes in looped networks could further complicate the extraction of meaningful features necessary for accurate severity assessment. The observed performance degradation suggests that advanced feature engineering or specialized model architectures that can better capture the intricate dependencies in a multi-sensor framework are essential to improve the severity classification accuracy in looped networks.

The LSTM model exhibits inconsistent performance across different severity classes and looped network configurations. The classification, in Table 4.38, of severe leaks shows the highest effectiveness, with recall values ranging from 0.73 in LO_CC to approximately 0.91 in LO_LC and LO_OL, reflecting the model's strong capability in identifying severe leak events. However, the precision scores for severe cases remain relatively lower, ranging between 0.63 and 0.74, suggesting a tendency for false positives. Moderate leak detection appears to be weaker compared to tree-based models, with recall values fluctuating between 0.57 and 0.70 across configurations, highlighting its difficulty in distinguishing moderate leaks. The model struggles significantly with minor leaks, where recall is particularly low, reaching only 0.38 in LO_LC and 0.45 in LO_OL. The overall balanced accuracy values range between 0.66 and 0.85, with the lowest observed in LO_LC for minor leaks. This suggests that the LSTM model is highly sensitive to the underlying network topology, possibly due to its reliance on temporal dependencies rather

than static decision boundaries. The specificity values for minor leaks remain high, often exceeding 0.90, indicating that while the model can rule out minor leaks effectively, it fails to correctly classify them when present.

Table 4.38: By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

LSTM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9167	0.8446	0.7347	0.5777	0.6468	0.7472
	moderate	0.7867	0.8282	0.6419	0.7009	0.6701	0.7438
	severe	0.8064	0.8442	0.6832	0.7372	0.7092	0.7718
LO_LC	minor	0.9510	0.8548	0.6758	0.3870	0.4922	0.6690
	moderate	0.7693	0.7783	0.6255	0.6374	0.6314	0.7034
	severe	0.7834	0.9110	0.7447	0.8919	0.8117	0.8377
LO_GL	minor	0.9581	0.8348	0.8875	0.6354	0.7406	0.7967
	moderate	0.9985	0.9882	0.9120	0.5720	0.7030	0.7852
	severe	0.6460	0.9107	0.8229	0.9629	0.8874	0.8045
LO_OL	minor	0.9285	0.8221	0.7039	0.4582	0.5551	0.6934
	moderate	0.8207	0.8292	0.6363	0.6497	0.6429	0.7352
	severe	0.7916	0.9279	0.7472	0.9092	0.8203	0.8504

The confusion matrices provide deeper insight into the misclassification trends observed in Table 4.38. Figure 4.48a reveals that minor leaks are frequently misclassified as moderate (8,554 cases) or severe (7,472 cases), demonstrating the model's struggle to differentiate lower-severity events. The misclassification of minor leaks is also apparent in Figure 4.48b, where 17,136 minor leaks are wrongly labeled as moderate or severe, further emphasizing the model's difficulty in distinguishing subtle leak patterns. The severe class, while generally well detected, still suffers from substantial misclassification, as seen in Figure 4.48c, where 4,944 severe cases are identified as moderate. A similar trend is noticeable in Figure 4.48d, where 3,347 severe leaks are classified as moderate. These findings suggest that the LSTM model struggles to draw a clear boundary between moderate and severe leaks, likely due to overlapping signal characteristics. The failure to

classify minor leaks correctly can be attributed to the model's dependency on sequential pressure changes, which may not be sufficiently distinct in looped networks where pressure variations are more stabilized.

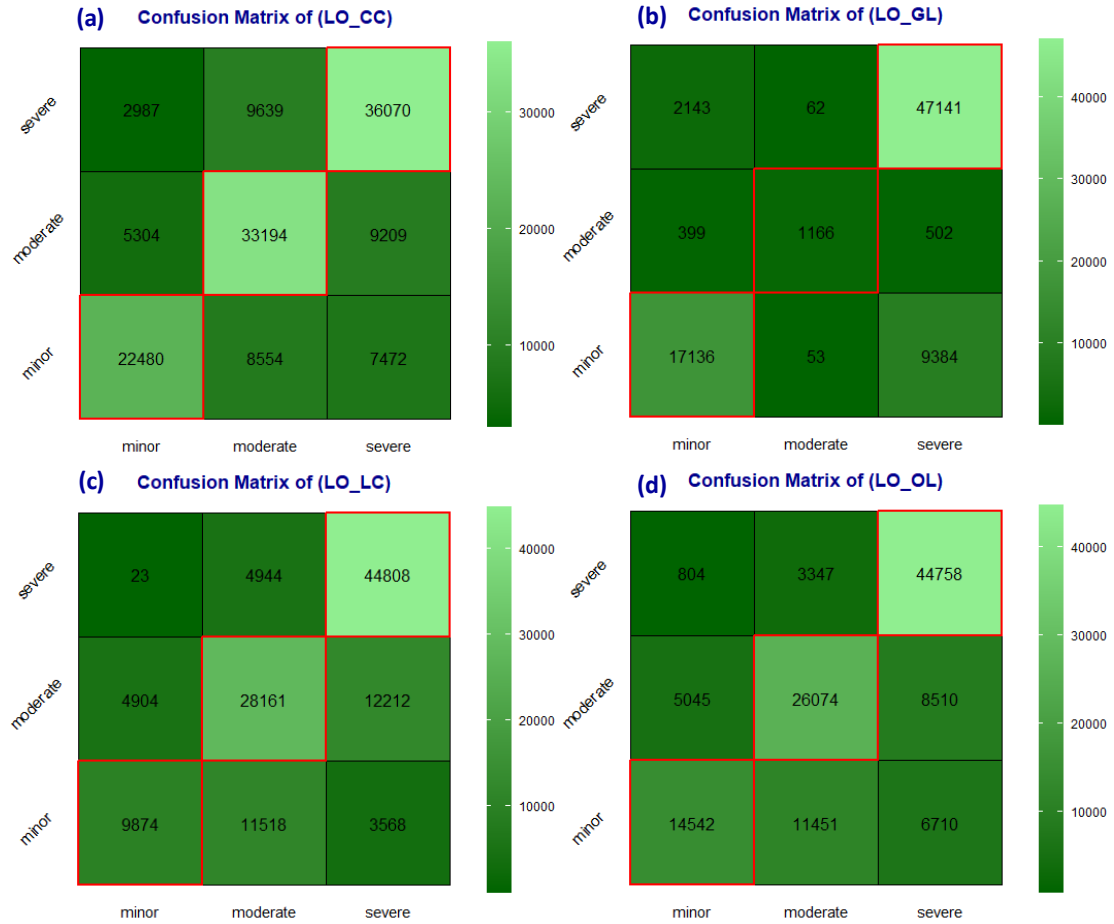


Figure 4.48: Confusion matrices for four Looped network LSTM models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

The bar charts in Figure 4.49 further illustrate the LSTM model's imbalanced performance across severity classes. Severe leak detection achieves the highest recall values across all configurations, reinforcing the model's strength in capturing major leak events. However, moderate and minor leak detection presents significant inconsistencies. Notably, in Figure 4.49b, which corresponds to LO_LC, recall and F1 scores for minor

leaks are significantly lower than other severity levels, indicating a lack of predictive confidence for this category. The moderate class displays varied precision and recall values, suggesting an unstable classification pattern, particularly in Figure 4.49c, where the model demonstrates higher precision but lower recall, hinting at a conservative classification approach. The balanced accuracy scores across all severity classes confirm the model's general weakness in minor leak identification. The bar chart patterns further support the hypothesis that LSTM struggles with minor leaks due to the smooth pressure redistribution in looped networks, leading to weaker signal variations that are insufficient for accurate temporal pattern recognition.

The LSTM model shows its strongest performance in detecting severe leaks but suffers from significant classification errors for minor and moderate leaks, with particularly poor recall for minor cases. This can be attributed to the model's reliance on temporal patterns, which may not exhibit distinct signatures for small leaks in looped networks. The confusion matrices and performance charts collectively indicate that the LSTM model is better suited for identifying major disruptions but lacks the granularity required to differentiate between moderate and minor leak severities. Future improvements could involve incorporating additional signal preprocessing techniques, such as pressure trend normalization or anomaly detection mechanisms, to enhance sensitivity to minor leak events. Additionally, alternative deep learning architectures, such as hybrid CNN-LSTM models, could be explored to extract more robust spatial-temporal features and mitigate misclassification challenges.

(a) LSTM - Looped network - CC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(b) LSTM - Looped network - LC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(c) LSTM - Looped network - GL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(d) LSTM - Looped network - OL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

Figure 4.49: Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

For the leak detection task in a Looped WDS, the LSTM model performs with high accuracy across most leak types, as detailed in Table 4.39. Balanced Accuracy remains consistently strong, exceeding 0.995 for CC, GL, and NL in both test and train datasets. However, LC exhibits a slightly lower Balanced Accuracy of 0.9927 in the test dataset, indicating minor classification difficulties. The F1 Scores also reflect this trend, surpassing 0.99 for most classes, ensuring effective precision-recall balance.

Table 4.39: By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99658	0.99855	0.98641	0.99420	0.99029	0.99539
	GL	0.99950	0.99951	0.99800	0.99804	0.99802	0.99877
	LC	0.99836	0.99678	0.99340	0.98710	0.99024	0.99273
	NL	0.99953	0.99973	0.99812	0.99892	0.99852	0.99923
	OL	0.99985	0.99925	0.99938	0.99698	0.99818	0.99841
Train	CC	0.99691	0.99867	0.98773	0.99468	0.99119	0.99580
	GL	0.99967	0.99961	0.99867	0.99843	0.99855	0.99905
	LC	0.99846	0.99708	0.99380	0.98829	0.99104	0.99337
	NL	0.99965	0.99982	0.99861	0.99928	0.99894	0.99946
	OL	0.99991	0.99943	0.99962	0.99770	0.99866	0.99880

The radar charts in Figure 4.50 confirm these findings, illustrating strong alignment between train and test results. Precision and Recall values remain high, with NL and OL maintaining 0.998 or higher across all metrics. The model demonstrates excellent generalization, with minimal deviation between datasets. These results confirm the LSTM model's robustness in leak detection for Looped WDS, making it a reliable choice with minor room for improvement in LC classification.

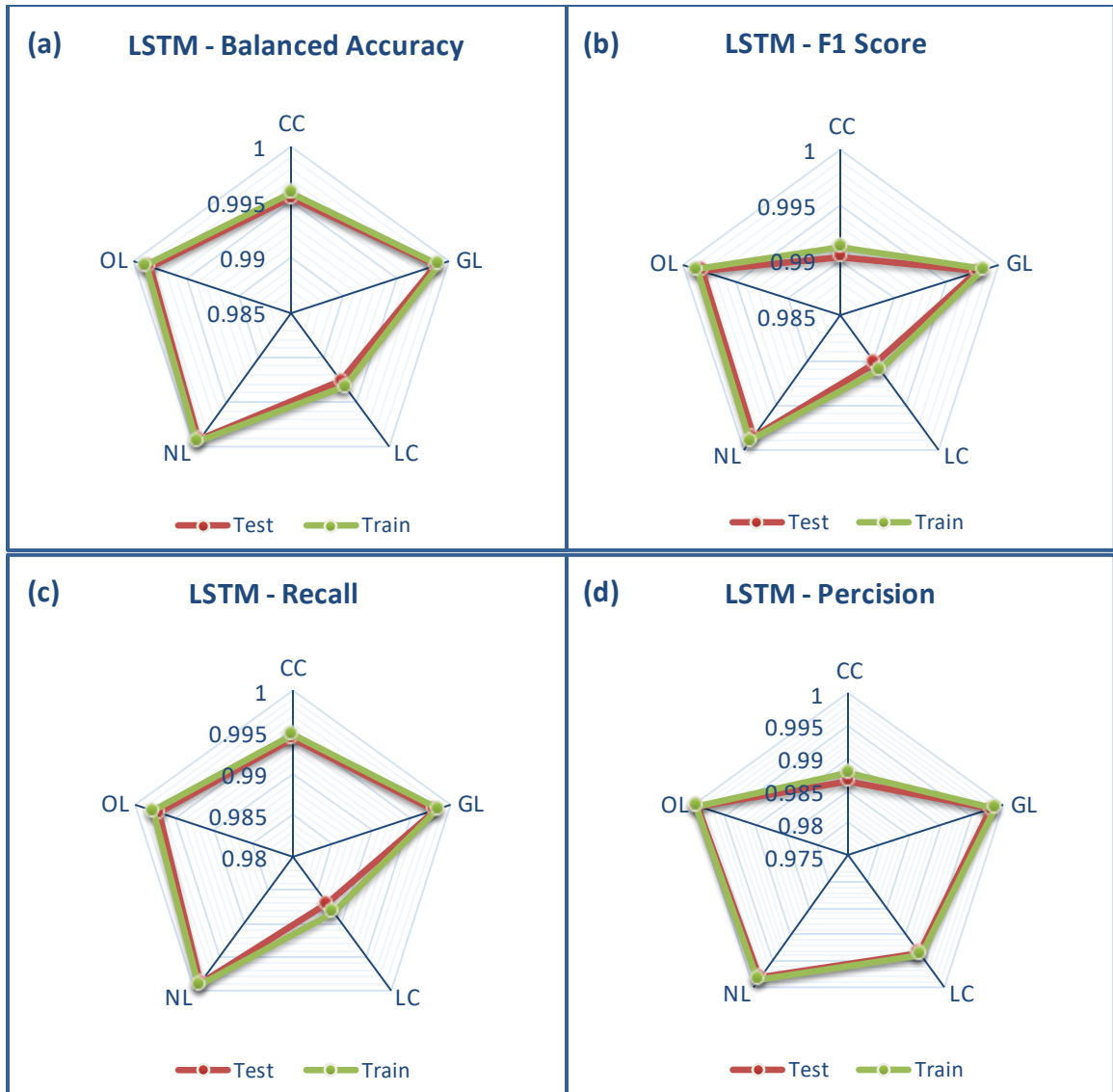


Figure 4.50: Radar charts illustrating LSTM model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The Transformer model demonstrates varying levels of performance in leak assessment across different network configurations. Considering Table 4.40, The specificity values are consistently high, particularly for minor and moderate leaks, indicating that the model effectively excludes negative cases. For instance, in the LO_GL network, the specificity for minor leaks reaches 0.9781, ensuring that normal cases are not misclassified as leaks. However, recall values for minor leaks are notably lower,

particularly in the LO_LC configuration, where the recall drops to 0.3954, implying a significant proportion of minor leaks are being overlooked. The F1-scores also reflect this trend, with severe leaks generally achieving higher F1 values due to a balanced precision-recall trade-off. The balanced accuracy values for severe leaks remain the highest across all configurations, reaching 0.9081 in LO_OL, while minor leaks exhibit the lowest values, such as 0.6752 in LO_LC. These findings suggest that while the Transformer model performs well in distinguishing severe leaks, it struggles to effectively classify minor leaks, a pattern observed in previous models.

The confusion matrices provide deeper insights into the classification tendencies of the Transformer model. In Figure 4.51a, a significant misclassification issue arises where 21,892 minor leaks are predicted as moderate, and 8,774 are classified as severe. This indicates that the model is not sufficiently differentiating small pressure variations associated with minor leaks. In Figure 4.51b, the misclassification pattern persists, with a considerable number of minor cases (16,578) classified as moderate or severe. Conversely, severe leaks are identified with high reliability, with only a small fraction being misclassified. The LO_LC confusion matrix (Figure 4.51c) further supports this trend, where the number of correctly classified minor leaks is significantly lower than those misclassified into higher severity classes. The LO_OL matrix (Figure 4.51d) also exhibits a high rate of misclassification for minor leaks, with 12,832 cases labeled as moderate and 6,125 as severe. This systematic pattern suggests that the Transformer model, while excelling in severe leak detection, lacks sensitivity to the smaller fluctuations associated with minor leaks.

Table 4.40: By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

RNN Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9424	0.8344	0.7865	0.5316	0.6344	0.7370
	moderate	0.7864	0.8337	0.6454	0.7125	0.6773	0.7495
	severe	0.7857	0.8609	0.6721	0.7758	0.7203	0.7807
LO_LC	minor	0.9555	0.8522	0.6876	0.3717	0.4826	0.6636
	moderate	0.8251	0.7552	0.6583	0.5574	0.6037	0.6913
	severe	0.7072	0.9587	0.6984	0.9570	0.8075	0.8321
LO_GL	minor	0.9651	0.8324	0.9032	0.6262	0.7396	0.7956
	moderate	0.9990	0.9884	0.9402	0.5790	0.7167	0.7890
	severe	0.6367	0.9259	0.8202	0.9702	0.8889	0.8034
LO_OL	minor	0.9581	0.8094	0.7761	0.3916	0.5206	0.6749
	moderate	0.8254	0.8365	0.6480	0.6659	0.6568	0.7456
	severe	0.7557	0.9562	0.7246	0.9490	0.8217	0.8523

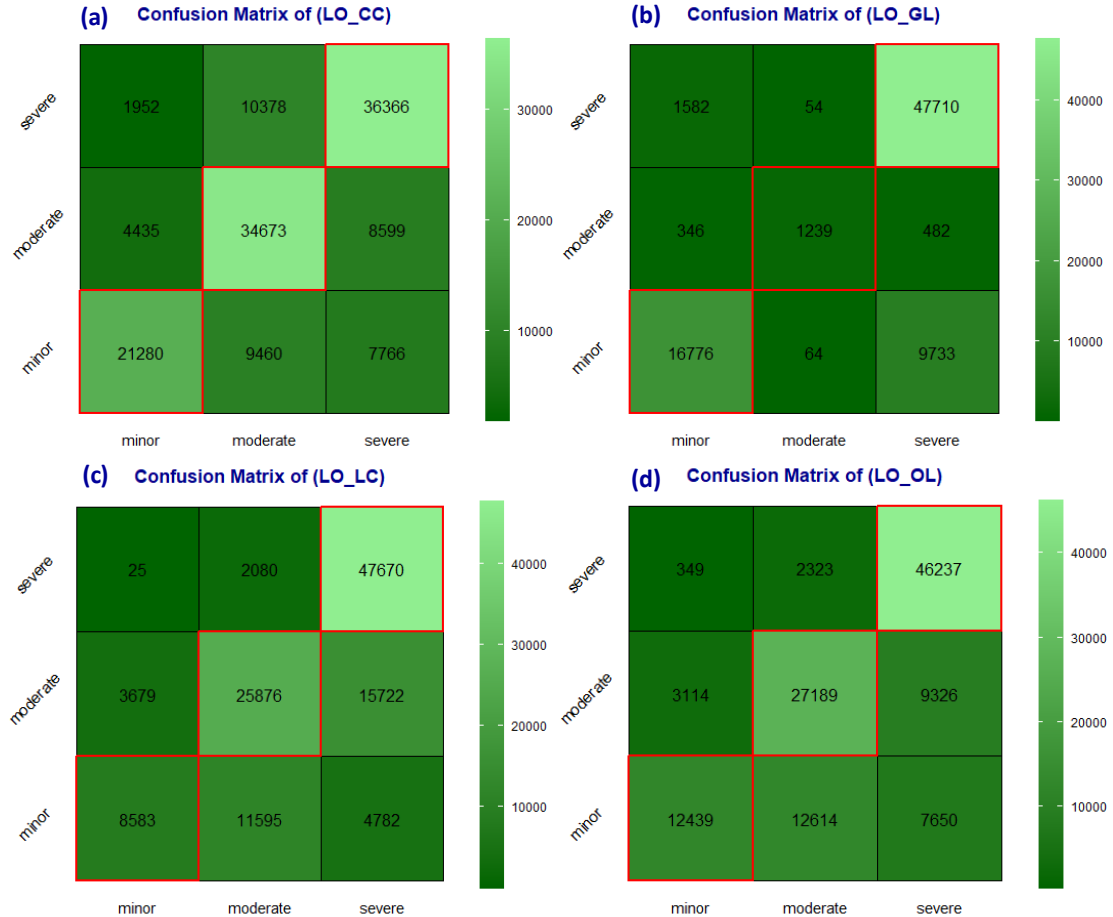


Figure 4.51: Confusion matrices for four Looped network RNN models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

The bar charts representing key performance metrics across different severity classes further reinforce the patterns observed in Table 4.40 and Figures 63. Balanced accuracy remains consistently high for severe leaks in all configurations, reflecting the model's strong ability to distinguish critical failures. The F1-scores for minor leaks are considerably lower compared to severe leaks, aligning with the recall issues identified in the confusion matrices. Precision values remain relatively stable across all severity levels, indicating that when the model predicts a given class, it does so with reasonable confidence. However, the recall values for minor leaks show significant drops, particularly in LO_LC and LO_GL, underscoring the challenge of accurately capturing minor leak patterns. This suggests that further refinement in feature representation or data resampling techniques may be necessary to enhance the detection of minor leaks in looped networks. Overall, while the Transformer model offers robust performance in detecting severe failures, its limited capability in capturing minor leaks highlights a critical area for improvement in real-world leak detection applications.

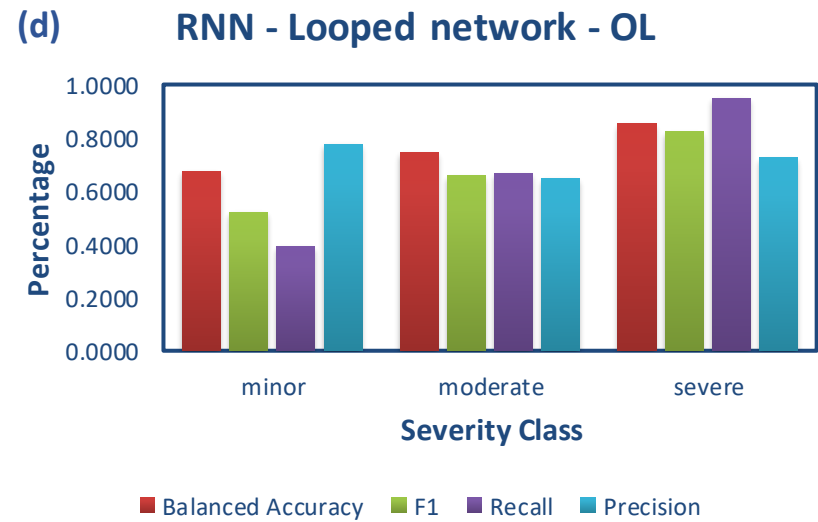
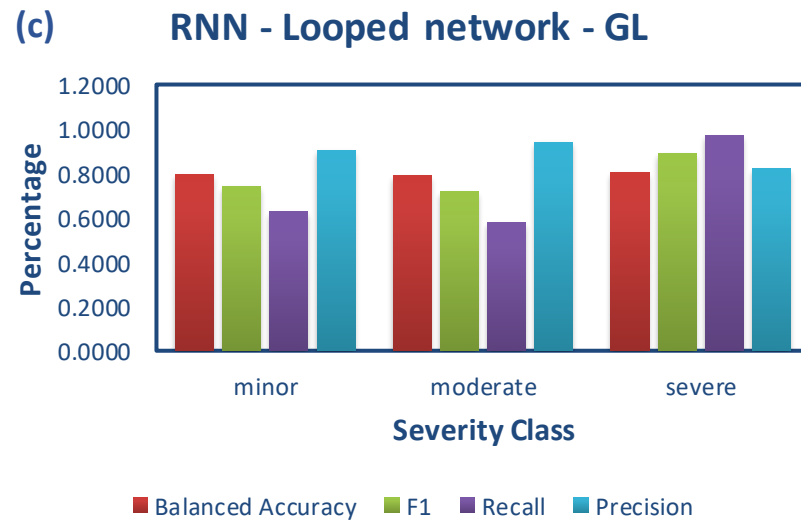
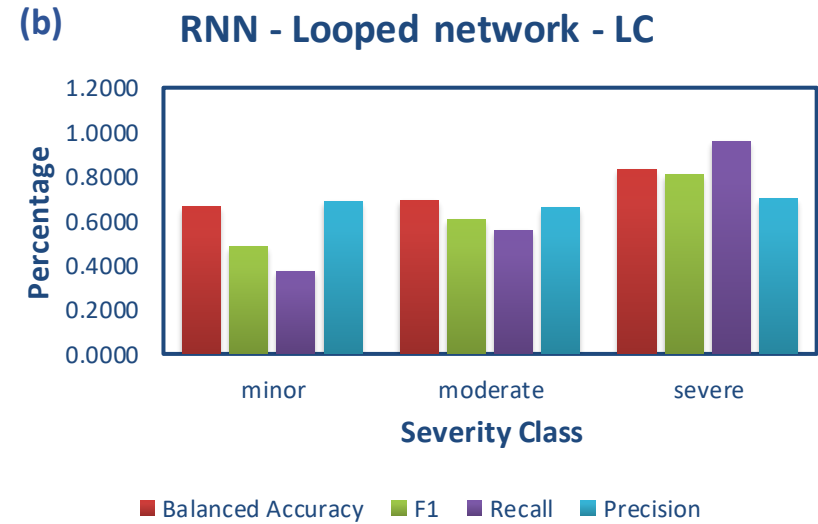
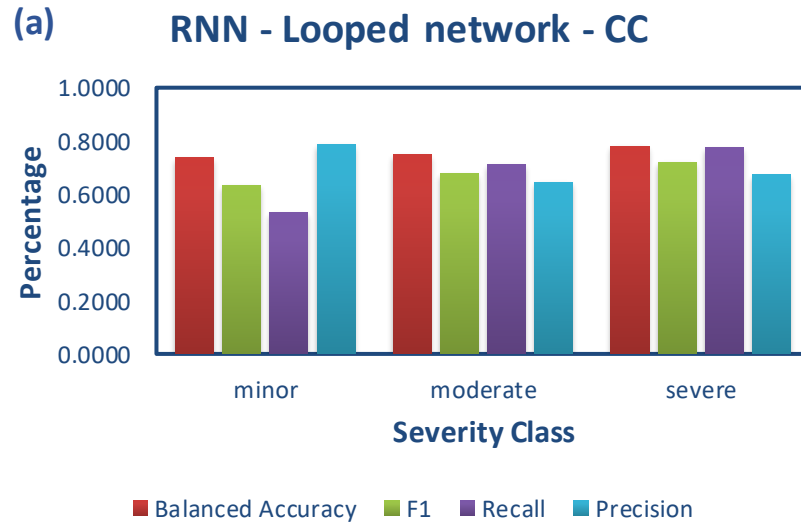


Figure 4.52: Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

The RNN model demonstrated strong performance in leakage detection task in the looped WDS network, achieving consistently high scores across all metrics. Table 4.41 shows that Balanced accuracy exceeded 0.995 for most classes, with CC and GL reaching 0.998+, indicating robust detection capabilities. Precision remained high, with values ranging between 0.992 and 0.999, ensuring minimal false positives. Recall followed a similar trend, exceeding 0.99 for most cases, ensuring strong leak identification.

While training and test results closely aligned, as shown in Figure 4.53, the slight gap in recall for NL and OL suggests minor generalization differences. The radar charts confirm these insights, showing an overall well-calibrated model. These findings highlight RNN's effectiveness in identifying leaks in a looped WDS, reinforcing its suitability for real-world applications.

Table 4.41: By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99953	0.99960	0.99812	0.99842	0.99827	0.99898
	GL	0.99972	0.99986	0.99886	0.99944	0.99915	0.99958
	LC	0.99955	0.99977	0.99820	0.99910	0.99865	0.99933
	NL	0.99888	0.99980	0.99554	0.99920	0.99736	0.99904
	OL	0.99994	0.99858	0.99976	0.99430	0.99702	0.99712
Train	CC	0.99974	0.99969	0.99895	0.99876	0.99885	0.99925
	GL	0.99985	0.99990	0.99941	0.99960	0.99950	0.99972
	LC	0.99958	0.99987	0.99832	0.99949	0.99890	0.99953
	NL	0.99914	0.99988	0.99659	0.99951	0.99804	0.99932
	OL	0.99996	0.99894	0.99985	0.99575	0.99780	0.99786

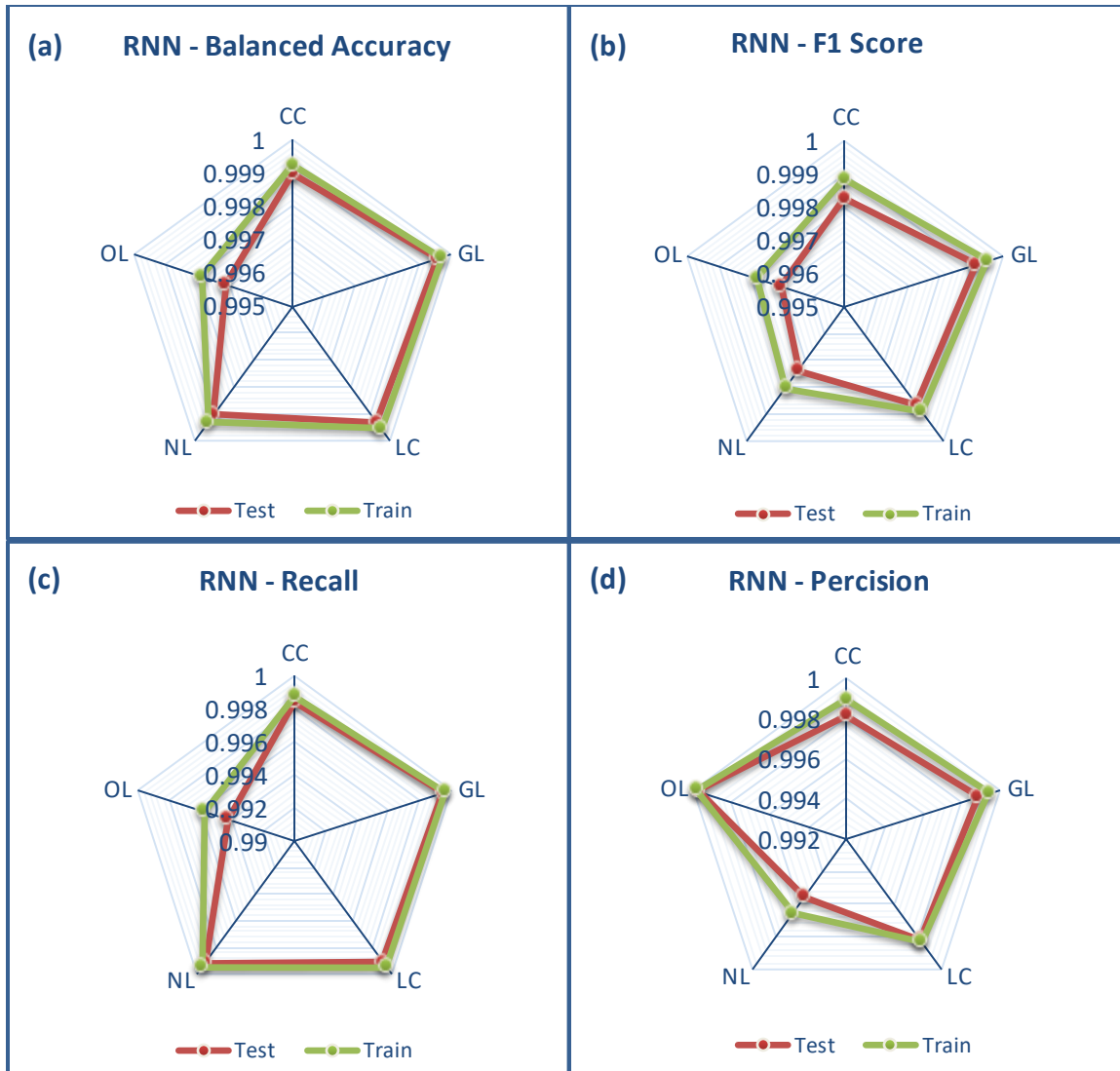


Figure 4.53: Radar charts illustrating RNN model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The GRU model exhibits a varied performance across different network configurations in classifying severity levels tabulated in Table 4.42. Specificity values remain consistently high, particularly for minor and moderate severity cases, demonstrating the model's capability in correctly identifying non-severe conditions. For instance, in the LO_GL network, the specificity for minor severity reaches 0.9587, indicating strong exclusion of false positives. However, recall values for minor severity

cases tend to be significantly lower. This trend is most apparent in the LO_LC configuration, where recall for minor severity drops to 0.4126, highlighting a considerable proportion of minor cases being misclassified. Similarly, F1-scores reflect this discrepancy, with severe cases achieving the highest values due to a balanced trade-off between precision and recall. The balanced accuracy for severe cases remains the highest across all configurations, reaching 0.8515 in LO_OL, whereas minor severity cases exhibit lower values, such as 0.6810 in LO_LC. These findings suggest that while the GRU model excels in distinguishing severe cases, it struggles with minor classifications, mirroring challenges identified in earlier models.

The confusion matrices provide a deeper understanding of the misclassification patterns of the GRU model. In Figure 4.54a, 22,117 minor cases are misclassified as moderate, and 8,113 are categorized as severe, emphasizing the model's difficulty in differentiating small variations in severity. In Figure 4.54b, this pattern persists, with 16,712 minor cases being classified as moderate or severe, while severe cases are reliably detected with minimal misclassification. The confusion matrix in Figure 4.54c further illustrates the challenge, with a high misclassification rate among minor severity cases, where 9,872 instances are reassigned to moderate severity. The LO_OL confusion matrix (Figure 4.54d) follows a similar trend, where 15,626 minor severity cases are incorrectly classified as moderate, and 6,706 are misclassified as severe. These systematic errors indicate that while the GRU model excels in identifying severe cases, it lacks the sensitivity required to accurately capture minor severity conditions, leading to frequent overestimation of severity.

Table 4.42: By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over looped network after removing the outliers.

GRU Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9228	0.8430	0.7467	0.5696	0.6462	0.7462
	moderate	0.7985	0.8252	0.6514	0.6900	0.6701	0.7443
	severe	0.7910	0.8534	0.6732	0.7602	0.7140	0.7756
LO_LC	minor	0.9429	0.8603	0.6595	0.4193	0.5126	0.6811
	moderate	0.7911	0.7707	0.6385	0.6105	0.6242	0.7008
	severe	0.7742	0.9196	0.7394	0.9045	0.8136	0.8393
LO_GL	minor	0.9589	0.8354	0.8895	0.6365	0.7420	0.7977
	moderate	0.9986	0.9882	0.9182	0.5720	0.7049	0.7853
	severe	0.6467	0.9124	0.8233	0.9636	0.8879	0.8051
LO_OL	minor	0.9242	0.8267	0.7003	0.4775	0.5679	0.7009
	moderate	0.8373	0.8240	0.6514	0.6295	0.6403	0.7334
	severe	0.7831	0.9353	0.7419	0.9200	0.8214	0.8516

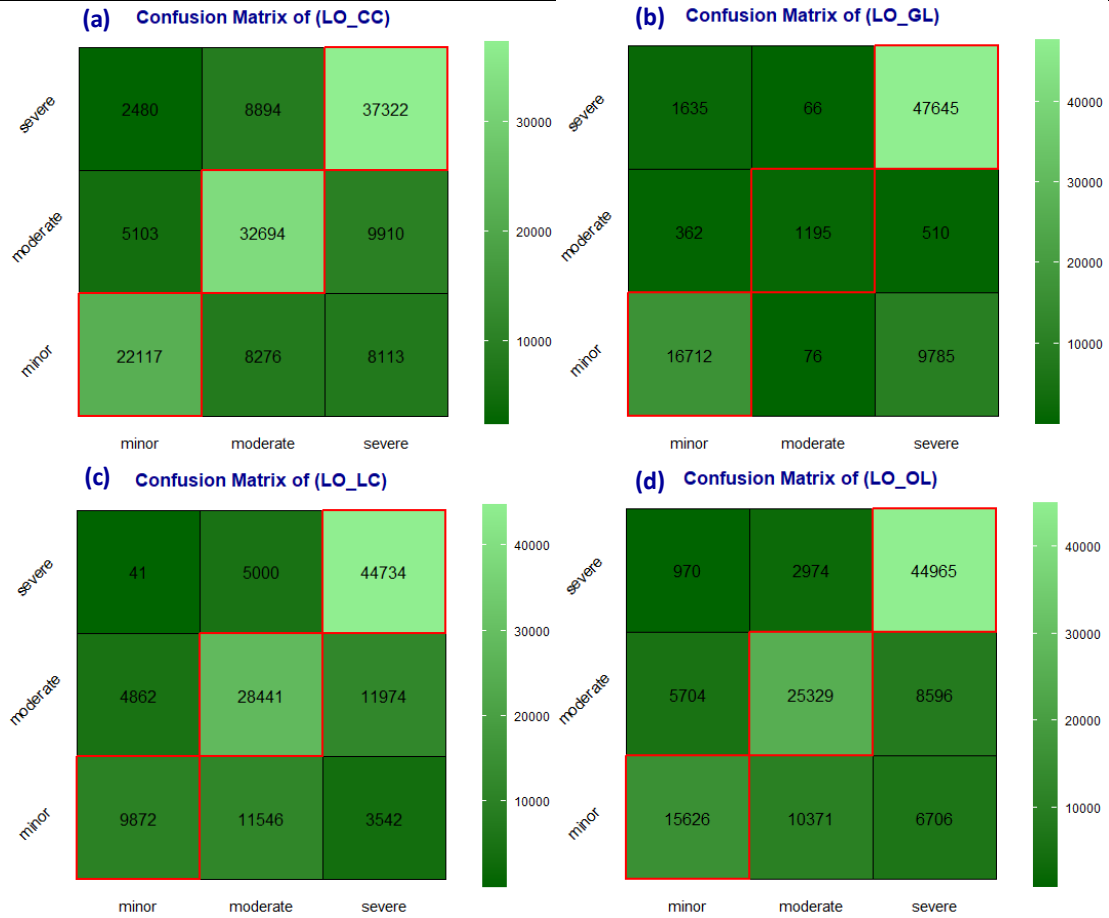


Figure 4.54: Confusion matrices for four Looped network GRU models, providing a granular view of classification accuracy in severity assessment (a) LO_CC (b) LO_GL (c) LO_LC (d) LO_OL.

The bar charts visualizing key performance metrics across different severity levels further reinforce the trends observed in Table 4.42 and Figure 4.55. Balanced accuracy remains consistently high for severe cases across all datasets, highlighting the model's robust ability to identify critical failures. However, F1-scores for minor severity remain considerably lower than those for severe cases, aligning with the recall issues identified in the confusion matrices. Precision values, on the other hand, remain relatively stable across severity classes, suggesting that when the model predicts a class, it does so with confidence. Nevertheless, recall for minor cases experiences significant drops, particularly in LO_LC and LO_GL, underlining the challenge of accurately capturing minor severity fluctuations. This suggests that further refinement in feature selection, improved data augmentation, or resampling techniques may be required to enhance minor severity detection in looped networks.

(a) GRU - Looped network - CC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(b) GRU - Looped network - LC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(c) GRU - Looped network - GL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(d) GRU - Looped network - OL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

Figure 4.55: Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task after outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

Overall, the GRU model demonstrates strong performance in identifying severe cases, but its inability to consistently capture minor severity conditions remains a key limitation, warranting further improvements for real-world severity assessment applications.

The results presented in Table 4.43 and visualized in Figure 4.56 demonstrate the performance of the GRU model applied to the Looped network for leakage detection. The model exhibits exceptional accuracy across all classes, with Balanced Accuracy values exceeding 0.995, indicating highly reliable predictions. The F1 scores, which measure the balance between Precision and Recall, remain consistently above 0.993, showcasing the model's robustness in identifying leaks with minimal false positives and false negatives. The Precision values, ranging between 0.994 and 0.999, further validate the model's ability to make precise classifications, ensuring a low rate of incorrect leak detections.

The radar plots in Figure 4.56 illustrate the near-identical performance of the model on both training and test datasets, indicating minimal overfitting. The Recall values, consistently above 0.99, confirm the model's high sensitivity, ensuring that actual leaks are detected with remarkable accuracy. This is particularly critical in real-world scenarios where undetected leaks could lead to significant resource loss. While slight variations are observed between different leak classes, the overall performance remains nearly perfect across all evaluation metrics, reinforcing the GRU model's effectiveness in handling the complexities of the Looped network structure.

Table 4.43: By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over looped network after removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99916	0.99862	0.99661	0.99446	0.99554	0.99681
	GL	0.99965	0.99982	0.99860	0.99928	0.99894	0.99947
	LC	0.99859	0.99927	0.99438	0.99708	0.99573	0.99784
	NL	0.99943	0.99979	0.99772	0.99916	0.99844	0.99930
	OL	0.99993	0.99926	0.99970	0.99702	0.99836	0.99847
Train	CC	0.99936	0.99878	0.99744	0.99511	0.99627	0.99723
	GL	0.99976	0.99987	0.99904	0.99950	0.99927	0.99963
	LC	0.99872	0.99940	0.99488	0.99761	0.99624	0.99816
	NL	0.99960	0.99988	0.99839	0.99951	0.99895	0.99955
	OL	0.99996	0.99946	0.99982	0.99786	0.99884	0.99891

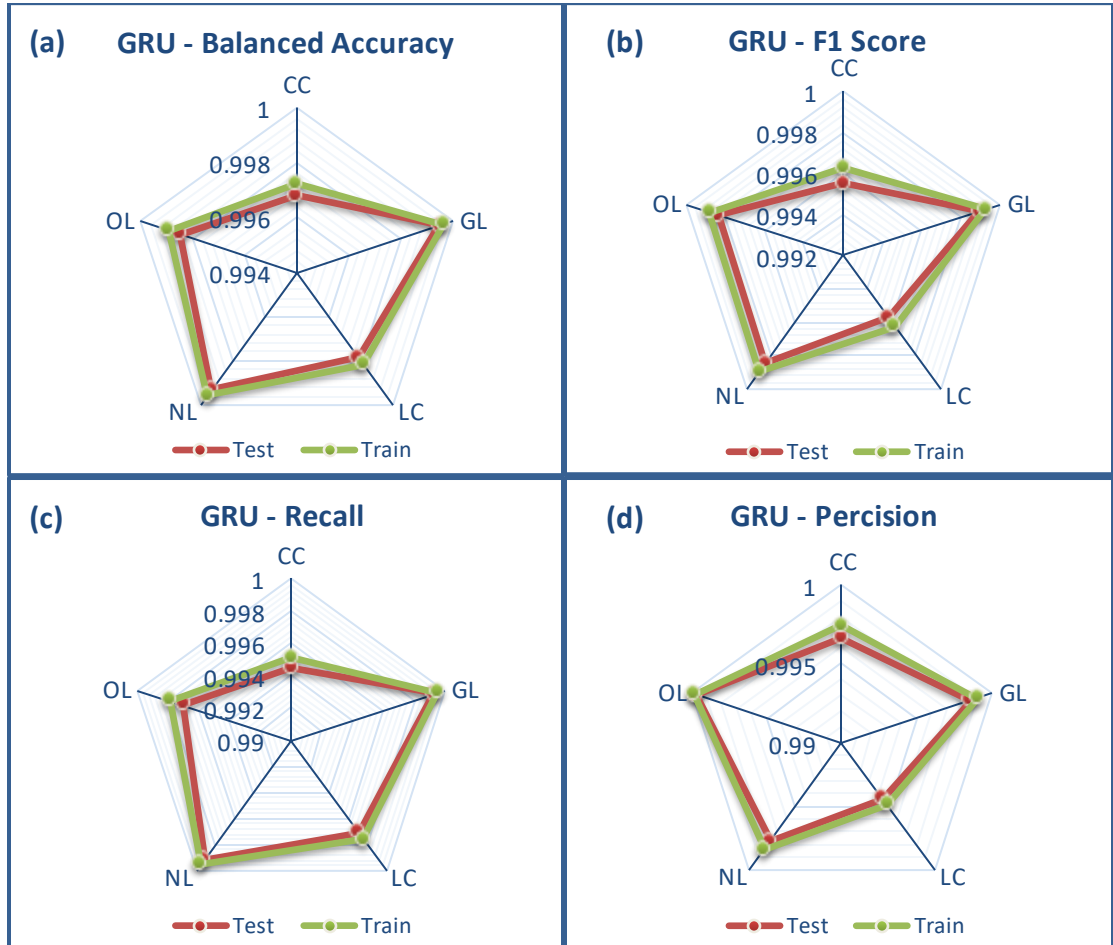


Figure 4.56: Radar charts illustrating GRU model performance after removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

In summary, the GRU-based approach provides a highly reliable framework for leakage detection in Looped networks, achieving near-perfect classification performance. The alignment between training and test scores suggests strong generalization, making this model suitable for real-world deployment with minimal risk of false alarms or undetected leaks. These results highlight the potential of GRU networks in enhancing water distribution system monitoring, ensuring efficient and accurate leak detection in large-scale infrastructure.

4.3.2 With outliers

4.3.2.1 Tree-based models

Based on the comparison of Table 4.30 and Table 4.44 for looped networks (XGB, RF, and LightGBM), it is evident that outlier removal has a mixed impact on performance metrics. For XGB, the accuracy for the LO_GL metric decreased from 0.92738667 to 0.8634335, and the Kappa score dropped from 0.89108 to 0.6999319, indicating a significant decline in performance. Similarly, the LO_OL metric experienced a reduction in accuracy from 0.81182 to 0.7857125 and a drop in Kappa from 0.71773 to 0.6691144. For RF, the LO_GL metric's accuracy showed a slight decrease from 0.93268667 to 0.871884 with Kappa reducing from 0.89903 to 0.7177369. Interestingly, the LO_OL metric maintained relatively stable accuracy at around 0.864 for both datasets, suggesting that outlier removal had a limited impact on this specific metric.

In the case of LightGBM, the performance was also affected but less drastically. For the LO_GL metric, accuracy dropped from 0.92665333 to 0.87391, and Kappa decreased from 0.88998 to 0.725469. The LO_OL metric saw a reduction in accuracy from 0.85871333 to 0.8442333 and in Kappa from 0.78807 to 0.7598265. The confidence

intervals narrowed in most cases, indicating a decrease in prediction variability. The graphical representations in Figure 4.37 and Figure 4.57 further reinforce these observations, showing more compact yet slightly lower performance bars post-outlier removal. This suggests that while the removal of outliers reduced variance, it also led to a moderate decline in peak performance for looped networks, particularly in metrics where the original performance was already high.

Table 4.44: Overall performance metrics of Tree-based models to evaluate their efficiency in severity assessment task over looped network before removing the outliers.

model	Metric	LO_CC	LO_LC	LO_GL	LO_OL
XGB	Accuracy	0.754873333	0.7845	0.92738667	0.81182
	Kappa	0.63231	0.67675	0.89108	0.71773
	CI Lower	0.752688205	0.78241042	0.92606182	0.80983268
	CI Upper	0.757048606	0.78657858	0.92869496	0.81379526
	No Info Rate	0.333333333	0.33333333	0.33333333	0.33333333
RF	Accuracy	0.83258	0.81173333	0.93268667	0.88411333
	Kappa	0.74887	0.7176	0.89903	0.82617
	CI Lower	0.830680864	0.80974567	0.93140694	0.88248273
	CI Upper	0.834466271	0.81370894	0.93394963	0.88572908
	No Info Rate	0.333333333	0.33333333	0.33333333	0.33333333
LightGBM	Accuracy	0.793313333	0.80491333	0.92665333	0.85871333
	Kappa	0.68997	0.70737	0.88998	0.78807
	CI Lower	0.791255156	0.80289876	0.92532241	0.85694037
	CI Upper	0.795360168	0.80691611	0.92796773	0.86047242
	No Info Rate	0.333333333	0.33333333	0.33333333	0.33333333

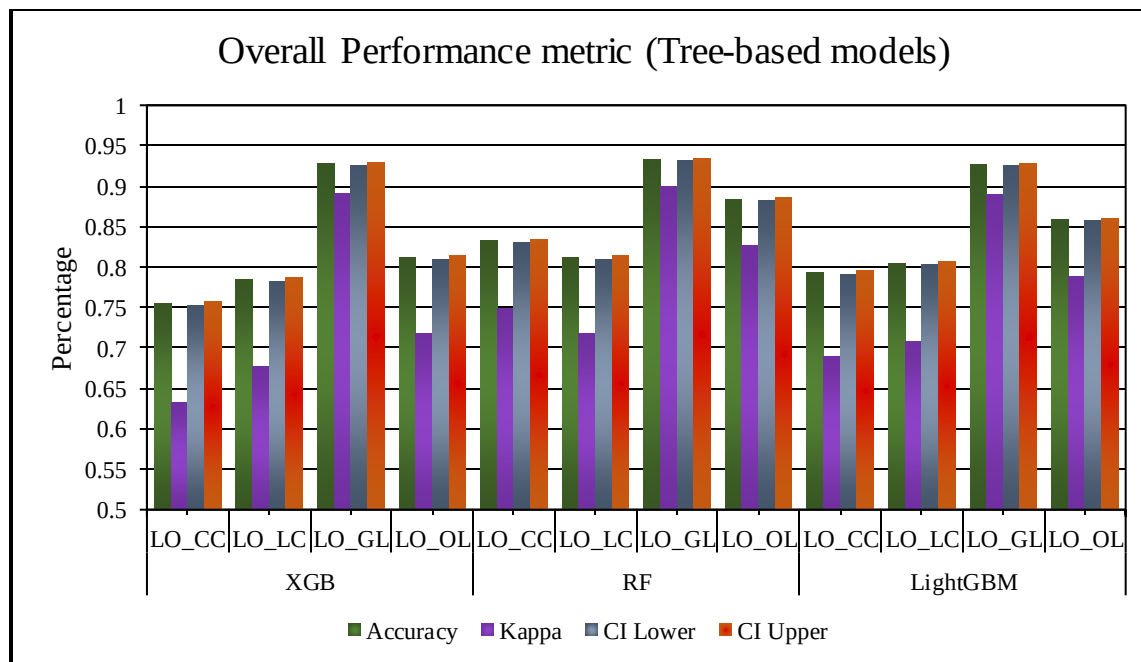


Figure 4.57: Bar chart visualization of overall performance metrics of Tree-based models to visually compare their efficiency in severity assessment task over looped network before removing the outliers.

The results from Table 4.45, which includes outliers, reveal significant shifts in the assessment performance of the Random Forest (RF) model when compared to Table 4.31, where outliers were removed. Across all looped network configurations (LO_CC, LO_LC, LO_GL, and LO_OL), specificity remained consistently high, particularly for severe and moderate cases. Minor class specificity ranged from 0.937 in LO_LC to 0.967 in LO_OL, demonstrating that the model effectively distinguishes negative cases. However, recall values for minor cases exhibited notable variations after incorporating outliers. One of the most striking improvements occurred in LO_GL and LO_OL, where minor class recall increased significantly. In LO_GL, recall improved from 0.514 (Table 4.31) to 0.835 (Table 4.45), and in LO_OL, recall rose from 0.540 to 0.861. These improvements suggest that outlier inclusion has allowed the model to recognize a more diverse range of minor leaks, thereby reducing misclassification into higher severity categories. Conversely, recall

values for moderate and severe cases remained relatively stable, with minor fluctuations. For instance, in LO_CC, severe class recall changed marginally from 0.935 to 0.936, indicating that the model's ability to detect critical failures was preserved despite the presence of outliers. However, precision values saw slight declines in some cases. In LO_LC, severe class precision dropped from 0.857 to 0.813. This suggests that while the model improved in detecting minor leaks, it generated slightly more false positives in severe cases, a known trade-off when including outliers.

Table 4.45: By class performance metrics of RF models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

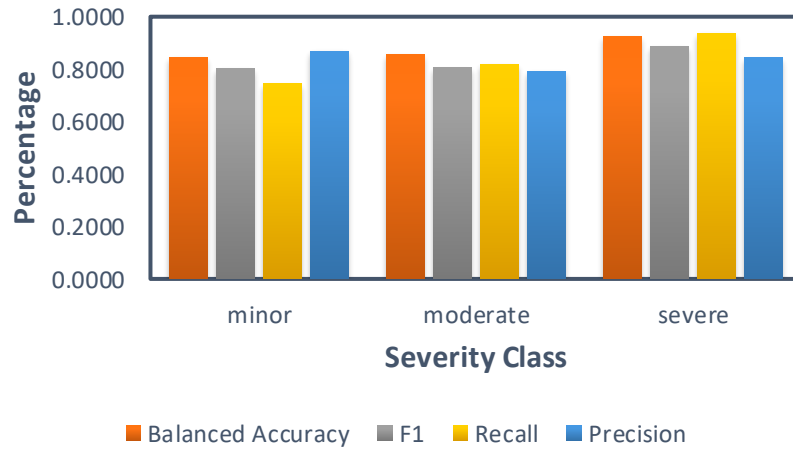
RF Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9440	0.8804	0.8692	0.7436	0.8015	0.8438
	moderate	0.8922	0.9077	0.7916	0.8186	0.8049	0.8554
	severe	0.9126	0.9659	0.8426	0.9356	0.8866	0.9241
LO_LC	minor	0.9376	0.8855	0.8586	0.7575	0.8049	0.8476
	moderate	0.8904	0.8626	0.7658	0.7164	0.7402	0.8034
	severe	0.8895	0.9787	0.8131	0.9613	0.8810	0.9254
LO_GL	minor	0.9867	0.9228	0.9691	0.8350	0.8971	0.9108
	moderate	0.9987	0.9908	0.9974	0.9815	0.9894	0.9901
	severe	0.9136	0.9900	0.8503	0.9816	0.9112	0.9476
LO_OL	minor	0.9679	0.9085	0.9262	0.8051	0.8614	0.8865
	moderate	0.9409	0.9302	0.8790	0.8589	0.8688	0.8999
	severe	0.9173	0.9937	0.8567	0.9883	0.9178	0.9528

The bar charts in Figure 4.58 further validate these findings. The most notable improvement is observed in LO_GL and LO_OL, where recall for minor cases increased significantly. Previously, LO_GL's minor class recall was one of the lowest at 0.514 in Table 4.31 but improved to 0.835 in Table 4.45, suggesting that the model is now better at capturing the subtle characteristics of minor leaks. Similarly, LO_OL's minor recall rose from 0.540 to 0.861, demonstrating improved detection of small pressure variations. These

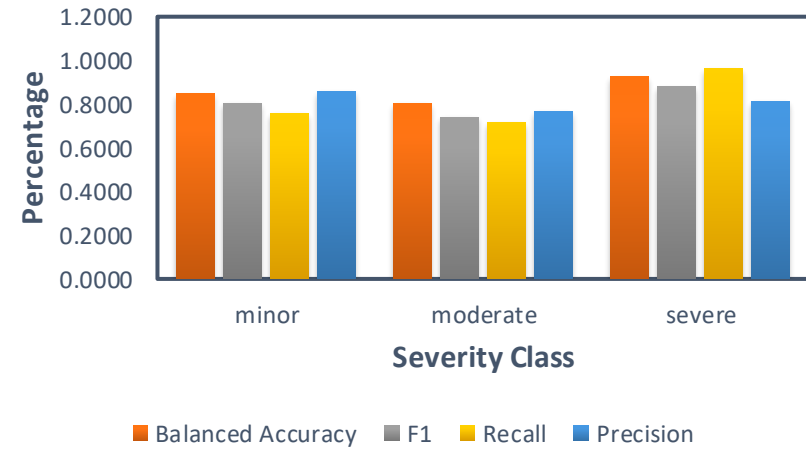
results indicate that outliers contributed valuable information, particularly in identifying minor leaks that were previously overlooked.

However, Figure 4.58 also highlights a trade-off. While balanced accuracy for severe cases remained high, precision values declined slightly, indicating an increase in false positives. This trend is evident in LO_CC and LO_LC, where severe class precision dropped from 0.857 to 0.813 in LO_LC. This suggests that including outliers made the model more sensitive to detecting minor leaks but at the cost of some classification precision for severe cases.

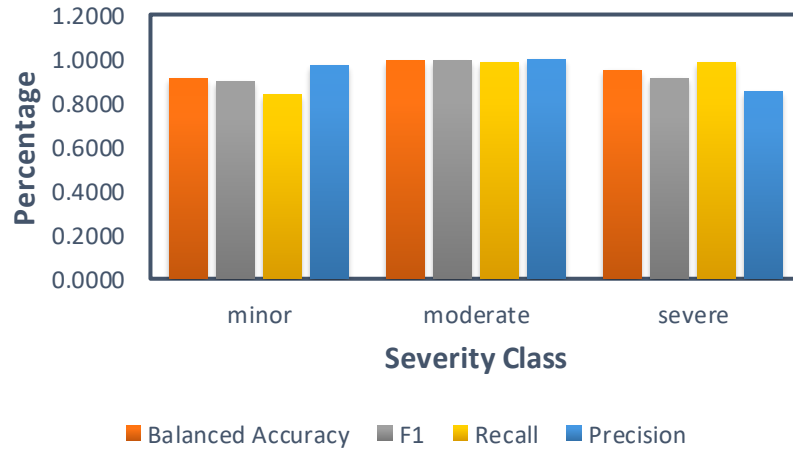
(a) Random Forest - Looped network - CC



(b) Random Forest - Looped network - LC



(c) Random Forest - Looped network - GL



(d) Random Forest - Looped network - OL

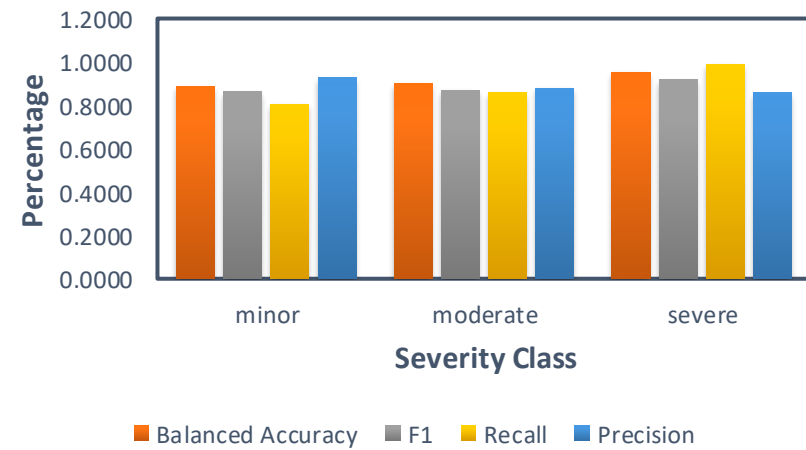


Figure 4.58: Bar charts presenting classification performance of RF model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

The observed performance changes align with typical ML behavior in real-world datasets, where outliers can contain critical but difficult-to-model information. In practical water distribution systems, outliers can originate from sensor anomalies, transient pressure fluctuations, or genuine rare but significant leak events. Including them in the training process can enhance generalizability, explaining the improvement in minor class recall for LO_GL and LO_OL. However, the impact of outliers remains highly context-dependent. If outliers represent true extreme cases rather than sensor noise, their inclusion enhances robustness, as seen in the improved minor class recall. Conversely, if outliers are mostly sensor errors or inconsistencies, they introduce bias and reduce classification precision, as seen in the slight drop in severe class precision. The decision to include outliers should therefore depend on data reliability and the trade-off between sensitivity and precision.

Training the RF models with the full dataset, including outliers, led to substantial improvements in minor class recall while maintaining strong performance for moderate and severe cases. The most significant enhancements were observed in LO_GL and LO_OL, where minor class recall increased by over 30%. However, this improvement came at a slight cost to precision, particularly for severe cases in LO_CC and LO_LC.

The performance of the RF model for leak detection in the Looped WDS before removing outliers, as reflected in Table 4.46 and Figure 4.59, presents some notable differences compared to the results after outlier removal. While Balanced Accuracy remains high for some leak types, such as CC (0.9976 in validation and 0.9977 in test) and GL (0.9986 in validation and 0.9975 in test), a marked decline is observed for OL, where it drops to 0.9829 in validation and 0.9994 in test, as opposed to the lower values noted in Table 4.32 after outlier removal. Similarly, the F1 Score follows this pattern, with CC

reaching 0.9959 in validation and 0.9963 in test, but OL showing a relatively lower F1 Score of 0.9828 in validation and 0.9994 in test. These results indicate that before outlier removal, the model exhibits more consistent but still imperfect classification of OL leaks, while other leak types maintain high predictive reliability. Additionally, Precision and Recall metrics in Table 4.46 for LC and NL are slightly lower than their counterparts in Table 4.32, suggesting that the presence of outliers may have contributed to minor misclassifications or label noise affecting the overall confidence in these categories.

Table 4.46: By class performance metrics of RF models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99888	0.99909	0.99552	0.99635	0.99594	0.99761
	GL	0.99926	0.99947	0.99704	0.99789	0.99747	0.99858
	LC	0.99891	0.99887	0.99563	0.99549	0.99556	0.99720
	NL	0.99935	0.99978	0.99742	0.99912	0.99827	0.99924
	OL	0.99997	0.99915	0.99987	0.99661	0.99824	0.99829
Test	CC	0.99907	0.99911	0.99629	0.99643	0.99636	0.99775
	GL	0.99929	0.99949	0.99718	0.99797	0.99757	0.99863
	LC	0.99896	0.99895	0.99584	0.99581	0.99583	0.99739
	NL	0.99942	0.99987	0.99768	0.99947	0.99857	0.99944
	OL	0.99998	0.99931	0.99992	0.99723	0.99857	0.99860
Train	CC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	GL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	LC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	NL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	OL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

The radar charts in Figure 4.59 further illustrate these variations, emphasizing that outlier presence does not drastically degrade the model's general performance but does introduce inconsistencies in specific cases. In Figure 4.59a, Balanced Accuracy remains tightly clustered for CC and GL, but slight drops for LC and OL can be observed, particularly in the validation dataset. Figure 4.59b (F1 Score) and Figure 4.59c (Recall)

show strong agreement with the table, with most metrics hovering near 0.999 while NL and OL demonstrate more noticeable dips. The most pronounced difference appears in Figure 4.59d (Precision), where the test dataset shows a lower confidence level in leak type predictions, particularly for LC and OL, compared to the sharper decline observed in Figure 4.40d. Overall, while removing outliers in Table 4.32 and Figure 4.40 helped stabilize the classification of underrepresented leak types, Table 4.46 and Figure 4.59 show that retaining outliers does not significantly impact model robustness but does introduce fluctuations, particularly in OL classification. These findings reinforce the importance of data preprocessing, as removing outliers appears to enhance the consistency of classification, particularly for leak types that are more difficult to identify.

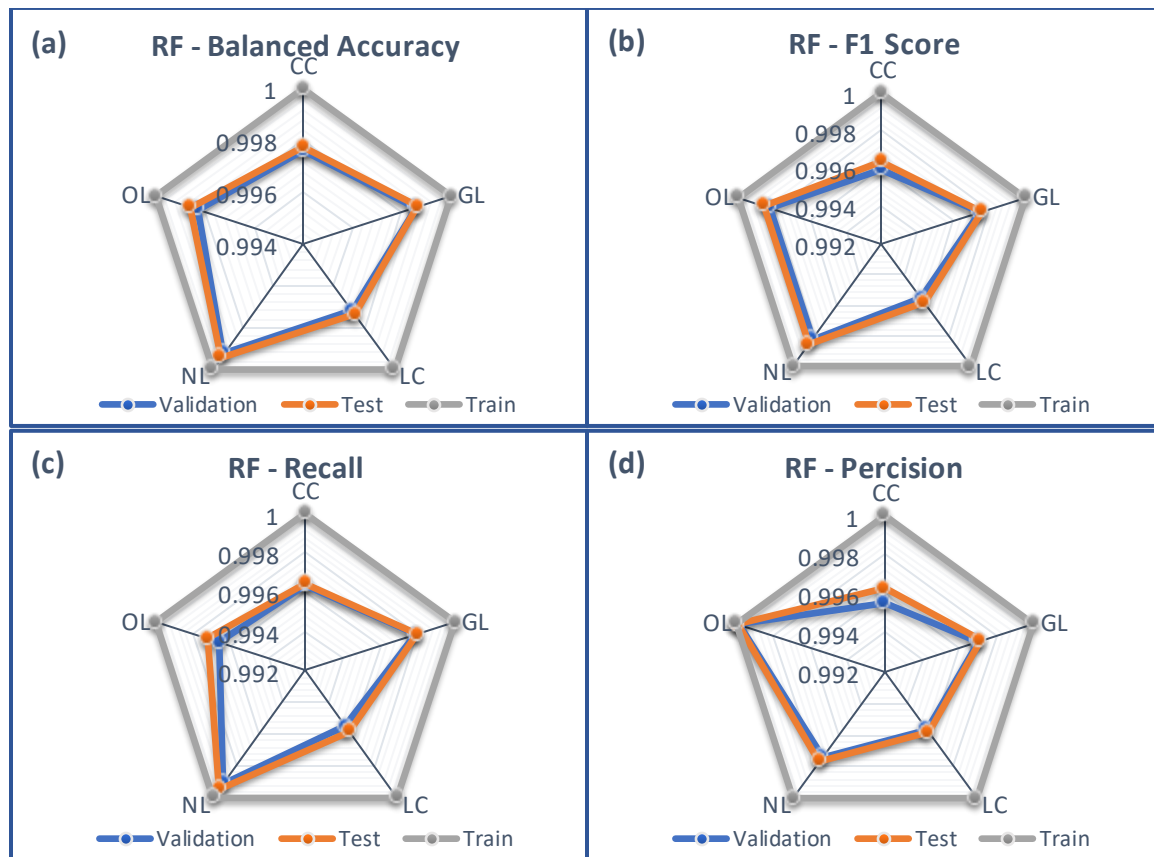


Figure 4.59: Radar charts illustrating RF model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The results in Table 4.47, where outliers are included, highlight noticeable shifts in LightGBM's classification performance when compared to Table 4.33, where outliers were removed. Across all looped network configurations (LO_CC, LO_LC, LO_GL, and LO_OL), specificity remained relatively high across all severity levels. For minor leaks, specificity values ranged from 0.940 in LO_OL to 0.969 in LO_GL, which confirms that the model is effective in correctly identifying negative cases. However, the inclusion of outliers introduces variations in recall values, particularly for minor leaks, which is a crucial factor in assessing the model's overall robustness.

A notable improvement is observed in minor class recall in LO_GL and LO_OL, where recall increased from 0.572 to 0.826 in LO_GL and from 0.621 to 0.772 in LO_OL. This suggests that the presence of outliers has allowed the model to capture a broader spectrum of minor leak patterns, leading to a higher rate of correct classifications. In contrast, moderate and severe class recall values remained relatively stable, with minor fluctuations. For example, severe class recall in LO_CC saw a minor shift from 0.947 in Table 4.33 to 0.951 in Table 4.47, reinforcing the model's reliability in detecting critical leak cases. However, the precision for severe classifications exhibited a slight decline, such as in LO_LC, where severe class precision decreased from 0.813 to 0.790. This suggests that while the model improved sensitivity for minor leaks, it also introduced a higher rate of false positives in severe classifications.

Table 4.47: By class performance metrics of LightGBM models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

LightGBM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9444	0.8765	0.8683	0.7339	0.7955	0.8391
	moderate	0.8672	0.8924	0.7486	0.7909	0.7692	0.8291
	severe	0.8784	0.9238	0.7786	0.8551	0.8151	0.8668
LO_LC	minor	0.9408	0.8866	0.8650	0.7593	0.8087	0.8500
	moderate	0.8885	0.8556	0.7585	0.7001	0.7281	0.7943
	severe	0.8742	0.9709	0.7902	0.9476	0.8618	0.9109
LO_GL	minor	0.9828	0.9187	0.9600	0.8261	0.8880	0.9045
	moderate	0.9943	0.9914	0.9886	0.9828	0.9857	0.9886
	severe	0.9125	0.9840	0.8472	0.9704	0.9046	0.9414
LO_OL	minor	0.9692	0.8951	0.9261	0.7727	0.8425	0.8710
	moderate	0.9214	0.9175	0.8414	0.8343	0.8378	0.8778
	severe	0.8975	0.9831	0.8255	0.9691	0.8915	0.9333

The bar charts in Figure 4.60 further confirm these trends. The most substantial improvement is evident in LO_GL and LO_OL, where the recall for minor cases increased significantly. Previously, LO_GL's minor class recall was among the lowest at 0.572 in Table 4.33, but it increased to 0.826 in Table 4.47, suggesting that outlier inclusion has enhanced the model's ability to detect small leaks that were previously misclassified. Similarly, LO_OL's minor class recall improved from 0.621 to 0.772, demonstrating better detection of subtle pressure fluctuations associated with minor leaks. These results indicate that outliers' removal caused ML models to favor some classes over others due to the imbalance occurred to the data where the majority of outliers were concentrated in the observations labeled as minor.

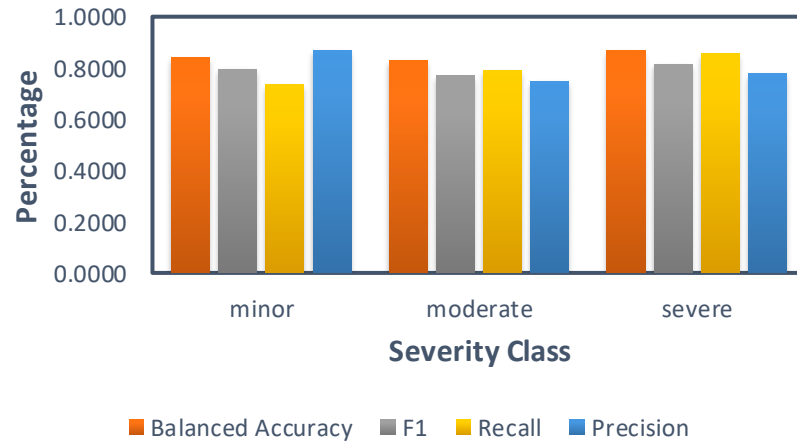
However, Figure 4.60 also highlights a critical trade-off. While the balanced accuracy for severe cases remained consistently high, precision experienced slight declines in certain models, indicating a slight increase in false positives. This is particularly noticeable in LO_CC and LO_LC, where precision for severe classifications dropped from

0.813 to 0.790 in LO_LC and from 0.778 to 0.748 in LO_CC. This trade-off suggests that the model, when trained with outliers, becomes more sensitive to detecting minor leaks but at the cost of some misclassifications in higher severity levels. Yet, the increase in minor recall was more significant than the decrease in severe precision.

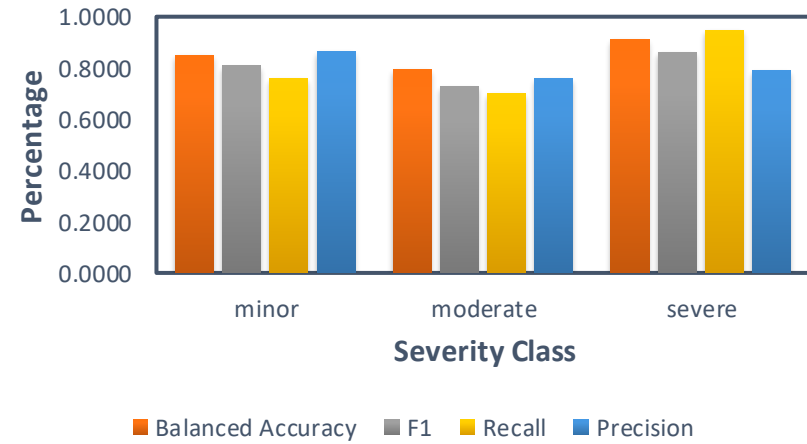
These findings align with expected trends in real-world ML applications, where outliers can introduce valuable but difficult-to-model information. In real-world leak detection systems, outliers often stem from sensor noise, transient pressure fluctuations, or rare but critical leak events. Their inclusion in training may enhance the model's ability to generalize, particularly for detecting small leaks, as evidenced by the recall improvements in LO_GL and LO_OL.

However, the impact of outliers remains context-dependent. If the outliers represent genuine extreme cases rather than sensor anomalies, their inclusion strengthens the model's robustness, as demonstrated by the increased recall for minor leaks. Conversely, if outliers predominantly consist of sensor errors or measurement inconsistencies, they may introduce unwanted noise, reducing precision and increasing false positives in severe classifications, as observed in LO_CC and LO_LC. Therefore, the decision to include outliers in model training should be made carefully, balancing the trade-offs between sensitivity and precision while considering data quality.

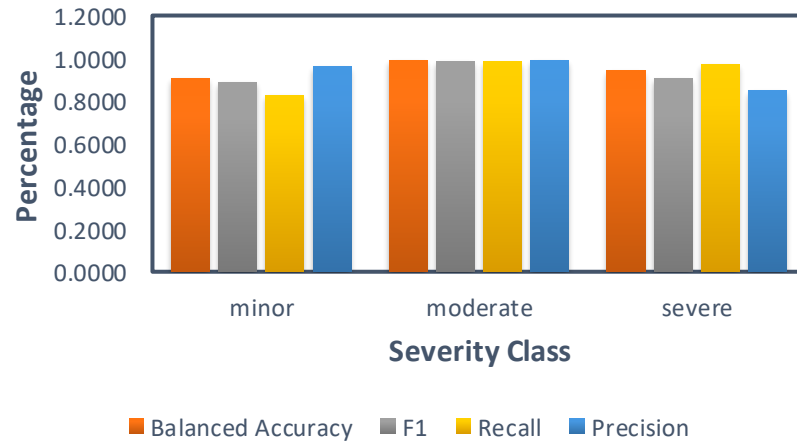
(a) LightGBM - Looped network - CC



(b) LightGBM - Looped network - LC



(c) LightGBM - Looped network - GL



(d) LightGBM - Looped network - OL

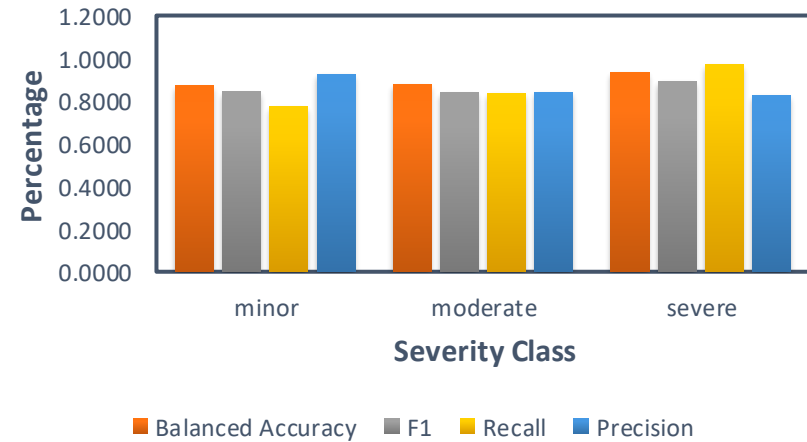


Figure 4.60: Bar charts presenting classification performance of LightGBM model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

The performance evaluation of the LightGBM model for leakage detection in the Looped WDS before removing outliers, as illustrated in Table 4.48 and Figure 4.61, highlights both strengths and challenges compared to the performance observed after outlier removal. While Balanced Accuracy remains high for most leak types, exceeding 0.995 in several cases such as CC (0.9956 in validation, 0.9956 in test) and GL (0.9992 in validation, 0.9977 in test), there are noticeable deviations when detecting OL, where Balanced Accuracy reaches 0.9993 in validation and 0.9994 in test. This represents a significant improvement over Table 4.34, where OL detection was more inconsistent. Similarly, F1 Scores exhibit strong performance across most classes, with CC achieving 0.9935 in validation and 0.9936 in test, but with some discrepancies in LC (0.9930 validation, 0.9923 test) and NL (0.9940 validation, 0.9951 test). These values are slightly lower than the nearly perfect scores observed in Table 4.34, suggesting that keeping outliers may have introduced small fluctuations in classification confidence. The overall trend suggests that while the LightGBM model generalizes well across datasets, the removal of outliers in Table 4.34 slightly stabilized performance across all leak types, particularly in challenging cases such as OL.

The radar charts in Figure 4.61 provide a visual representation of these discrepancies, reinforcing the numerical observations from Table 4.48. Figure 4.61a, displaying Balanced Accuracy, shows an overall high performance across all leak types but with some variation, particularly in LC and NL, compared to the more uniform distribution seen in Figure 4.43a. Figure 4.61b (F1 Score) and Figure 4.61c (Recall) indicate similar trends, with test set values deviating slightly for certain leak types, particularly NL, which displays minor fluctuations compared to the more consistent results in Figure 4.43b and

54c. The most noticeable impact of keeping outliers is observed in Figure 4.61d (Precision), where LC and OL show slightly more variations, reinforcing the earlier observation that outliers introduce inconsistencies in classification confidence. Despite these fluctuations, the LightGBM model maintains strong generalization capabilities before outlier removal, but Table 4.34 and Figure 4.43 confirm that eliminating outliers slightly refines the model's stability, particularly for underrepresented leak types. These findings suggest that while LightGBM remains highly effective in both cases, preprocessing steps such as outlier removal can provide small yet valuable improvements in classification uniformity, especially for challenging cases like OL detection in the Looped WDS.

Table 4.48: By class performance metrics of LightGBM models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99839	0.99838	0.99357	0.99352	0.99355	0.99596
	GL	0.99907	0.99984	0.99628	0.99936	0.99782	0.99921
	LC	0.99853	0.99800	0.99412	0.99200	0.99306	0.99527
	NL	0.99980	0.99990	0.99920	0.99960	0.99940	0.99970
	OL	1.00000	0.99967	1.00000	0.99869	0.99935	0.99935
Test	CC	0.99863	0.99817	0.99452	0.99269	0.99361	0.99566
	GL	0.99918	0.99987	0.99673	0.99947	0.99810	0.99932
	LC	0.99835	0.99827	0.99341	0.99307	0.99324	0.99571
	NL	0.99982	0.99993	0.99928	0.99973	0.99951	0.99978
	OL	0.99999	0.99974	0.99997	0.99896	0.99947	0.99948
Train	CC	0.99901	0.99886	0.99604	0.99546	0.99575	0.99723
	GL	0.99974	0.99994	0.99895	0.99976	0.99935	0.99975
	LC	0.99891	0.99885	0.99564	0.99541	0.99553	0.99716
	NL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
	OL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

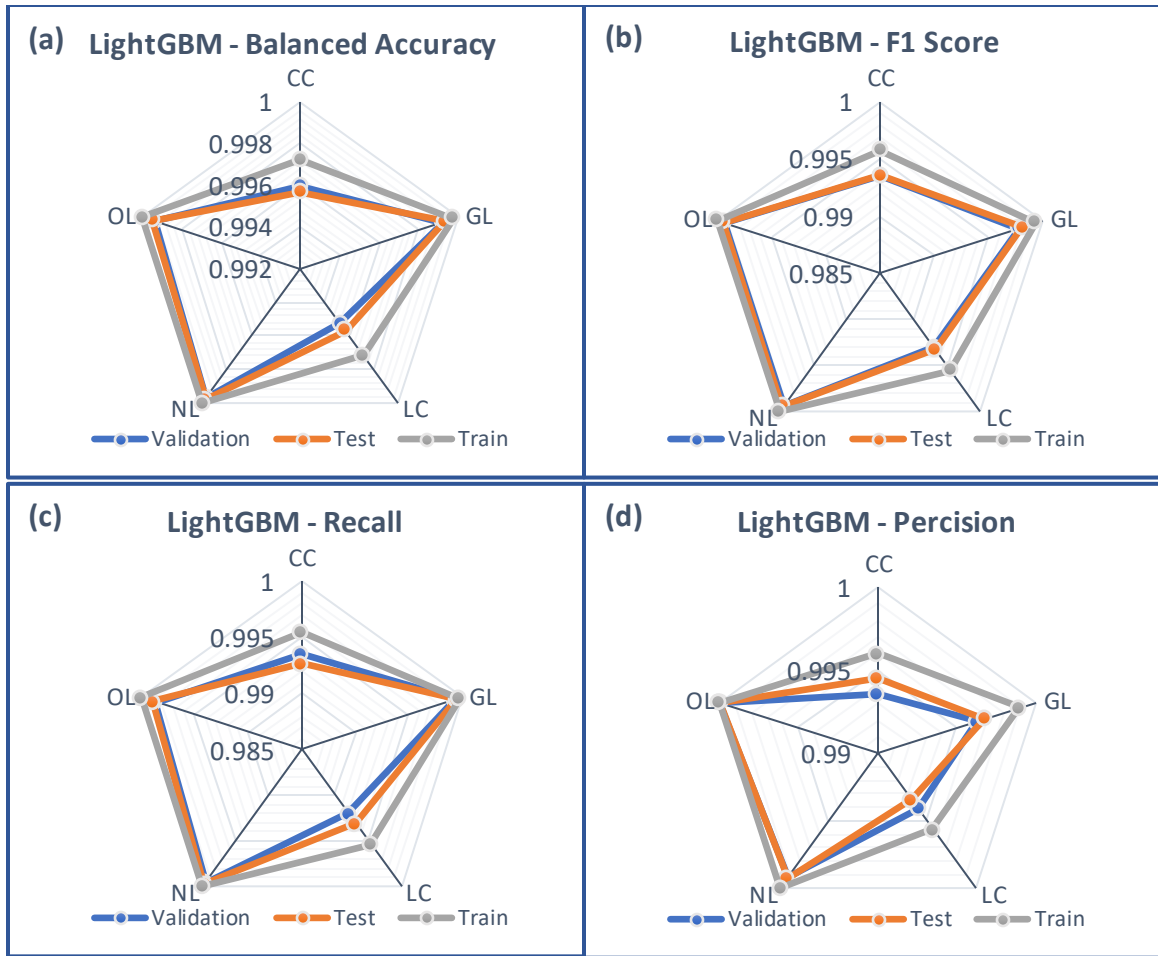


Figure 4.61: Radar charts illustrating LightGBM model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The XGB model demonstrates strong classification capabilities for severe leaks across different looped network configurations, as indicated in Table 4.49, which includes outliers. Similar to the observations in Table 4.35, the recall for severe cases remains consistently high, exceeding 0.94 in most configurations. This stability suggests that outlier inclusion does not significantly affect the model's ability to detect critical leak events. However, recall for moderate and minor leaks shows noticeable differences between the two datasets, indicating that the presence of outliers impacts the model's performance in distinguishing lower-severity cases.

A key improvement seen in Table 4.49 is the increased recall for minor leaks in certain configurations. For instance, in LO_GL, minor leak recall increased from 0.51 in Table 4.35 to 0.72 in Table 4.49, highlighting a more comprehensive detection of minor leaks when outliers are included. Similarly, LO_OL saw an improvement in minor recall from 0.58 to 0.72. These gains suggest that outliers may have introduced additional variability in minor leak patterns, helping the model capture previously overlooked cases. However, this improvement comes with a trade-off, as precision for minor leaks slightly decreased in some configurations, such as in LO_CC, where precision dropped from 0.84 in Table 4.35 to 0.83 in Table 4.49. This indicates a slight increase in false positives, where non-minor cases are mistakenly classified as minor.

Table 4.49: By class performance metrics of XGB models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

XGB Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9299	0.8640	0.8345	0.7072	0.7656	0.8185
	moderate	0.8439	0.8721	0.7068	0.7526	0.7289	0.7982
	severe	0.8585	0.8980	0.7399	0.8049	0.7710	0.8317
LO_LC	minor	0.9349	0.8823	0.8522	0.7505	0.7982	0.8427
	moderate	0.8758	0.8437	0.7312	0.6756	0.7023	0.7757
	severe	0.8660	0.9597	0.7758	0.9274	0.8448	0.8967
LO_GL	minor	0.9822	0.9193	0.9587	0.8275	0.8883	0.9048
	moderate	0.9988	0.9919	0.9975	0.9837	0.9906	0.9913
	severe	0.9101	0.9843	0.8438	0.9709	0.9029	0.9405
LO_OL	minor	0.9474	0.8717	0.8727	0.7212	0.7897	0.8343
	moderate	0.8938	0.8876	0.7846	0.7737	0.7791	0.8337
	severe	0.8765	0.9672	0.7921	0.9406	0.8600	0.9086

For moderate leaks, the model's recall values remain fairly stable, though some configurations exhibit slight fluctuations. In LO_LC, moderate recall dropped from 0.68 in Table 4.35 to 0.67 in Table 4.49, while in LO_CC, it slightly improved from 0.72 to 0.75. These minor shifts suggest that outliers had a limited but case-dependent impact on

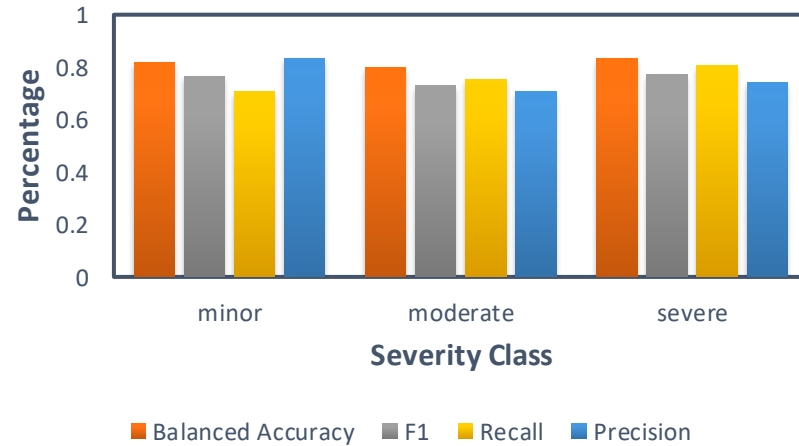
moderate leak classification. On the other hand, precision for moderate leaks declined slightly across all configurations, reinforcing the idea that the inclusion of outliers led to a slight increase in misclassified cases.

The bar charts in Figure 4.62 visually confirm these trends. The most notable improvement appears in minor leak recall, particularly in LO_GL and LO_OL, where the model now identifies a higher proportion of minor leaks compared to Figure 4.45. Conversely, the precision drop for minor and moderate leaks is evident, with moderate class precision experiencing slight reductions across all configurations. The balanced accuracy scores remain consistently high for severe cases, further confirming that the model maintains strong performance in detecting high-severity leaks.

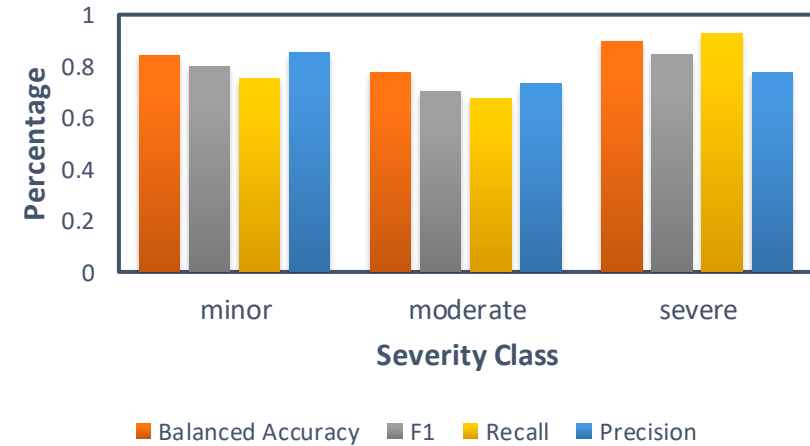
These results suggest that outlier inclusion introduces both benefits and challenges. On one hand, it enhances minor leak detection, likely by exposing the model to a broader range of minor leak characteristics that were not well represented in the original dataset. On the other hand, the slight decline in precision across minor and moderate cases indicates an increase in classification noise, possibly due to the presence of outliers that do not fit well within the model's learned patterns.

The impact of outliers in leak classification is particularly relevant in looped networks, where pressure fluctuations tend to be more stable, making it harder to differentiate small leaks from normal operational variations. The improved recall for minor leaks suggests that outliers introduced additional variability that helped the model learn more subtle distinctions between normal conditions and small leaks. However, the slight decrease in precision raises concerns about potential false alarms, which could lead to unnecessary interventions.

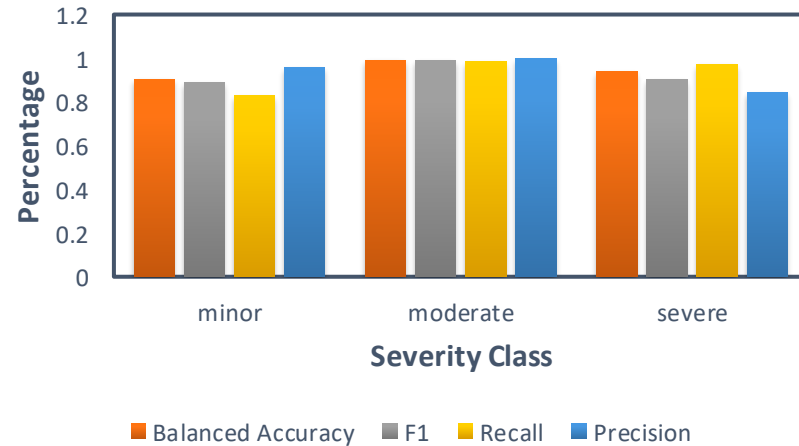
(a) XGB - Looped network - CC



(b) XGB - Looped network - LC



(c) XGB - Looped network - GL



(d) XGB - Looped network - OL

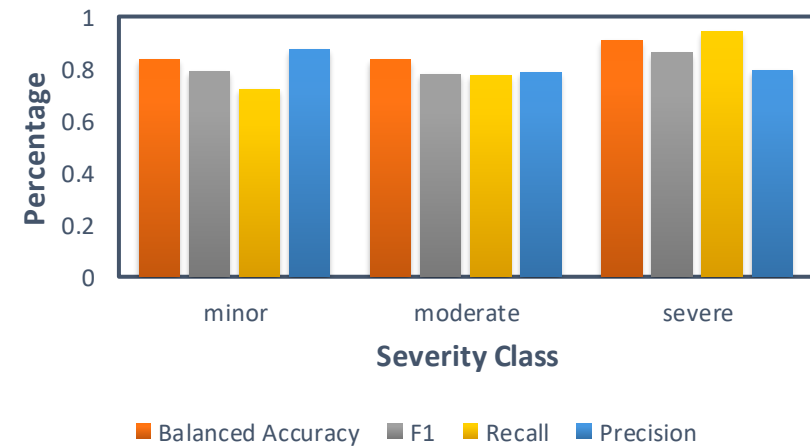


Figure 4.62: Bar charts presenting classification performance of XGB model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

In practical applications, the decision to include or exclude outliers should consider the trade-off between sensitivity and precision. If the primary goal is to ensure that no minor leaks are missed, incorporating outliers appears beneficial, as it significantly boosts minor leak recall. However, if precision is a higher priority to minimize false alarms, a more refined approach to outlier handling may be required. Techniques such as outlier-weighted learning, feature engineering to better capture minor leak signals, or selective outlier filtering could help strike a balance between improved recall and maintaining precision.

Overall, training the XGB model with the full dataset, including outliers, has led to notable improvements in minor leak recall while preserving strong performance in severe leak detection. The most significant enhancements are observed in LO_GL and LO_OL, where minor recall increased by over 20%. However, this improvement came with a slight trade-off in precision, particularly for moderate leaks. These findings highlight the potential value of outlier inclusion in enhancing model generalization while emphasizing the need for careful calibration to maintain precision across all severity levels.

The performance evaluation of the XGB model for leakage detection in a Looped WDS before removing outliers, as reflected in Table 4.50 and Figure 4.63, reveals consistently strong results with some variations in the classification of specific leak types. Balanced Accuracy remains high for most classes, with CC achieving 0.9994 in validation and 0.9993 in test, while GL maintains similarly high values at 0.9974 in validation and 0.9998 in test. However, there is a noticeable decline for OL, which records a Balanced Accuracy of 0.9992 in validation and 0.9993 in test, showing an improvement compared to Table 4.36, where OL classification had greater variability. Similarly, F1 Scores follow

a comparable trend, with CC reaching 0.9994 in validation and 0.9993 in test, while LC and NL show minor fluctuations but still exceed 0.9992 in both cases. These differences, while relatively small, suggest that outlier removal may contribute to stabilizing classification performance across all leak types, particularly for challenging categories such as OL. Furthermore, Recall scores for NL and OL in Table 4.50 indicate minor deviations, with values dropping slightly compared to their counterparts in Table 4.36, further highlighting the impact of outlier removal in ensuring a more uniform classification.

Table 4.50: By class performance metrics of XGB models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Validation	CC	0.99979	0.99977	0.99915	0.99909	0.99912	0.99944
	GL	0.99965	0.99996	0.99859	0.99984	0.99921	0.99974
	LC	0.99983	0.99973	0.99933	0.99891	0.99912	0.99937
	NL	0.99977	0.99993	0.99909	0.99971	0.99940	0.99974
	OL	0.99999	0.99964	0.99995	0.99856	0.99925	0.99927
Test	CC	0.99986	0.99980	0.99944	0.99920	0.99932	0.99953
	GL	0.99962	0.99996	0.99848	0.99984	0.99916	0.99973
	LC	0.99987	0.99973	0.99949	0.99891	0.99920	0.99939
	NL	0.99981	0.99994	0.99925	0.99976	0.99951	0.99979
	OL	0.99999	0.99973	0.99997	0.99893	0.99945	0.99946
Train	CC	1.00000	1.00000	0.99998	0.99999	0.99999	0.99999
	GL	1.00000	1.00000	0.99998	1.00000	0.99999	1.00000
	LC	1.00000	1.00000	1.00000	0.99999	1.00000	1.00000
	NL	1.00000	1.00000	0.99999	1.00000	1.00000	1.00000
	OL	1.00000	0.99999	1.00000	0.99998	0.99999	0.99999

The radar charts in Figure 4.63 reinforce these observations, illustrating slight deviations in the classification of NL and LC compared to the more stable patterns seen in Figure 4.46. Figure 4.63a (Balanced Accuracy) displays a high level of consistency for CC, GL, and LC, though minor dips are visible in OL and NL, indicating slightly less reliable classification performance before outlier removal. Figures 74b (F1 Score) and 74c (Recall)

similarly show small variations, with NL and OL in the test dataset exhibiting minor drops compared to the more stable figures observed in the radar charts of no outliers. The most pronounced fluctuation appears in Figure 4.63d (Precision), where LC and OL display greater variation, reflecting the slight instability in classification confidence when outliers are retained. Despite these variations, the XGB model maintains strong generalization capabilities, and while outlier removal contributes to more consistent results, its effect is relatively minor. This suggests that XGB is inherently robust in leak detection tasks, but preprocessing techniques such as outlier removal still play a role in refining classification performance, particularly for underrepresented leak types in a Looped WDS.

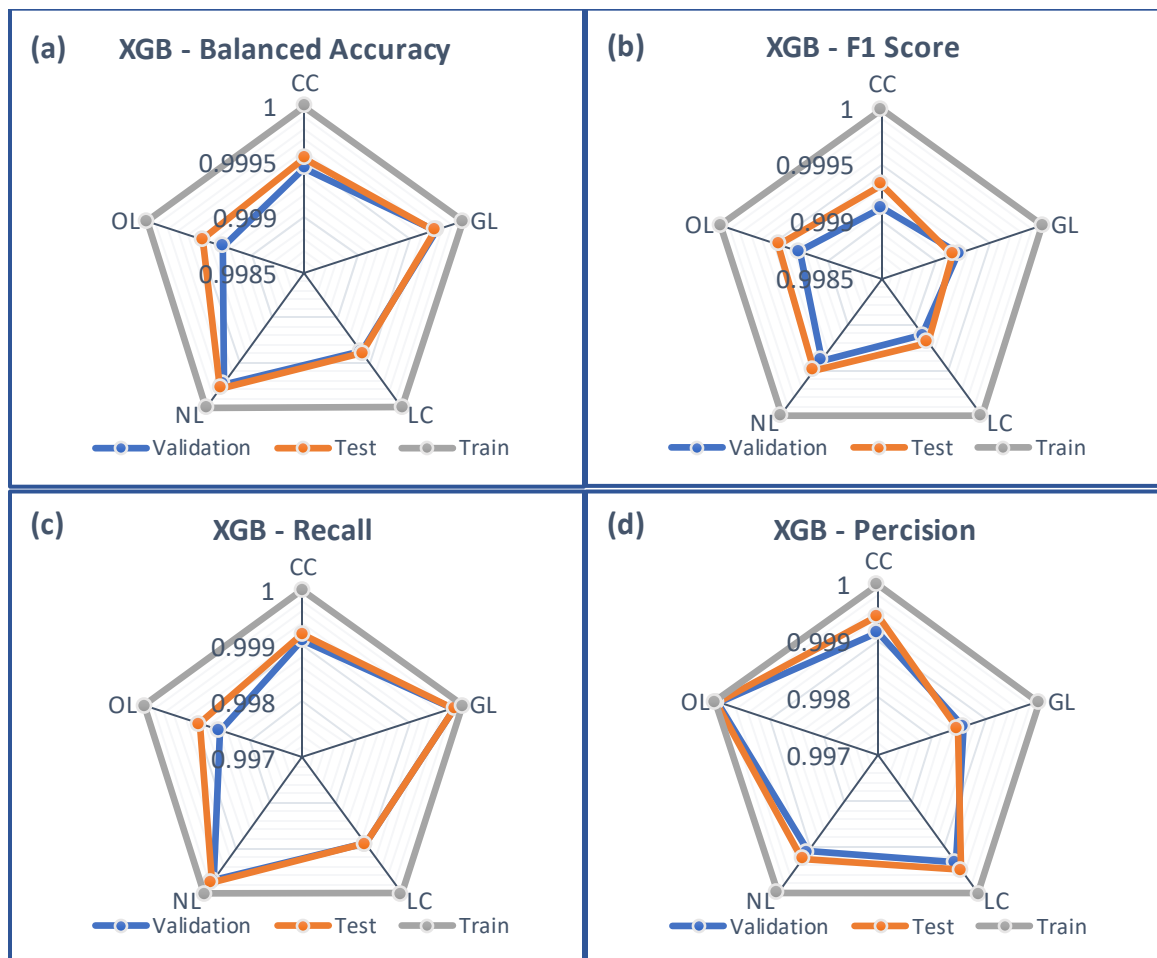


Figure 4.63: Radar charts illustrating XGB model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

4.3.2.2 Deep Learning models

Based on the comparison of Table 4.37 and Table 4.51 for deep learning models (LSTM, RNN, and GRU), it is evident that outlier removal has a noticeable impact on performance metrics. For LSTM, the LO_GL metric's accuracy decreased from 0.916662 to 0.8402406, and the Kappa score dropped from 0.87493 to 0.6451627, indicating a significant reduction in performance. The LO_OL metric also experienced a decline in accuracy from 0.74376667 to 0.7027078 and in Kappa from 0.61565 to 0.5395369. For RNN, the LO_GL metric's accuracy fell from 0.91842667 to 0.8418691, and Kappa decreased from 0.87764 to 0.6468864, showing a similar trend. Interestingly, the LO_OL metric for RNN remained relatively stable, with accuracy slightly decreasing from 0.74534 to 0.7060513 and Kappa from 0.61801 to 0.5410238, suggesting that outlier removal had a limited impact on this specific metric.

In the case of GRU, the performance was also affected, with the LO_GL metric's accuracy reducing from 0.91758 to 0.8410869 and Kappa from 0.87637 to 0.6469675. The LO_OL metric saw a smaller reduction in accuracy from 0.74564667 to 0.7057184 and in Kappa from 0.61847 to 0.5440043. The confidence intervals narrowed across most cases, indicating decreased prediction variability. The graphical representations in Figure 4.47 and Figure 4.64 further support these findings, showing more compact yet slightly lower performance bars after outlier removal. This suggests that while the removal of outliers reduced variance, it also led to a moderate decline in peak performance for deep learning models, particularly in metrics where the original performance was relatively high.

Table 4.51: Overall performance metrics of Deep learning models to evaluate their efficiency in severity assessment task over looped network before removing the outliers.

Model	Metric	LO_CC	LO_LC	LO_GL	LO_OL
LSTM	Accuracy	0.70969333	0.73734	0.91662	0.74376667
	Kappa	0.56454	0.60601	0.87493	0.61565
	CI Lower	0.70738893	0.73510503	0.91520957	0.74154942
	CI Upper	0.711989625	0.7395658	0.9180143	0.74597449
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333
RNN	Accuracy	0.71408	0.74126	0.91842667	0.74534
	Kappa	0.57112	0.61189	0.87764	0.61801
	CI Lower	0.711785902	0.73903576	0.91703007	0.74312718
	CI Upper	0.71636582	0.74347491	0.91980706	0.74754333
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333
GRU	Accuracy	0.71204667	0.74040667	0.91758	0.74564667
	Kappa	0.56807	0.61061	0.87637	0.61847
	CI Lower	0.70974776	0.73818007	0.9161769	0.74343472
	CI Upper	0.71433737	0.74262396	0.91896694	0.74784911
	No Info Rate	0.33333333	0.33333333	0.33333333	0.33333333

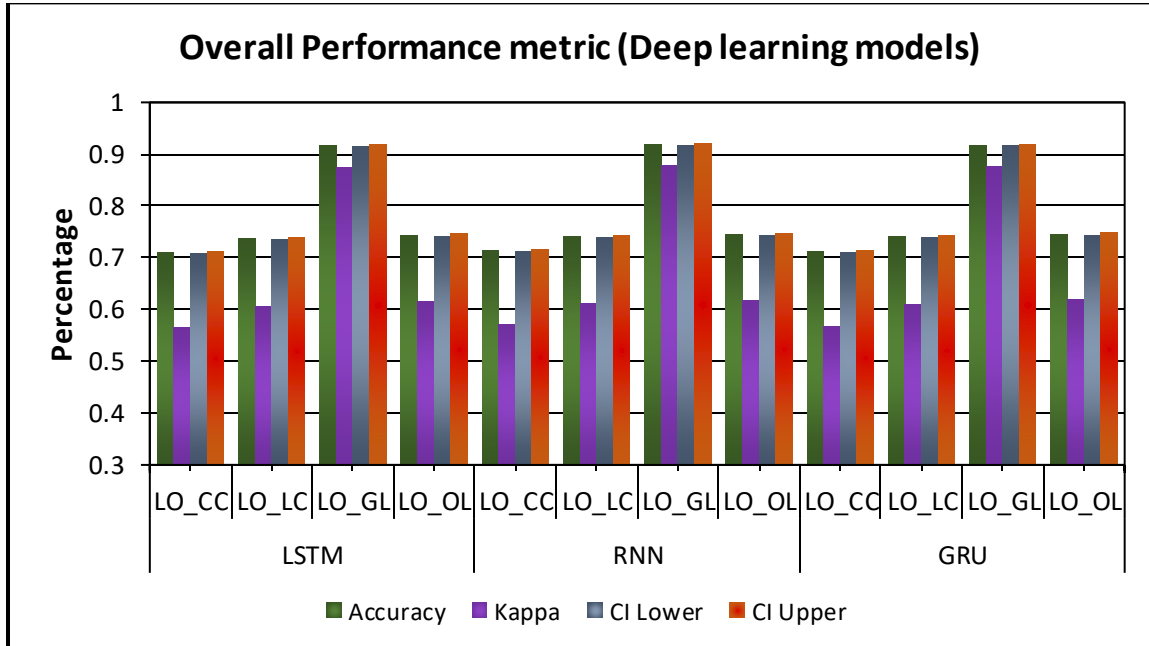


Figure 4.64: Bar chart visualization of overall performance metrics of Deep learning models to visually compare their efficiency in severity assessment task over looped network before removing the outliers.

The LSTM model's classification performance for leak severity levels in looped water distribution networks is analyzed by comparing Table 4.38 (without outliers) and Table 4.52 (with outliers), along with the corresponding performance bar charts. This comparison aims to assess how the inclusion of outliers influences the model's predictive capabilities across different network configurations (LO_CC, LO_LC, LO_GL, and LO_OL).

The model consistently demonstrates strong performance in detecting severe leaks, maintaining high recall values across all configurations. In Table 4.38, recall for severe cases ranged from 0.73 in LO_CC to approximately 0.91 in LO_LC and LO_OL. With the inclusion of outliers in Table 4.52, severe leak recall remains relatively stable, reflecting the model's robustness in identifying critical leaks despite data variability. However, precision for severe cases shows a minor decline, indicating an increase in false positives. This suggests that while outliers do not significantly impact the model's sensitivity to severe leaks, they may introduce some misclassifications into this category. Moderate leaks exhibit notable shifts in classification performance after incorporating outliers. In Table 4.38, recall for moderate leaks fluctuated between 0.57 and 0.70, revealing inconsistencies in the model's ability to correctly identify this class. In Table 4.52, moderate leak recall values show marginal improvements in some configurations, while precision fluctuates. These results suggest that the presence of outliers adds some noise to the classification process, making moderate leak identification more variable. The bar charts (Figure 4.64) confirm this pattern, illustrating slight variations in recall and F1 scores for moderate leaks compared to Figure 4.47.

One of the most significant impacts of outlier inclusion is observed in the detection of minor leaks. In Table 4.38, recall for minor leaks was particularly low, reaching only

0.38 in LO_LC and 0.45 in LO_OL. With outliers included (Table 4.52), recall values for minor leaks show mixed effects, some configurations experience slight improvements, while others remain poor. Balanced accuracy for minor cases remains low, confirming the persistent difficulty of distinguishing minor leaks from normal pressure fluctuations in looped networks. Specificity remains high for minor leaks, indicating that the model effectively rules out non-minor cases but struggles to correctly identify actual minor leak events.

Table 4.52: By class performance metrics of LSTM models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

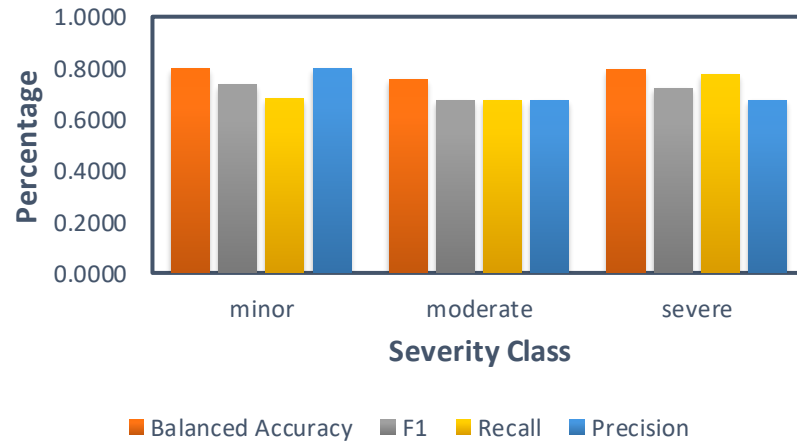
LSTM Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9139	0.8507	0.7978	0.6793	0.7338	0.7966
	moderate	0.8372	0.8368	0.6740	0.6734	0.6737	0.7553
	severe	0.8129	0.8786	0.6745	0.7753	0.7214	0.7941
LO_LC	minor	0.9281	0.8590	0.8286	0.6953	0.7562	0.8117
	moderate	0.8422	0.8098	0.6570	0.6043	0.6295	0.7233
	severe	0.8348	0.9492	0.7338	0.9107	0.8127	0.8728
LO_GL	minor	0.9739	0.9130	0.9398	0.8144	0.8726	0.8942
	moderate	0.9986	0.9898	0.9971	0.9795	0.9882	0.9890
	severe	0.9028	0.9766	0.8310	0.9567	0.8895	0.9297
LO_OL	minor	0.9300	0.8265	0.8133	0.6096	0.6969	0.7698
	moderate	0.8433	0.8491	0.6908	0.7002	0.6955	0.7718
	severe	0.8419	0.9549	0.7443	0.9206	0.8231	0.8812

The confusion matrices in Figure 4.48 previously revealed frequent misclassification of minor leaks into moderate or severe categories, with LO_CC and LO_GL showing particularly high misclassification rates. The bar charts indicate that these misclassification trends persist. The inclusion of outliers appears to amplify this issue rather than mitigate it, as recall for minor leaks remains low while moderate and severe categories absorb a proportion of misclassified minor cases. This suggests that the model

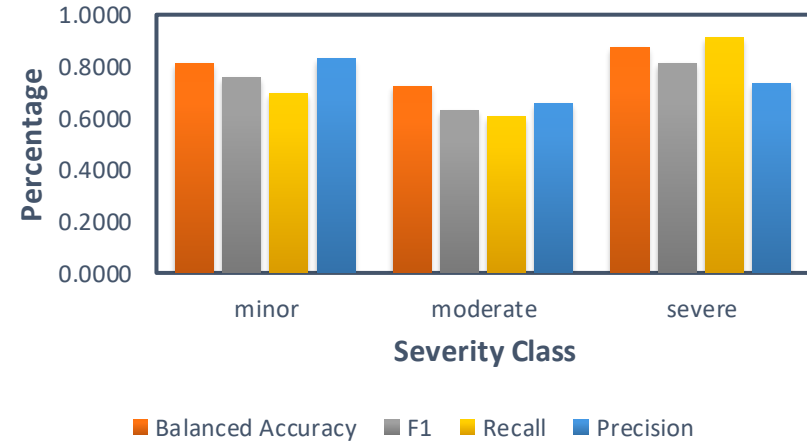
does not effectively learn from the additional data variance introduced by outliers, particularly in distinguishing minor leaks.

Figure 4.65 (LSTM bar charts with outliers) illustrate key changes in classification performance. Severe leaks maintain high recall and F1 scores across all network configurations, confirming the model's reliability in identifying critical failures. However, moderate and minor leak detection remains inconsistent, as evidenced by fluctuating recall and precision values across configurations. In particular, LO_LC and LO_CC show the lowest recall scores for minor leaks, supporting previous findings that the model struggles to capture the subtle pressure changes associated with minor leaks in looped networks. When compared to Figure 4.49, these bar charts reinforce the hypothesis that outlier inclusion does not significantly improve minor leak detection and may introduce additional classification noise.

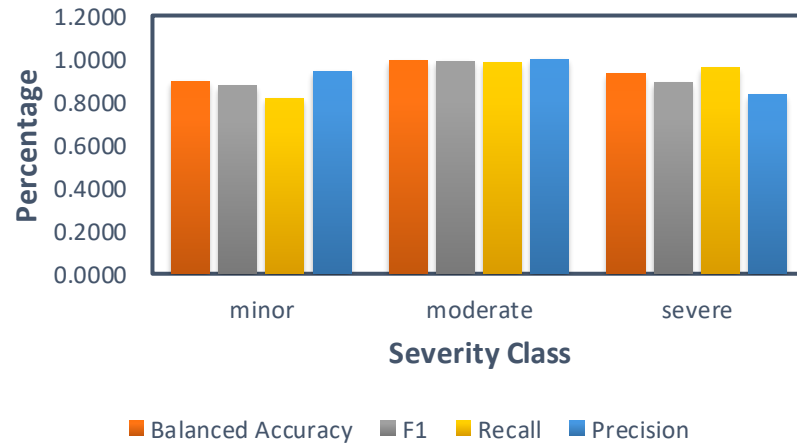
(a) **LSTM - Looped network - CC**



(b) **LSTM - Looped network - LC**



(c) **LSTM - Looped network - GL**



(d) **LSTM - Looped network - OL**

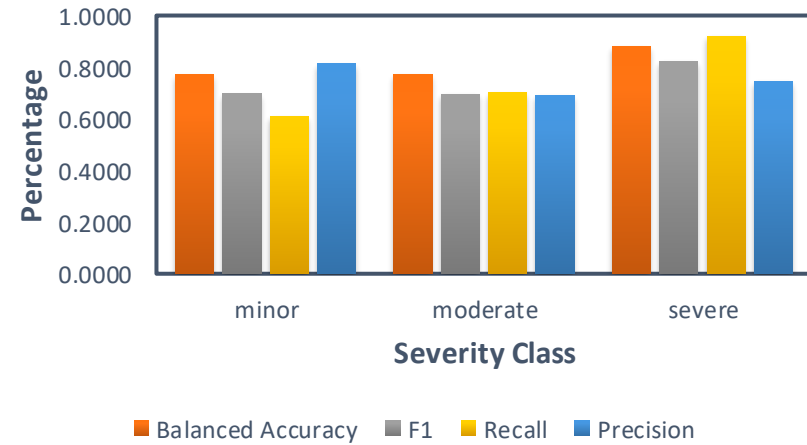


Figure 4.65: Bar charts presenting classification performance of LSTM model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

Future enhancements could focus on refining feature extraction techniques to improve minor leak sensitivity, incorporating advanced preprocessing to distinguish outliers from meaningful data variations, and exploring hybrid deep learning architectures such as CNN-LSTM models for improved spatial-temporal feature learning.

The performance evaluation of the LSTM model for leak detection in a Looped WDS before removing outliers, as illustrated in Table 4.53 and Figure 4.66, confirms the model's high accuracy but highlights minor inconsistencies when compared to the results after outlier removal. Balanced Accuracy remains strong, with CC scoring 0.9924 in the test dataset and 0.9938 in training, while GL maintains 0.9981 in test and 0.9985 in training. These values closely align with those in Table 4.39, where Balanced Accuracy exceeded 0.995 for most classes. However, LC presents a slightly lower Balanced Accuracy of 0.9949 in the test dataset and 0.9954 in training, indicating minor classification challenges. This aligns with Table 4.39, where LC also exhibited the lowest Balanced Accuracy among the major classes. Similarly, F1 Scores in Table 4.53 show strong performance across most categories, with CC reaching 0.9893 in the test dataset and 0.9905 in training, while OL achieves 0.9977 in test and 0.9982 in training. Compared to Table 4.39, where F1 Scores were also above 0.99, these values indicate that while classification remains stable before outlier removal, the process of removing outliers slightly refines the classification consistency, particularly in LC detection.

The radar charts in Figure 4.66 further reinforce these findings, highlighting minor deviations between the test and train datasets, particularly for LC and OL. Figure 4.66a (Balanced Accuracy) shows a consistently high level of performance across most leak types but with a slight drop in LC compared to the more uniform distribution observed in Figure

4.50a. Figure 4.66b and Figure 4.66c also reveal a strong alignment between train and test sets, but LC continues to show minor variation in recall, which is consistent with Table 4.53's numerical data. Figure 4.66d confirms this trend, as most classes maintain precision above 0.99, but LC experiences slight instability compared to the near-perfect precision values recorded in Table 4.39. These results indicate that while the LSTM model effectively generalizes across datasets, removing outliers provides minor improvements in classification consistency, particularly for leak types that exhibit slight classification difficulties. Nonetheless, the LSTM model remains a highly reliable choice for leak detection in a Looped WDS, demonstrating strong predictive performance with minimal differences before and after outlier removal.

Table 4.53: By class performance metrics of LSTM models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99793	0.99674	0.99168	0.98696	0.98931	0.99245
	GL	0.99947	0.99921	0.99786	0.99682	0.99734	0.99814
	LC	0.99634	0.99840	0.98548	0.99360	0.98952	0.99497
	NL	0.99939	0.99969	0.99756	0.99876	0.99816	0.99908
	OL	0.99989	0.99898	0.99954	0.99590	0.99772	0.99789
Train	CC	0.99832	0.99697	0.99323	0.98785	0.99053	0.99308
	GL	0.99956	0.99937	0.99825	0.99746	0.99785	0.99851
	LC	0.99655	0.99858	0.98632	0.99434	0.99031	0.99544
	NL	0.99953	0.99981	0.99812	0.99923	0.99867	0.99938
	OL	0.99994	0.99918	0.99976	0.99673	0.99824	0.99833

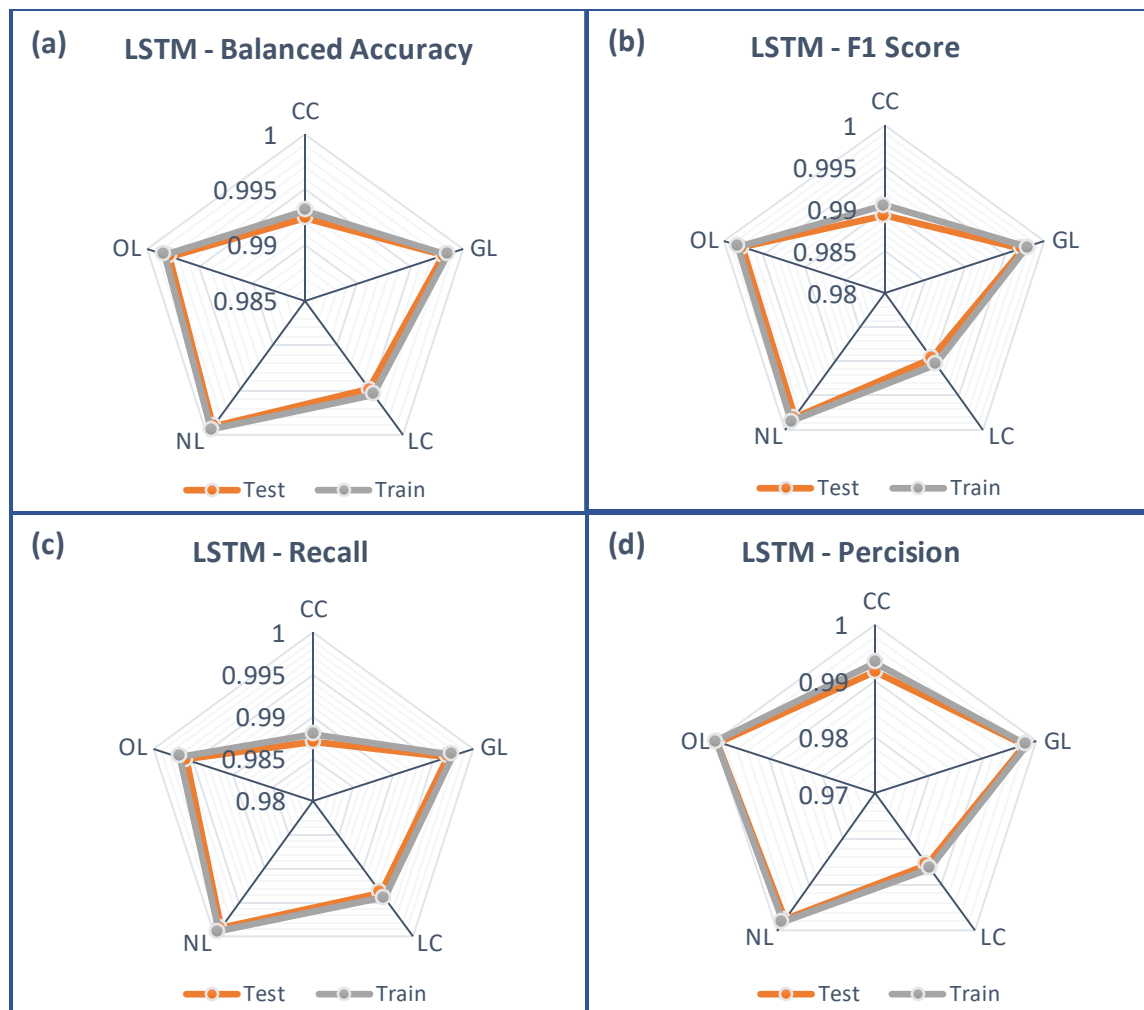


Figure 4.66: Radar charts illustrating LSTM model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The RNN model's classification performance for leak severity levels in looped WDS is analyzed by comparing Table 4.40 (without outliers) and Table 4.54 (with outliers), along with the corresponding performance bar charts. This comparison aims to assess how the inclusion of outliers influences the model's predictive capabilities across different network configurations (LO_CC, LO_LC, LO_GL, and LO_OL).

Table 4.54: By class performance metrics of RNN models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

RNN Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9335	0.8442	0.8313	0.6554	0.7330	0.7945
	moderate	0.8180	0.8527	0.6634	0.7175	0.6894	0.7677
	severe	0.8226	0.8798	0.6861	0.7753	0.7279	0.7990
LO_LC	minor	0.9370	0.8562	0.8448	0.6853	0.7567	0.8112
	moderate	0.8593	0.8079	0.6776	0.5915	0.6316	0.7254
	severe	0.8134	0.9660	0.7164	0.9427	0.8141	0.8781
LO_GL	minor	0.9817	0.9087	0.9564	0.8027	0.8729	0.8922
	moderate	0.9989	0.9901	0.9978	0.9799	0.9888	0.9894
	severe	0.8966	0.9845	0.8245	0.9717	0.8921	0.9341
LO_OL	minor	0.9475	0.8188	0.8469	0.5805	0.6888	0.7640
	moderate	0.8400	0.8577	0.6926	0.7212	0.7066	0.7806
	severe	0.8337	0.9657	0.7388	0.9407	0.8276	0.8872

The model consistently demonstrates strong performance in detecting severe leaks, maintaining high recall values across all configurations. In Table 4.40, recall for severe cases ranged from 0.77 in LO_CC to approximately 0.94 in LO_LC and LO_OL. With the inclusion of outliers in Table 4.54, severe leak recall remains relatively stable, reflecting the model's robustness in identifying critical leaks despite data variability. However, precision for severe cases shows a minor decline, indicating an increase in false positives. This suggests that while outliers do not significantly impact the model's sensitivity to severe leaks, they may introduce some misclassifications into this category.

Moderate leaks exhibit notable shifts in classification performance after incorporating outliers. In Table 4.40, recall for moderate leaks fluctuated between 0.59 and 0.72, revealing inconsistencies in the model's ability to correctly identify this class. In Table 4.54, moderate leak recall values show marginal improvements in some configurations, while precision fluctuates. These results suggest that the presence of

outliers adds some noise to the classification process, making moderate leak identification more variable.

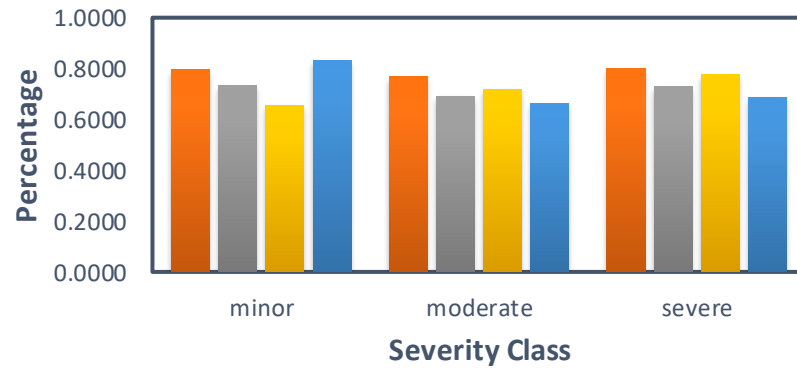
One of the most significant impacts of outlier inclusion is observed in the detection of minor leaks. In Table 4.40, recall for minor leaks was particularly low, reaching only 0.58 in LO_OL and 0.68 in LO_LC. With outliers included (Table 4.54), recall values for minor leaks show mixed effects, some configurations experience slight improvements, while others remain poor. Balanced accuracy for minor cases remains low, confirming the persistent difficulty of distinguishing minor leaks from normal pressure fluctuations in looped networks. Specificity remains high for minor leaks, indicating that the model effectively rules out non-minor cases but struggles to correctly identify actual minor leak events.

The confusion matrices in Figure 4.51 previously revealed frequent misclassification of minor leaks into moderate or severe categories, with LO_CC and LO_GL showing particularly high misclassification rates. The bar charts (Figure 4.67) indicate that these misclassification trends persist. The inclusion of outliers appears to amplify this issue rather than mitigate it, as recall for minor leaks remains low while moderate and severe categories absorb a proportion of misclassified minor cases. This suggests that the model does not effectively learn from the additional data variance introduced by outliers, particularly in distinguishing minor leaks.

Figure 4.67 illustrate key changes in classification performance. Severe leaks maintain high recall and F1 scores across all network configurations, confirming the model's reliability in identifying critical failures. However, moderate and minor leak detection remains inconsistent, as evidenced by fluctuating recall and precision values

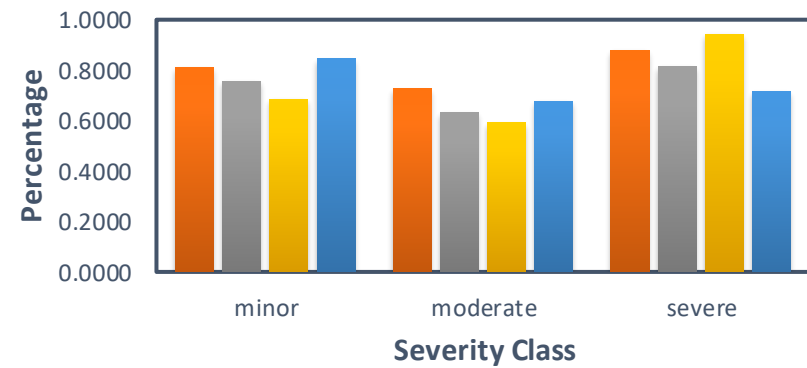
across configurations. In particular, LO_LC and LO_CC show the lowest recall scores for minor leaks, supporting previous findings that the model struggles to capture the subtle pressure changes associated with minor leaks in looped networks. When compared to Figure 4.52, these bar charts reinforce the hypothesis that outlier inclusion does not significantly improve minor leak detection and may introduce additional classification noise.

(a) **RNN - Looped network - CC**



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(b) **RNN - Looped network - LC**



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(c) **RNN - Looped network - GL**



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(d) **RNN - Looped network - OL**



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

Figure 4.67: Bar charts presenting classification performance of RNN model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

The RNN model for leak detection in the Looped WDS network exhibited strong performance across all classes, with Balanced Accuracy exceeding 0.995 for most leak types, as indicated in Table 4.41. The highest accuracy was observed for CC (0.9987) and GL (0.9985), reinforcing the model's ability to detect common leak types effectively. Precision values ranged between 0.992 and 0.999, ensuring minimal false positives, while recall values exceeded 0.99 for most cases, demonstrating strong sensitivity to leakage events. Figure 4.68 provide further validation, showing minimal deviation between the training and test datasets, with performance metrics aligning closely, suggesting a well-optimized and robust model.

Table 4.55: By class performance metrics of RNN models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99952	0.99983	0.99808	0.99932	0.99870	0.99942
	GL	0.99981	0.99986	0.99924	0.99944	0.99934	0.99963
	LC	0.99968	0.99973	0.99870	0.99892	0.99881	0.99930
	NL	0.99925	0.99971	0.99699	0.99884	0.99791	0.99904
	OL	0.99986	0.99898	0.99942	0.99590	0.99766	0.99788
Train	CC	0.99973	0.99985	0.99891	0.99942	0.99916	0.99957
	GL	0.99989	0.99988	0.99957	0.99952	0.99954	0.99971
	LC	0.99970	0.99987	0.99881	0.99948	0.99914	0.99959
	NL	0.99951	0.99984	0.99803	0.99937	0.99870	0.99944
	OL	0.99993	0.99932	0.99972	0.99726	0.99849	0.99860

In contrast, Table 4.55 shows the model's performance before outlier removal, where Balanced Accuracy was still high but exhibited slightly greater variance across leak types. While CC and GL remained above 0.997, classes such as NL and OL experienced minor fluctuations in recall, with a slight drop in sensitivity compared to their post-outlier-removal counterparts. Radar charts in Figure 4.68 reveal a similar pattern, where precision and recall scores displayed minor inconsistencies in NL and OL classifications, suggesting

that rare leaks were more challenging for the model to generalize before outlier filtering. The improved alignment in Table 4.41 and its radar charts highlights the benefits of outlier removal in enhancing model stability and reducing misclassification risk. These findings confirm that RNN remains highly effective in leakage detection for Looped WDS, with outlier removal further refining its generalization capabilities for real-world applications.

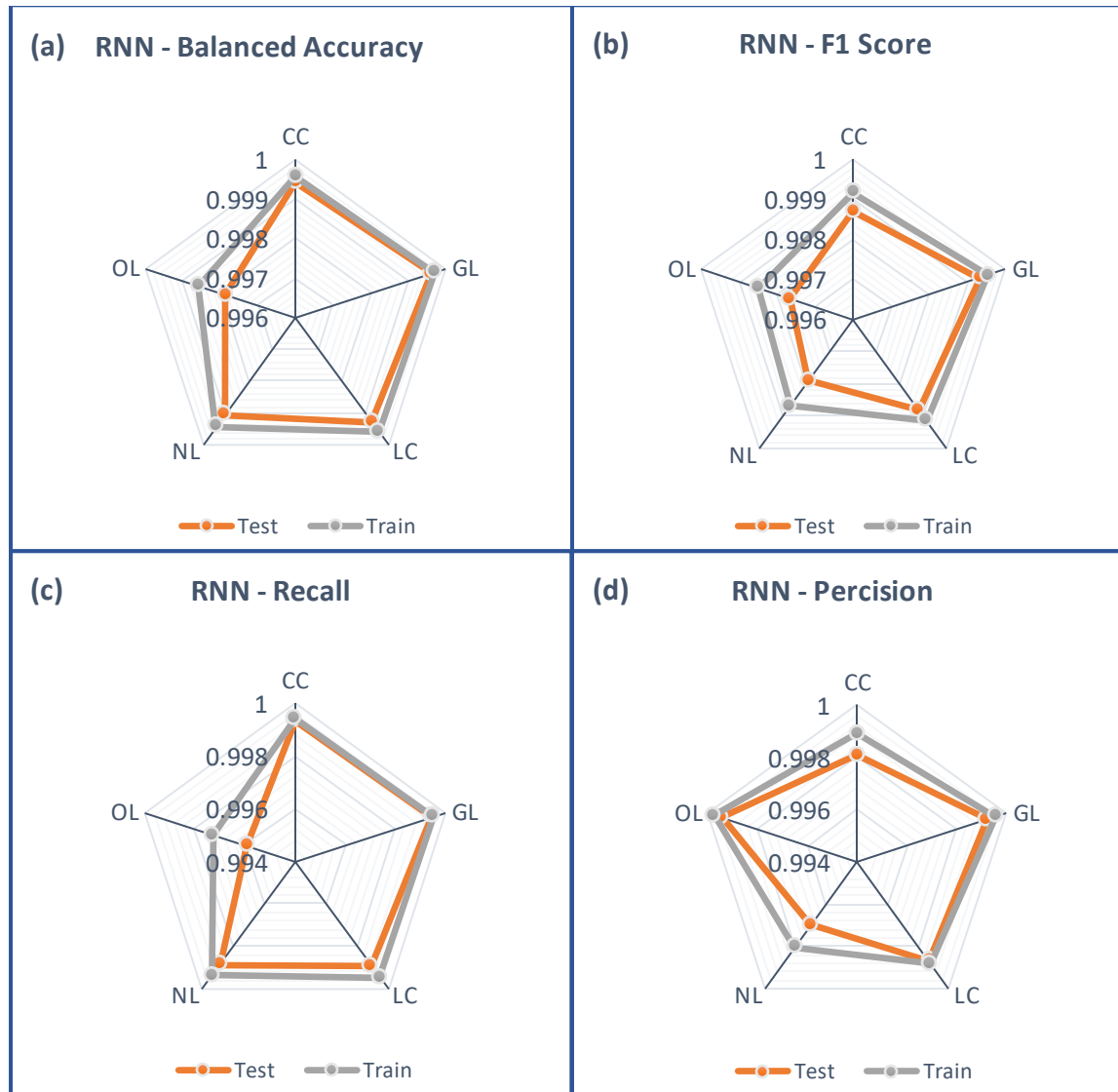


Figure 4.68: Radar charts illustrating RNN model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

The GRU model's classification performance for leak severity levels in looped water distribution networks is analyzed by comparing Table 4.42 (without outliers) and Table 4.56 (with outliers), along with the corresponding performance bar charts. This comparison provides insight into how outlier removal has influenced the model's ability to learn imbalanced data, favoring moderate severity classifications over minor ones, and impacting overall predictive performance across different network configurations (LO_CC, LO_LC, LO_GL, and LO_OL).

The removal of outliers has created a learning environment where the GRU model increasingly favors moderate leak classifications over minor ones. In Table 4.42, recall for minor leaks was already low, dropping to 0.41 in LO_LC, indicating a tendency to misclassify minor leaks into moderate or severe categories. In Table 4.56, after reintroducing outliers, the recall for minor leaks remains lower than other severity classes, suggesting that the model continues to struggle in recognizing minor leaks as a distinct category. Instead, moderate leak classifications are reinforced, leading to frequent misclassifications where minor cases are absorbed into the moderate class.

Table 4.56: By class performance metrics of GRU models to evaluate their efficiency in Severity Assessment over looped network before removing the outliers.

GRU Model	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
LO_CC	minor	0.9204	0.8499	0.8091	0.6748	0.7358	0.7976
	moderate	0.8367	0.8396	0.6756	0.6804	0.6780	0.7585
	severe	0.8115	0.8816	0.6747	0.7819	0.7244	0.7967
LO_LC	minor	0.9430	0.8511	0.8547	0.6701	0.7512	0.8066
	moderate	0.8267	0.8233	0.6505	0.6451	0.6478	0.7359
	severe	0.8408	0.9471	0.7400	0.9060	0.8147	0.8734
LO_GL	minor	0.9792	0.9098	0.9510	0.8058	0.8724	0.8925
	moderate	0.9993	0.9897	0.9986	0.9792	0.9888	0.9893
	severe	0.8978	0.9823	0.8256	0.9676	0.8910	0.9327
LO_OL	minor	0.9196	0.8353	0.7985	0.6373	0.7089	0.7785
	moderate	0.8621	0.8412	0.7098	0.6745	0.6917	0.7683
	severe	0.8388	0.9595	0.7424	0.9292	0.8254	0.8840

Severe leak classification remains the strongest across all configurations, with recall reaching 0.92 in LO_OL and 0.90 in LO_GL. The model maintains a high capability of identifying severe leak cases, a trend observed consistently in both Table 4.42 and Table 4.56. However, precision values for severe cases exhibit slight declines with outlier inclusion, indicating an increase in false positives. This suggests that the model, when trained without outliers, develops a more conservative decision boundary for severe cases, whereas the inclusion of outliers introduces noise, leading to more misclassified moderate cases as severe.

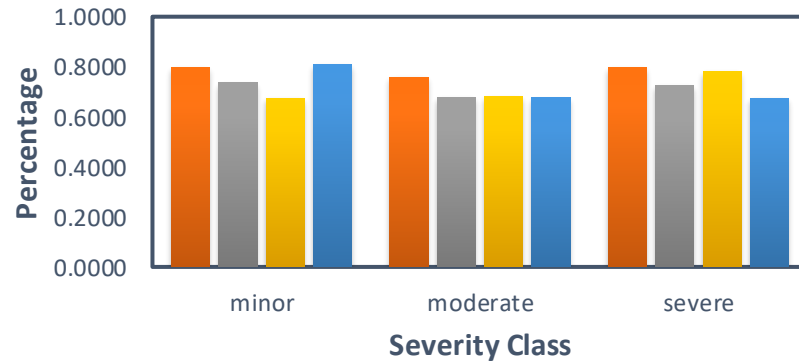
Moderate leak classification, on the other hand, benefits the most from the imbalance caused by outlier removal. In Table 4.42, recall for moderate leaks fluctuated between 0.64 and 0.72, demonstrating a range of inconsistencies in classification. After outliers were included in Table 4.56, the recall values for moderate leaks remain stable or improve slightly, reinforcing the idea that the model learns to prioritize moderate classifications over minor ones.

One of the most striking effects of outlier removal is the continued difficulty in identifying minor leaks accurately. The GRU model exhibits high specificity for minor cases, often above 0.90, confirming that it is good at ruling out non-minor cases. However, recall remains problematic, especially in LO_LC, where it drops significantly. This trend suggests that the model is biased towards classifying minor leaks as moderate, potentially due to the overlapping signal patterns in pressure data. Even with the reintroduction of outliers in Table 4.56, recall improvements for minor leaks are marginal, further confirming that the model has not effectively learned to distinguish minor leak characteristics.

The confusion matrices in Figure 4.54 revealed high misclassification rates, with thousands of minor leak cases being misclassified as moderate. In LO_CC alone, 22,117 minor cases were reclassified as moderate, emphasizing the tendency of the model to shift minor cases upward in severity classification. The bar charts in Figure 4.69 reflect similar trends, indicating that outliers do not significantly mitigate this bias. The moderate class absorbs a substantial portion of misclassified minor leaks, creating a false balance in model performance metrics while still underrepresenting minor leak detection.

Figure 4.69 illustrate how the balanced accuracy remains highest for severe cases across all configurations, reinforcing the model's strength in detecting critical leaks. However, F1-scores for minor cases remain the lowest, aligning with their recall issues. Precision values remain stable across severity classes, suggesting that when the model predicts a class, it does so with a reasonable level of confidence. Despite this, the drop in minor leak recall across configurations confirms that the GRU model struggles to adapt to the subtle variations in pressure patterns associated with minor leaks, especially in looped networks.

(a) GRU - Looped network - CC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(b) GRU - Looped network - LC



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(c) GRU - Looped network - GL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

(d) GRU - Looped network - OL



■ Balanced Accuracy ■ F1 ■ Recall ■ Precision

Figure 4.69: Bar charts presenting classification performance of GRU model across different Leak types for severity assessment task before outliers removal of Looped network dataset where (a) CC (b) LC (c) GL (d) OL.

The GRU model demonstrates strong performance in detecting severe leaks, but its inability to effectively capture minor leaks is exacerbated by the imbalanced learning process introduced by outlier removal. By reinforcing moderate leak classifications at the expense of minor ones, the model develops a classification bias that remains even after outliers are reintroduced. This suggests that rather than enhancing minor leak detection, the removal and subsequent reintroduction of outliers lead to a misleading balance in model performance. To address this, future improvements could involve resampling techniques, weighted loss functions, or refined feature selection methods to better differentiate minor severity levels in looped network configurations.

The results presented in Table 4.57 and visualized in Figure 4.70 provide a detailed performance assessment of the GRU model applied to the Looped WDS for leakage detection. Balanced Accuracy in the test dataset remains consistently above 0.995 across most classes, with CC and GL achieving 0.9967 and 0.9993, respectively. However, LC shows a slightly lower value of 0.9978, indicating minor classification challenges in this category. Compared to Table 4.43, where Balanced Accuracy exceeded 0.995 for all classes, the new results exhibit similar robustness, reinforcing the GRU model's high reliability in both cases.

The F1 scores in Table 4.57 further validate these trends, consistently exceeding 0.993, with CC at 0.9956 and OL at 0.9986 in the test dataset. These values closely match the results observed in Table 4.43, where F1 scores also remained strong, confirming the model's precision-recall balance. Figure 4.70 highlight this alignment between train and test results, with minimal deviations, supporting the model's effective generalization. However, compared to Figure 4.56, minor fluctuations in recall values for specific classes,

such as LC (0.9974 in Figure 4.70 vs. 0.9992 in Figure 4.56), suggest slight performance variations in leak identification.

Table 4.57: By class performance metrics of GRU models to evaluate their efficiency in Leakage Detection over looped network before removing the outliers.

dataset	Class	Specificity	Neg Pred Value	Precision	Recall	F1	Balanced Accuracy
Test	CC	0.99928	0.99853	0.99711	0.99410	0.99560	0.99669
	GL	0.99958	0.99976	0.99830	0.99906	0.99868	0.99932
	LC	0.99850	0.99928	0.99402	0.99714	0.99558	0.99782
	NL	0.99973	0.99968	0.99892	0.99870	0.99881	0.99922
	OL	0.99974	0.99958	0.99896	0.99830	0.99863	0.99902
Train	CC	0.99948	0.99881	0.99792	0.99525	0.99659	0.99737
	GL	0.99973	0.99982	0.99891	0.99929	0.99910	0.99951
	LC	0.99873	0.99948	0.99494	0.99791	0.99642	0.99832
	NL	0.99986	0.99975	0.99943	0.99898	0.99920	0.99942
	OL	0.99979	0.99974	0.99917	0.99894	0.99905	0.99937

Figure 4.70 illustrate the GRU model's high classification reliability. Figure 4.70a shows Balanced Accuracy consistency, with only minimal differences between train and test datasets. Figure 4.70b confirms stable F1-scores across leak classes, closely following Balanced Accuracy trends. Figure 4.70c presents recall values above 0.99, aligning well with Table 4.43's findings, though NL exhibits a slight dip in Table 4.57 (0.9988 vs. 0.9992). Lastly, Figure 4.70d reinforces precision stability, with values ranging from 0.994 to 0.999, nearly identical to the trends observed in Table 4.43.

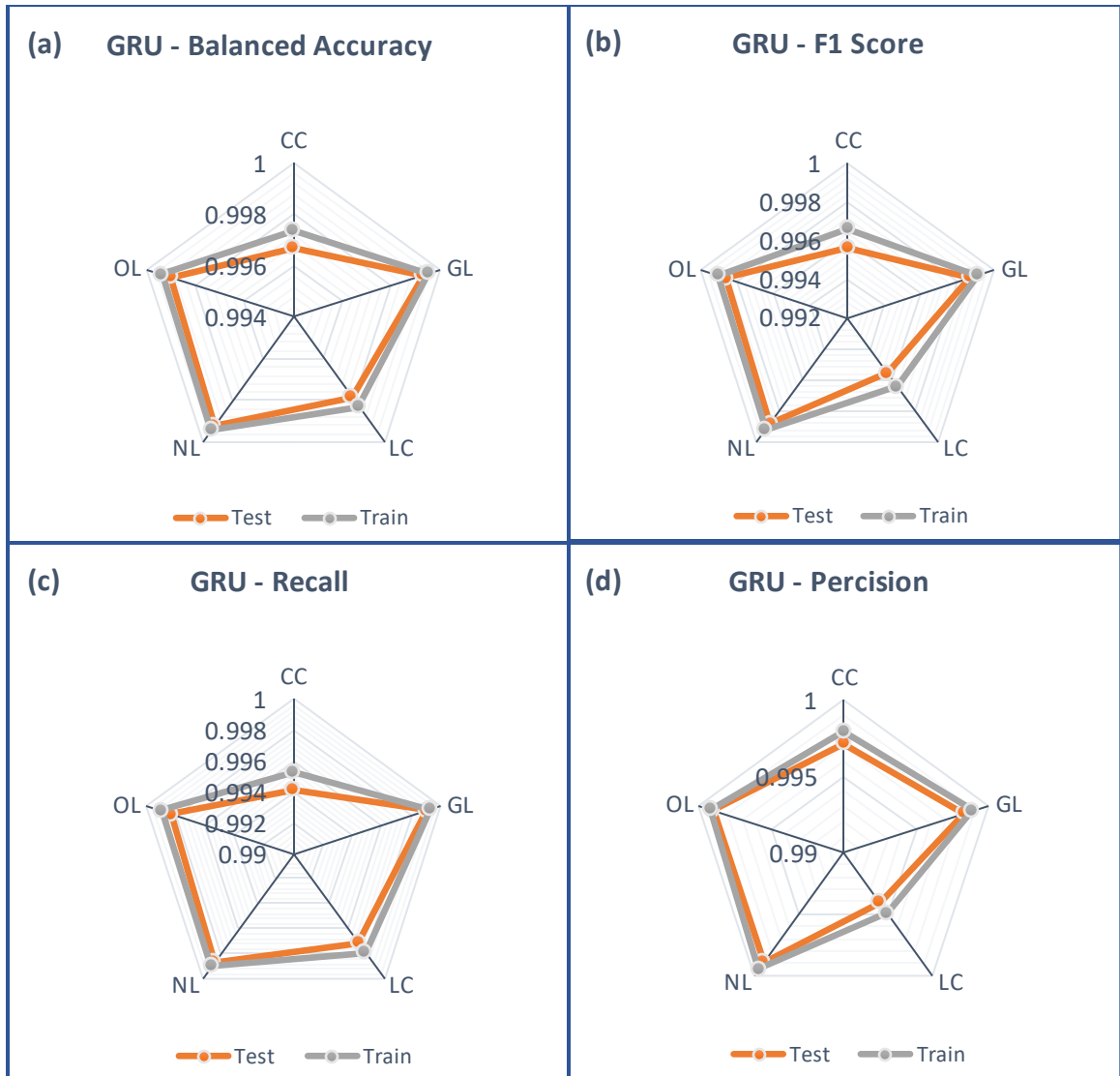


Figure 4.70: Radar charts illustrating GRU model performance before removing outliers for Leakage type determination task of Looped network across different leakage classes in terms of accuracy metrics (a) Balanced Accuracy (b) F1 Score (c) Recall (d) Precision.

CHAPTER 5

CONCLUSION

This research successfully developed high-performing machine learning (ML) models capable of accurately classifying leakage types and assessing severity levels within water distribution networks. Both tree-based supervised learning algorithms (XGB, Random Forest, and LightGBM) and deep supervised learning models (Long Short-Term Memory [LSTM], Recurrent Neural Networks [RNN], and Gated Recurrent Units [GRU]) demonstrated high performance metrics, including accuracy, precision, recall, and F1-score, when distinguishing between leakage types (Non-Leak, Longitudinal Crack, Circumferential Crack, Gasket Leak, and Orifice Leak). These results were derived from multi-sensor data, specifically readings from dynamic pressure transducers, accelerometers, and hydrophones, collected under varying operational conditions.

The impact of outlier removal was thoroughly analyzed and found to significantly influence the leakage detection task. Specifically, removing outliers introduced notable data scarcity for the Orifice Leak (OL) and Gasket Leak (GL) classes, leading the ML classifiers to favor other leakage categories during the prediction process of unseen data. This phenomenon was most evident in tree-based models, where OL and GL classes initially had the lowest recall and F1-score values post-outlier removal. However, incorporating all outliers back into the training dataset notably improved these metrics, highlighting the importance of careful preprocessing strategies to ensure balanced model performance across all leakage classes. The best-performing model, XGB, achieved

approximately 99.9% balanced accuracy and an F1-score around 0.99, confirming its robustness and suitability for practical implementation.

For leakage severity assessment, eight ML models based on both tree-based and deep learning algorithms were developed to classify leak severity levels into severe, moderate, or minor categories, based on the quantified leak flow rates. Among these models, LightGBM demonstrated superior performance, achieving an overall accuracy of approximately 93% with a Kappa statistic (κ) around 0.85. This model successfully identified severe leak scenarios with a high recall of about 97%. However, classifying minor leaks posed challenges due to their subtle signatures, occasionally overlapping with normal operational fluctuations, resulting in lower precision rates. The developed severity assessment framework proved effective at reliably differentiating leak scenarios from normal operations using steady-state demand sensor data, thus minimizing false positives and enhancing the decision-making reliability.

The expected outcomes of this research extend beyond the accurate classification of leakage types and severity levels. By facilitating early detection and precise identification of leaks, the proposed models and frameworks are anticipated to significantly reduce water losses, leading to notable energy savings in water extraction, treatment, and distribution processes. This directly contributes to sustainability goals by reducing the carbon footprint associated with excessive pumping and treatment activities. Furthermore, the accurate prediction of leakage types and severity levels is expected to support more targeted and efficient maintenance strategies, minimizing unnecessary interventions and reducing operational costs. The integration of advanced ML models into real-time monitoring systems will empower utility managers with actionable insights, enabling

proactive measures that not only enhance the reliability and resilience of water distribution networks but also promote the sustainable use of resources. Economically, these advancements are projected to lower overall water supply costs, extending the lifespan of infrastructure and fostering a more efficient allocation of financial resources within the water sector. By addressing both environmental and economic dimensions, this research aims to contribute meaningfully to global efforts in achieving water security and sustainable urban development.

Figure 5.1 presents a conceptual overview of the expected outcomes of applying ML models to classify water leakage types and assess their severity in WDS. It highlights how supervised learning approaches can enhance leakage detection, prioritize maintenance efforts, improve system efficiency, and promote sustainability by reducing resource wastage and operational costs. The figure emphasizes the broader benefits of integrating advanced ML-driven frameworks into water management, including ecological, economic, and infrastructural advantages.

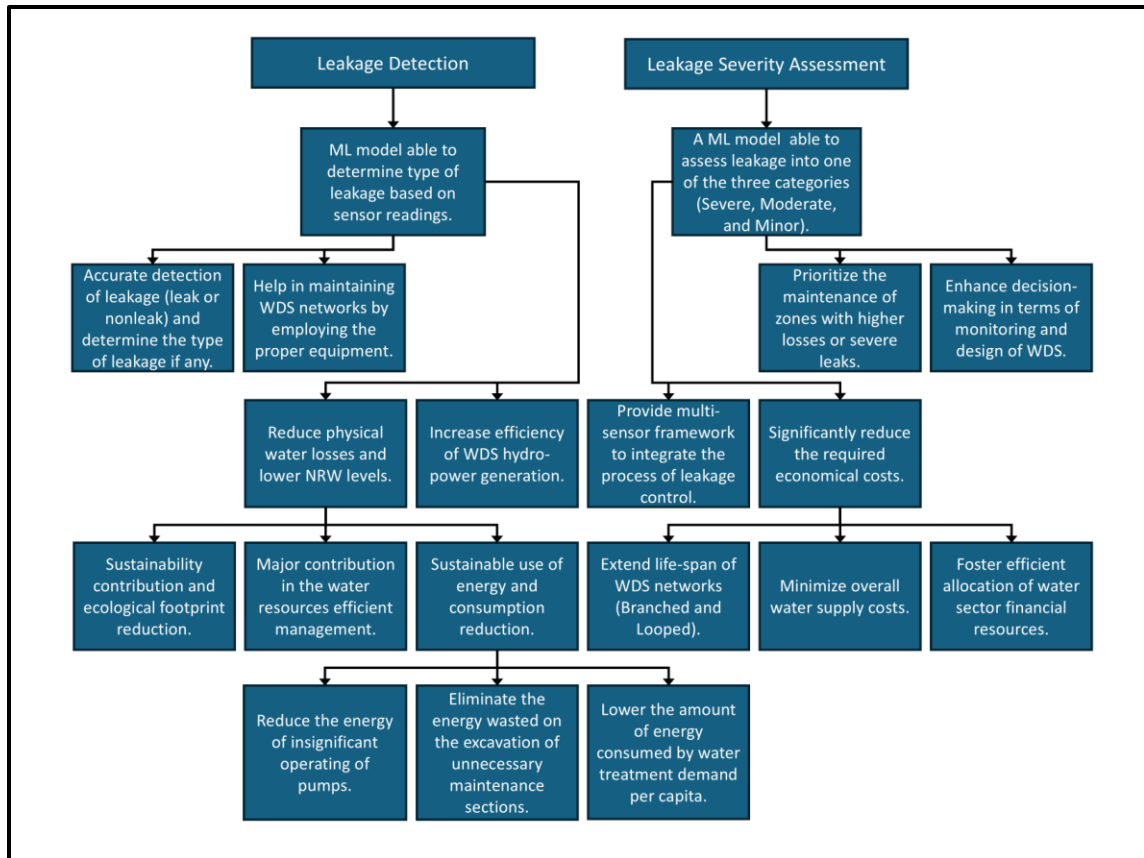


Figure 5.1: visual presentation of the expected impact and outcomes of ML models of water leakage type classification and severity assessment.

The integration of Leakage detection and Severity Assessment is anticipated to deliver a multi-sensor ML framework that can effectively classify leak types and determine their severity levels. This framework will serve as a practical tool for prioritizing pipeline maintenance schedules and establishing appropriate NRW minimization policies. Ultimately, the implemented ML-based diagnostic system provides water utilities with actionable insights, enabling prioritization of leak repairs, optimization of maintenance schedules, and reduction in non-revenue water losses. The comprehensive performance validation across different network topologies (branched and looped) and dynamic operational conditions underscores the scalability, practicality, and economic viability of

the developed framework, significantly advancing sustainable water management practices.

References

- [1] S. El-Zahab and T. Zayed, "Leak detection in water distribution networks: an introductory overview," *Smart Water*, vol. 4, no. 1, p. 5, Dec. 2019, doi: 10.1186/s40713-019-0017-x.
- [2] S. S. Aljameel *et al.*, "Oil and Gas Pipelines Leakage Detection Approaches: A Systematic Review of Literature," *Int. J. Saf. Secur. Eng.*, vol. 14, no. 3, pp. 773–786, Jun. 2024, doi: 10.18280/ijssse.140310.
- [3] A. V. Serafeim, N. T. Fourniotis, R. Deidda, G. Kokosalakis, and A. Langousis, "Leakages in Water Distribution Networks: Estimation Methods, Influential Factors, and Mitigation Strategies—A Comprehensive Review," *Water*, vol. 16, no. 11, p. 1534, May 2024, doi: 10.3390/w16111534.
- [4] E. Pauline, Carreño-Alvarado, and G. R. Meza, "Water Leak Detection by Thermographic Image Analysis, In Laboratory Tests," in *The 4th International Electronic Conference on Water Sciences*, Basel Switzerland: MDPI, May 2020, p. 15. doi: 10.3390/ECWS-4-06440.
- [5] Chandru K S, Hema Raja Shri P K, and Brindhavani C, "UNDERGROUND WATER PIPE LEAKAGE SYSTEM," *Int. J. Eng. Technol. Manag. Sci.*, vol. 7, no. 2, pp. 416–422, 2023, doi: 10.46647/ijetms.2023.v07i02.050.
- [6] W. Li *et al.*, "Development of systems for detection, early warning, and control of pipeline leakage in drinking water distribution: A case study," *J. Environ. Sci.*, vol. 23, no. 11, pp. 1816–1822, 2011, doi: 10.1016/S1001-0742(10)60577-3.
- [7] D. P. Sousa, R. Du, J. Mairton Barros da Silva Jr, C. C. Cavalcante, and C. Fischione, "Leakage detection in water distribution networks using machine-learning strategies," *Water Supply*, vol. 23, no. 3, pp. 1115–1126, Mar. 2023, doi: 10.2166/ws.2023.054.
- [8] A. Malekpour and Y. She, "Real-time leak detection in oil pipelines using an Inverse Transient Analysis model," *J. Loss Prev. Process Ind.*, vol. 70, p. 104411, May 2021, doi: 10.1016/j.jlp.2021.104411.
- [9] R. Li, H. Huang, K. Xin, and T. Tao, "A review of methods for burst/leakage detection and location in water distribution systems," *Water Supply*, vol. 15, no. 3, pp. 429–441, Jun. 2015, doi: 10.2166/ws.2014.131.
- [10] A. Haghighi and H. Shamloo, "Transient generation in pipe networks for leak detection," *Proc. Inst. Civ. Eng. - Water Manag.*, vol. 164, no. 6, pp. 311–318, Jun. 2011, doi: 10.1680/wama.2011.164.6.311.
- [11] M. Javadiha, J. Blesa, A. Soldevila, and V. Puig, "Leak Localization in Water Distribution Networks using Deep Learning," in *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, IEEE, Apr. 2019, pp. 1426–1431. doi: 10.1109/CoDIT.2019.8820627.
- [12] J. Li, W. Zheng, and C. Lu, "An Accurate Leakage Localization Method for Water Supply Network Based on Deep Learning Network," *Water Resour. Manag.*, vol. 36, no. 7, pp. 2309–2325, May 2022, doi: 10.1007/s11269-022-03144-x.
- [13] E. Mohammed, E. M. Jamal, and J. Abdelilah, "Innovative Methodologies for Leak Detection Using Hydrophone Data and Machine Learning Techniques," in

- 2024 IEEE 12th International Symposium on Signal, Image, Video and Communications (ISIVC), IEEE, May 2024, pp. 1–6. doi: 10.1109/ISIVC61350.2024.10577921.
- [14] N. M. Mahdi, A. H. Jassim, S. A. Abulqasim, A. Basem, A. A. F. Ogaili, and L. A. Al-Haddad, “Leak detection and localization in water distribution systems using advanced feature analysis and an Artificial Neural Network,” *Desalin. Water Treat.*, vol. 320, p. 100685, Oct. 2024, doi: 10.1016/j.dwt.2024.100685.
 - [15] D. U. Leonzio, P. Bestagini, M. Marcon, G. P. Quarta, and S. Tubaro, “Water Leak Detection and Classification Using Multiple Sensors,” in *2024 IFIP Networking Conference (IFIP Networking)*, IEEE, Jun. 2024, pp. 690–695. doi: 10.23919/IFIPNetworking62109.2024.10619914.
 - [16] S. Lee and B. Kim, “Machine Learning Model for Leak Detection Using Water Pipeline Vibration Sensor,” *Sensors*, vol. 23, no. 21, p. 8935, Nov. 2023, doi: 10.3390/s23218935.
 - [17] Y. Shen and W. Cheng, “A Tree-Based Machine Learning Method for Pipeline Leakage Detection,” *Water*, vol. 14, no. 18, p. 2833, Sep. 2022, doi: 10.3390/w14182833.
 - [18] M. F. Siddique, Z. Ahmad, N. Ullah, S. Ullah, and J.-M. Kim, “Pipeline Leak Detection: A Comprehensive Deep Learning Model Using CWT Image Analysis and an Optimized DBN-GA-LSSVM Framework,” *Sensors*, vol. 24, no. 12, p. 4009, Jun. 2024, doi: 10.3390/s24124009.
 - [19] G. Kunkel and R. Sturm, “Piloting proactive, advanced leakage management technologies,” *J. AWWA*, vol. 103, no. 2, pp. 62–75, Feb. 2011, doi: 10.1002/j.1551-8833.2011.tb11402.x.
 - [20] G. He, Y. Li, Y. Huang, L. Sun, and K. Liao, “A framework of smart pipeline system and its application on multiproduct pipeline leakage handling,” *Energy*, vol. 188, p. 116031, Dec. 2019, doi: 10.1016/j.energy.2019.116031.
 - [21] L. McMillan, J. Fayaz, and L. Varga, “Machine-Learning-Based Health Monitoring and Leakage Management of Water Distribution Systems,” *Proc. Int. Conf. Evol. Cities*, pp. 43–50, Jul. 2023, doi: 10.55066/proc-icec.2022.37.
 - [22] S. El-Zahab, A. Al-Sakkaf, E. Mohammed Abdelkader, and T. Zayed, “A Novel Lazy Serpent Algorithm for the Prioritization of Leak Repairs in Water Networks,” *Water*, vol. 12, no. 8, p. 2235, Aug. 2020, doi: 10.3390/w12082235.
 - [23] Z. Poulakis, D. Valougeorgis, and C. Papadimitriou, “Leakage detection in water pipe networks using a Bayesian probabilistic framework,” *Probabilistic Eng. Mech.*, vol. 18, no. 4, pp. 315–327, Oct. 2003, doi: 10.1016/S0266-8920(03)00045-6.
 - [24] R. Puust, Z. Kapelan, D. A. Savic, and T. Koppel, “A review of methods for leakage management in pipe networks,” *Urban Water J.*, vol. 7, no. 1, pp. 25–45, Feb. 2010, doi: 10.1080/15730621003610878.
 - [25] D. DeSilva *et al.*, “Sustainable management of leakage from wastewater pipelines,” *Water Sci. Technol.*, vol. 52, no. 12, pp. 189–198, Dec. 2005, doi: 10.2166/wst.2005.0459.
 - [26] M. De Marchis and B. Milici, “Leakage Estimation in Water Distribution Network: Effect of the Shape and Size Cracks,” *Water Resour. Manag.*, vol. 33, no. 3, pp. 1167–1183, Feb. 2019, doi: 10.1007/s11269-018-2173-4.

- [27] N. Ullah, Z. Ahmed, and J.-M. Kim, "Pipeline Leakage Detection Using Acoustic Emission and Machine Learning Algorithms," *Sensors*, vol. 23, no. 6, p. 3226, Mar. 2023, doi: 10.3390/s23063226.
- [28] A. Berglund, "An Iterative Simulation and Mathematical Programming Optimization Approach to Leak Detection in Water Distribution Systems,," 2013. [Online]. Available: <https://api.semanticscholar.org/CorpusID:63080104>
- [29] D. Tu, S. Yuan, Y. Jie, G. Zhang, H. Zhang, and D. Hou, "In-Pipe Free-Swimming Device for Leak Detection in Water Distribution Pipe Based on SVM,," 2018. [Online]. Available: <https://api.semanticscholar.org/CorpusID:201881558>
- [30] G. M. Komba, T. E. Mathonsi, and P. A. Owolawi, "Water Pipeline Leak Detection and Localisation in Water Distribution Networks,," in *2023 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, IEEE, Aug. 2023, pp. 140–145. doi: 10.1109/ETNCC59188.2023.10284934.
- [31] Y. Wu, S. Liu, and Z. Kapelan, "Addressing data limitations in leakage detection of water distribution systems: Data creation, data requirement reduction, and knowledge transfer," *Water Res.*, vol. 267, p. 122471, Dec. 2024, doi: 10.1016/j.watres.2024.122471.
- [32] A. Predescu, M. Mocanu, and C. Lupu, "A modern approach for leak detection in water distribution systems,," in *2018 22nd International Conference on System Theory, Control and Computing (ICSTCC)*, IEEE, Oct. 2018, pp. 486–491. doi: 10.1109/ICSTCC.2018.8540733.
- [33] M. Romano, Z. Kapelan, and D. A. Savić, "Real-Time Leak Detection in Water Distribution Systems,," in *Water Distribution Systems Analysis 2010*, Reston, VA: American Society of Civil Engineers, Dec. 2011, pp. 1074–1082. doi: 10.1061/41203(425)97.
- [34] G. Dvajasvie, B. P. Farisha, S. N. Babu, K. P. Saheen, and N. C. Binoy, "Leak Detection in Water-Distribution Pipe System,," in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, Jun. 2018, pp. 1–4. doi: 10.1109/ICCONS.2018.8663193.
- [35] G. M. Komba, T. E. Mathonsi, and P. A. Owolawi, "Optimizing Leak Detection and Location in Water Distribution Networks Using SVM-RF Algorithm,," in *2024 15th International Conference on Mechanical and Intelligent Manufacturing Technologies (ICMIMT)*, IEEE, May 2024, pp. 27–33. doi: 10.1109/ICMIMT61937.2024.10585935.
- [36] Q. Fang, J. Chen, and C. Xie, "Research on leakage area detection method in water distribution network based on gray wolf optimization,," *Water Supply*, vol. 23, no. 2, pp. 645–656, Feb. 2023, doi: 10.2166/ws.2023.014.
- [37] A. Daoudi, M. Benbrahim, and K. Benjelloun, "An Intelligent System to Classify Leaks in Water Distribution Pipes,," pp. 1–3, 2005.
- [38] J. Fontes, "Leak Detection and Location in Water Distribution Networks,," 2016.
- [39] L. Romero-Ben, D. Alves, J. Blesa, G. Cembrano, V. Puig, and E. Duviella, "Leak detection and localization in water distribution networks: Review and perspective,," *Annu. Rev. Control*, vol. 55, pp. 392–419, 2023, doi: 10.1016/j.arcontrol.2023.03.012.
- [40] R. Liu *et al.*, "Data-Driven Approaches for Vibroacoustic Localization of Leaks in

- Water Distribution Networks,” *Environ. Process.*, vol. 11, no. 1, p. 14, Mar. 2024, doi: 10.1007/s40710-024-00682-x.
- [41] M. R. Islam, S. Azam, B. Shanmugam, and D. Mathur, “A Review on Current Technologies and Future Direction of Water Leakage Detection in Water Distribution Network,” *IEEE Access*, vol. 10, pp. 107177–107201, 2022, doi: 10.1109/ACCESS.2022.3212769.
 - [42] S. Rathi, “A critical review of leakage detection strategies including pressure and water quality sensor placement in water distribution systems – sole and integrated approaches for leakage and contamination intrusion,” in *Proceedings - 2nd International Join Conference on Water Distribution System Analysis (WDSA)& Computing and Control in the Water Industry (CCWI)*, València: Editorial Universitat Politècnica de València, Jul. 2022. doi: 10.4995/WDSA-CCWI2022.2022.14869.
 - [43] X. Wan, P. K. Kuhanestani, R. Farmani, and E. Keedwell, “Literature Review of Data Analytics for Leak Detection in Water Distribution Networks: A Focus on Pressure and Flow Smart Sensors,” *J. Water Resour. Plan. Manag.*, vol. 148, no. 10, Oct. 2022, doi: 10.1061/(ASCE)WR.1943-5452.0001597.
 - [44] S. Tariq, Z. Hu, and T. Zayed, “Micro-electromechanical systems-based technologies for leak detection and localization in water supply networks: A bibliometric and systematic review,” *J. Clean. Prod.*, vol. 289, p. 125751, Mar. 2021, doi: 10.1016/j.jclepro.2020.125751.
 - [45] A. Scozzari, S. Mounce, D. Han, F. Soldovieri, and D. Solomatine, Eds., *ICT for Smart Water Systems: Measurements and Data Science*, vol. 102. in The Handbook of Environmental Chemistry, vol. 102. Cham: Springer International Publishing, 2021. doi: 10.1007/978-3-030-61973-2.
 - [46] M. Righetti, C. M. G. Bort, M. Bottazzi, A. Menapace, and A. Zanfei, “Optimal Selection and Monitoring of Nodes Aimed at Supporting Leakages Identification in WDS,” *Water*, vol. 11, no. 3, p. 629, Mar. 2019, doi: 10.3390/w11030629.
 - [47] Z. Hu, B. Chen, W. Chen, D. Tan, and D. Shen, “Review of model-based and data-driven approaches for leak detection and location in water distribution systems,” *Water Supply*, vol. 21, no. 7, pp. 3282–3306, Nov. 2021, doi: 10.2166/ws.2021.101.
 - [48] E. Farah and I. Shahrour, “Water Leak Detection: A Comprehensive Review of Methods, Challenges, and Future Directions,” *Water*, vol. 16, no. 20, p. 2975, Oct. 2024, doi: 10.3390/w16202975.
 - [49] T. H. Shabangu, Y. Hamam, J. A. Jordaan, and K. B. Adediji, “Model-Based Approach for Leak Detection and Localization in Water Distribution Networks: A Literature Survey,” *Int. J. Eng. Res. Africa*, vol. 67, pp. 159–188, Dec. 2023, doi: 10.4028/p-r2lbNg.
 - [50] T. M. Shirajuddin, N. S. Muhammad, and J. Abdullah, “Systematic Review on Research Trends on Sensor-Based Leak Detection Methods in Water Distribution Systems,” *J. Kejuruter.*, vol. 34, no. 2, pp. 201–209, Mar. 2022, doi: 10.17576/jkukm-2022-34(2)-03.
 - [51] S. Yazdekhashti, K. R. Piratla, J. C. Matthews, A. Khan, and S. Atamturktur, “Optimal selection of acoustic leak detection techniques for water pipelines using multi-criteria decision analysis,” *Manag. Environ. Qual. An Int. J.*, vol. 29, no. 2,

- pp. 255–277, Mar. 2018, doi: 10.1108/MEQ-05-2017-0043.
- [52] P. Mohan Doss, M. M. Rokstad, D. Steffebauer, and F. Tscheikner-Gratl, “Uncertainties in different leak localization methods for water distribution networks: a review,” *Urban Water J.*, vol. 20, no. 8, pp. 953–967, Sep. 2023, doi: 10.1080/1573062X.2023.2229301.
 - [53] A. W. Z. Chew, R. Kalfarisi, X. Meng, J. Pok, Z. Y. Wu, and J. Cai, “Acoustic feature-based leakage event detection in near real-time for large-scale water distribution networks,” *J. Hydroinformatics*, vol. 25, no. 2, pp. 526–551, Mar. 2023, doi: 10.2166/hydro.2023.192.
 - [54] P. Agrawal, S. Fong, D. Friesen, and S. Narasimhan, “Leak detection in an operational underground water distribution network using active acoustics,” *J. Acoust. Soc. Am.*, vol. 155, no. 3_Supplement, pp. A333–A333, Mar. 2024, doi: 10.1121/10.0027713.
 - [55] J. Xu *et al.*, “Low-Cost, Tiny-Sized MEMS Hydrophone Sensor for Water Pipeline Leak Detection,” *IEEE Trans. Ind. Electron.*, vol. 66, no. 8, pp. 6374–6382, Aug. 2019, doi: 10.1109/TIE.2018.2874583.
 - [56] S. Tariq, B. Bakhtawar, and T. Zayed, “Data-driven application of MEMS-based accelerometers for leak detection in water distribution networks,” *Sci. Total Environ.*, vol. 809, p. 151110, Feb. 2022, doi: 10.1016/j.scitotenv.2021.151110.
 - [57] B. Zhi, Z. Wu, C. Chen, M. Chen, X. Ding, and L. Lou, “A High Sensitivity AlN-Based MEMS Hydrophone for Pipeline Leak Monitoring,” *Micromachines*, vol. 14, no. 3, p. 654, Mar. 2023, doi: 10.3390/mi14030654.
 - [58] A. Fares, I. A. Tijani, Z. Rui, and T. Zayed, “Leak detection in real water distribution networks based on acoustic emission and machine learning,” *Environ. Technol.*, vol. 44, no. 25, pp. 3850–3866, Nov. 2023, doi: 10.1080/09593330.2022.2074320.
 - [59] R. Vanijirattikhan *et al.*, “AI-based acoustic leak detection in water distribution systems,” *Results Eng.*, vol. 15, p. 100557, Sep. 2022, doi: 10.1016/j.rineng.2022.100557.
 - [60] R. Liu, T. Zayed, and R. Xiao, “Advanced acoustic leak detection in water distribution networks using integrated generative model,” *Water Res.*, vol. 254, p. 121434, May 2024, doi: 10.1016/j.watres.2024.121434.
 - [61] T. Yu, X. Chen, W. Yan, Z. Xu, and M. Ye, “Leak detection in water distribution systems by classifying vibration signals,” *Mech. Syst. Signal Process.*, vol. 185, p. 109810, Feb. 2023, doi: 10.1016/j.ymssp.2022.109810.
 - [62] G. R. Anjana, K. R. S. Kumar, M. S. M. Kumar, and B. Amrutur, “A Particle Filter Based Leak Detection Technique for Water Distribution Systems,” *Procedia Eng.*, vol. 119, pp. 28–34, 2015, doi: 10.1016/j.proeng.2015.08.849.
 - [63] M. N. Jasper, G. (Kumar) Mahinthakumar, S. (Ranji) Ranjithan, and E. D. Brill, “A Sensitivity Analysis of Data Measurement Types for Leak Detection in Water Distribution Systems,” in *World Environmental and Water Resources Congress 2013*, Reston, VA: American Society of Civil Engineers, May 2013, pp. 611–616. doi: 10.1061/9780784412947.059.
 - [64] Z. Hu, W. Chen, D. Tan, B. Chen, and D. Shen, “Multi-objective and risk-based optimal sensor placement for leak detection in a water distribution system,” *Environ. Technol. Innov.*, vol. 28, p. 102565, Nov. 2022, doi:

- 10.1016/j.eti.2022.102565.
- [65] M. I. Rezzwan Radzman *et al.*, “Pipe leakage detection system with artificial neural network,” *IAES Int. J. Artif. Intell.*, vol. 11, no. 3, p. 977, Sep. 2022, doi: 10.11591/ijai.v11.i3.pp977-985.
 - [66] M. N. Jasper, G. (Kumar) Mahinthakumar, S. (Ranji) Ranjithan, and E. D. Brill, “Leak Detection in Water Distribution Systems Using the Dividing Rectangles (DIRECT) Search,” in *World Environmental and Water Resources Congress 2013*, Reston, VA: American Society of Civil Engineers, May 2013, pp. 813–820. doi: 10.1061/9780784412947.078.
 - [67] A. Khalifeh and N. Aweisi, “An Overview on Advancements in Water Leakage Detection Integrating Cutting-Edge Solutions,” in *2024 22nd International Conference on Research and Education in Mechatronics (REM)*, IEEE, Sep. 2024, pp. 364–368. doi: 10.1109/REM63063.2024.10735620.
 - [68] Z. Ahmad Fuad, H. S. Eddy, Y. Badronnisa, and I. Syazwani, “Water leak detection method in water distribution network,” *IOP Conf. Ser. Earth Environ. Sci.*, vol. 357, no. 1, p. 012033, Nov. 2019, doi: 10.1088/1755-1315/357/1/012033.
 - [69] D. Barros, A. Zanfei, A. Menapace, G. Meirelles, M. Herrera, and B. Brentan, “Leak detection and localization in water distribution systems via multilayer networks,” *Water Res. X*, vol. 26, p. 100280, Jan. 2025, doi: 10.1016/j.wroa.2024.100280.
 - [70] J. Son, I. Kim, and J. Park, “Random Network Model for Assessing the Topological Performance of Water Distribution Systems,” *KSCE J. Civ. Eng.*, vol. 27, no. 10, pp. 4101–4114, Oct. 2023, doi: 10.1007/s12205-023-1318-z.
 - [71] J. Deuerlein, A. Wolters, L. Meyer-Harries, and A. R. Simpson, “Graph Decomposition in Risk Analysis and Sensor Placement for Water Distribution Network Security,” in *Water Distribution Systems Analysis 2010*, Reston, VA: American Society of Civil Engineers, Dec. 2011, pp. 394–411. doi: 10.1061/41203(425)38.
 - [72] W. Li *et al.*, “Development of systems for detection, early warning, and control of pipeline leakage in drinking water distribution: A case study,” *J. Environ. Sci.*, vol. 23, no. 11, pp. 1816–1822, Nov. 2011, doi: 10.1016/S1001-0742(10)60577-3.
 - [73] U. Sangroula, K.-H. Han, K.-M. Koo, K. Gnawali, and K.-T. Yum, “Leakage Detection by Using Transient Tests on an Experimental Looped Water Distribution Network.” Jan. 28, 2022. doi: 10.20944/preprints202201.0439.v1.
 - [74] N. C. A. Wilhelmsen and O. M. Aamo, “Leak Detection, Size Estimation and Localization in Water Distribution Networks Containing Loops,” in *2022 IEEE 61st Conference on Decision and Control (CDC)*, IEEE, Dec. 2022, pp. 5429–5436. doi: 10.1109/CDC51059.2022.9993208.
 - [75] F. Boztaş and M. Firat, “Creation and Calibration of Hydraulic Model for Leakage Management in Water Distribution Systems,” *J. Stud. Adv. Technol.*, vol. 2, no. 2, pp. 95–107, Dec. 2024, doi: 10.63063/jsat.1571651.
 - [76] M. Aghashahi, L. Sela, and M. K. Banks, “Benchmarking dataset for leak detection and localization in water distribution systems,” *Data Br.*, vol. 48, p. 109148, Jun. 2023, doi: 10.1016/j.dib.2023.109148.
 - [77] F. Ebisi, I. P. Nikolakakos, J. V. Karunamurthi, A. N. Ahmed Binahmed Alnuaimi, E. Al Buraimi, and S. Alblooshi, “Machine Learning Schemes for Leak Detection

- in IoT-enabled Water Transmission System,” in *2023 International Conference on IT Innovation and Knowledge Discovery (ITIKD)*, IEEE, Mar. 2023, pp. 1–7. doi: 10.1109/ITIKD56332.2023.10100175.
- [78] E. A. Martinez-Ríos, D. Barrientos, and R. Bustamante, “Water Leakage Classification With Acceleration, Pressure, and Acoustic Data: Leveraging the Wavelet Scattering Transform, Unimodal Classifiers, and Late Fusion,” *IEEE Access*, vol. 12, pp. 84923–84951, 2024, doi: 10.1109/ACCESS.2024.3416056.
 - [79] E. A. Martinez-Ríos and R. Bustamante-Bello, “Parameter Selection of Generalized Morse Wavelets for Water Leakage Classification,” in *2024 21st International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, IEEE, Oct. 2024, pp. 1–6. doi: 10.1109/CCE62852.2024.10770991.
 - [80] A. S. Coma, “Robust leak localization in water distribution networks using machine learning techniques,” 2018.
 - [81] J. Luo, C. Wang, J. Yang, and X. Zhong, “A Transformer-Based Approach to Leakage Detection in Water Distribution Networks,” *Sensors*, vol. 24, no. 19, p. 6294, Sep. 2024, doi: 10.3390/s24196294.
 - [82] K. Joseph *et al.*, “Leak and Burst Detection in Water Distribution Network Using Logic- and Machine Learning-Based Approaches,” *Water*, vol. 16, no. 14, p. 1935, Jul. 2024, doi: 10.3390/w16141935.
 - [83] L. Basnet, R. Ranjithan, D. Brill, and K. Mahinthakumar, “Supervised machine learning models for leak detection in water distribution systems,” in *Proceedings - 2nd International Joint Conference on Water Distribution System Analysis (WDSA)& Computing and Control in the Water Industry (CCWI)*, València: Editorial Universitat Politècnica de València, Jul. 2022. doi: 10.4995/WDSA-CCWI2022.2022.14831.
 - [84] V. C. Fuentes and J. R. I. Pedrasa, “Leak Detection in Water Distribution Networks via Pressure Analysis Using a Machine Learning Ensemble,” 2020, pp. 31–44. doi: 10.1007/978-3-030-45293-3_3.
 - [85] N. Sourabh, P. V. Timbadiya, and P. L. Patel, “Leak detection in water distribution network using machine learning techniques,” *ISH J. Hydraul. Eng.*, vol. 29, no. sup1, pp. 177–195, Dec. 2023, doi: 10.1080/09715010.2023.2198988.
 - [86] T. Ma, “Data-Driven Leak Detection and Identification in Water Distribution Networks using Transductive Long Short-Term Memory,” in *2024 Second International Conference on Data Science and Information System (ICDSIS)*, IEEE, May 2024, pp. 1–4. doi: 10.1109/ICDSIS61070.2024.10594295.
 - [87] M. Kammoun, A. Kammoun, and M. Abid, “Experiments based comparative evaluations of machine learning techniques for leak detection in water distribution systems,” *Water Supply*, vol. 22, no. 1, pp. 628–642, Jan. 2022, doi: 10.2166/ws.2021.248.
 - [88] P. Choudhary, A. Modi, B. A. Botre, and S. A. Akbar, “Leak detection in smart water distribution network,” 2021, p. 050007. doi: 10.1063/5.0044005.
 - [89] H. Punukollu, A. Vasan, and K. Srinivasa Raju, “Leak detection in water distribution networks using deep learning,” *ISH J. Hydraul. Eng.*, vol. 29, no. 5, pp. 674–682, Oct. 2023, doi: 10.1080/09715010.2022.2134742.
 - [90] I. Al Balushi and M. Hussain, “Advanced-Data Analytics for Water Loss

- Management and Leakage Detection using Machine Learning Models: A Case Study of Al Seeb Area, Khoudh Six,” *Int. J. Res. Sci. Innov.*, vol. XI, no. XIV, pp. 43–54, 2024, doi: 10.51244/IJRSL.2024.1114SDAICO6.
- [91] G. M. Komba, T. E. Mathonsi, and P. A. Owolawi, “Enhancing Leak Detection Accuracy in Water Distribution Networks Using SVM-CNN-GT Algorithm,” in *2023 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, IEEE, Nov. 2023, pp. 1–7. doi: 10.1109/ICECET58911.2023.10389549.
- [92] H. M. Tornyeviadzi and R. Seidu, “Leakage detection in water distribution networks via 1D CNN deep autoencoder for multivariate SCADA data,” *Eng. Appl. Artif. Intell.*, vol. 122, p. 106062, Jun. 2023, doi: 10.1016/j.engappai.2023.106062.
- [93] H. Yu, S. Lin, H. Zhou, X. Weng, S. Chu, and T. Yu, “Leak detection in water distribution networks based on deep learning and kriging interpolation method,” *AQUA — Water Infrastructure, Ecosyst. Soc.*, vol. 73, no. 8, pp. 1741–1753, Aug. 2024, doi: 10.2166/aqua.2024.184.
- [94] S. Jun, D. Jung, and K. Lansey, “Three-dimensional convolutional neural network for leak detection and localization in smart water distribution systems,” *Water Res. X*, vol. 25, p. 100264, Dec. 2024, doi: 10.1016/j.wroa.2024.100264.
- [95] Y. Wu *et al.*, “Advancing deep learning-based acoustic leak detection methods towards application for water distribution systems from a data-centric perspective,” *Water Res.*, vol. 261, p. 121999, Sep. 2024, doi: 10.1016/j.watres.2024.121999.
- [96] P. Zhang *et al.*, “Water Pipeline Leak Detection Based on a Pseudo-Siamese Convolutional Neural Network: Integrating Handcrafted Features and Deep Representations,” *Water*, vol. 15, no. 6, p. 1088, Mar. 2023, doi: 10.3390/w15061088.
- [97] G. Guo *et al.*, “Leakage Detection in Water Distribution Systems Based on Time–Frequency Convolutional Neural Network,” *J. Water Resour. Plan. Manag.*, vol. 147, no. 2, Feb. 2021, doi: 10.1061/(ASCE)WR.1943-5452.0001317.
- [98] C.-W. Lee and D.-G. Yoo, “Development of Leakage Detection Model and Its Application for Water Distribution Networks Using RNN-LSTM,” *Sustainability*, vol. 13, no. 16, p. 9262, Aug. 2021, doi: 10.3390/su13169262.
- [99] Y. Wu *et al.*, “Hybrid method for enhancing acoustic leak detection in water distribution systems: Integration of handcrafted features and deep learning approaches,” *Process Saf. Environ. Prot.*, vol. 177, pp. 1366–1376, Sep. 2023, doi: 10.1016/j.psep.2023.08.011.
- [100] P. Mohan Doss, M. M. Rokstad, and F. Tscheikner-Gratl, “The performance of encoder–decoder neural networks for leak detection in water distribution networks,” *Water Supply*, vol. 24, no. 8, pp. 2750–2764, Aug. 2024, doi: 10.2166/ws.2024.174.
- [101] D. Tiquio and A. Villanueva, “Pipeline Leak Classification using Machine Learning Ensemble Methods and Hyperparameter Tuning,” in *2024 4th International Conference of Science and Information Technology in Smart Administration (ICSINTESA)*, IEEE, Jul. 2024, pp. 678–683. doi: 10.1109/ICSINTESA62455.2024.10748022.

- [102] L. Sabbatini, M. Esposito, A. Belli, and P. Pierleoni, "Comparison of Signal Pre-processing and Machine Learning Modelling for Water-leak Detection Using Vibration and Pressure Data," in *2024 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, IEEE, Sep. 2024, pp. 1–6. doi: 10.23919/SoftCOM62040.2024.10721866.
- [103] M. Adraoui *et al.*, "A two-phase approach for leak detection and localization in water distribution systems using wavelet decomposition and machine learning," *Comput. Ind. Eng.*, vol. 197, p. 110534, Nov. 2024, doi: 10.1016/j.cie.2024.110534.
- [104] Z. Ahmad, T.-K. Nguyen, and J.-M. Kim, "Leak detection and size identification in fluid pipelines using a novel vulnerability index and 1-D convolutional neural network," *Eng. Appl. Comput. Fluid Mech.*, vol. 17, no. 1, Dec. 2023, doi: 10.1080/19942060.2023.2165159.
- [105] L. Basnet, D. Brill, R. Ranjithan, and K. Mahinthakumar, "Supervised Machine Learning Approaches for Leak Localization in Water Distribution Systems: Impact of Complexities of Leak Characteristics," *J. Water Resour. Plan. Manag.*, vol. 149, no. 8, Aug. 2023, doi: 10.1061/JWRMD5.WRENG-6047.
- [106] A. Momeni and K. R. Piratla, "A Proof-of-Concept Study for Hydraulic Model-Based Leakage Detection in Water Pipelines Using Pressure Monitoring Data," *Front. Water*, vol. 3, Aug. 2021, doi: 10.3389/frwa.2021.648622.
- [107] M. S. Islam, R. Sadiq, M. J. Rodriguez, A. Francisque, H. Najjaran, and M. Hoorfar, "Leakage detection and location in water distribution systems using a fuzzy-based methodology," *Urban Water J.*, vol. 8, no. 6, pp. 351–365, Dec. 2011, doi: 10.1080/1573062X.2011.617829.
- [108] C. V. C. Geelen, D. R. Yntema, J. Molenaar, and K. J. Keesman, "Monitoring Support for Water Distribution Systems based on Pressure Sensor Data," *Water Resour. Manag.*, vol. 33, no. 10, pp. 3339–3353, Aug. 2019, doi: 10.1007/s11269-019-02245-4.
- [109] R. Rayaroth and S. G., "Random Bagging Classifier and Shuffled Frog Leaping Based Optimal Sensor Placement for Leakage Detection in WDS," *Water Resour. Manag.*, vol. 33, no. 9, pp. 3111–3125, Jul. 2019, doi: 10.1007/s11269-019-02296-7.
- [110] R. Vankdothu, H. Bhukya, and R. R. Bhukya, "Hybrid TDR-MI Based Wireless Sensor Network for Underground Water Pipeline Leakage Detection and Localization Using Pressure Residuals and Classifiers," *Wirel. Pers. Commun.*, vol. 139, no. 2, pp. 803–823, Nov. 2024, doi: 10.1007/s11277-024-11623-0.
- [111] X. Li, S. Chu, T. Zhang, T. Yu, and Y. Shao, "Leakage localization using pressure sensors and spatial clustering in water distribution systems," *Water Supply*, vol. 22, no. 1, pp. 1020–1034, Jan. 2022, doi: 10.2166/ws.2021.219.
- [112] M. I. Mohd Ismail *et al.*, "A Review of Vibration Detection Methods Using Accelerometer Sensors for Water Pipeline Leakage," *IEEE Access*, vol. 7, pp. 51965–51981, 2019, doi: 10.1109/ACCESS.2019.2896302.
- [113] Neeraj, P. K. Suri, and V. Yadav, "Design and implementation of wireless sensor network architecture for leak detection and monitoring in water supply distribution network," in *2017 2nd International Conference for Convergence in Technology (I2CT)*, IEEE, Apr. 2017, pp. 559–564. doi: 10.1109/I2CT.2017.8226192.

- [114] R. Qi, M. Cao, and D. R. Yntema, "Recent Developments of Subsurface Small-Leak Detection Techniques in Water Distribution Networks: A Review," *IEEE Robot. Autom. Mag.*, vol. 31, no. 1, pp. 108–118, Mar. 2024, doi: 10.1109/MRA.2024.3351483.
- [115] Y. Yu, A. Safari, X. Niu, B. Drinkwater, and K. V. Horoshenkov, "Acoustic and ultrasonic techniques for defect detection and condition monitoring in water and sewerage pipes: A review," *Appl. Acoust.*, vol. 183, p. 108282, Dec. 2021, doi: 10.1016/j.apacoust.2021.108282.
- [116] Z. Hu, S. Tariq, and T. Zayed, "A comprehensive review of acoustic based leak localization method in pressurized pipelines," *Mech. Syst. Signal Process.*, vol. 161, p. 107994, Dec. 2021, doi: 10.1016/j.ymssp.2021.107994.
- [117] K. Rambabu, S. Dubey, K. N. Srinivas, D. N. Reddy, and C. Rohith, "Smart Water Flow and Pipeline Leakage Detection Using IoT and Arduino UNO," in *2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, IEEE, Jan. 2024, pp. 174–180. doi: 10.1109/IDCIoT59759.2024.10467990.
- [118] A. M. Obeid, F. Karray, M. W. Jmal, M. Abid, S. Manzoor Qasim, and M. S. BenSaleh, "Towards realisation of wireless sensor network-based water pipeline monitoring systems: a comprehensive review of techniques and platforms," *IET Sci. Meas. Technol.*, vol. 10, no. 5, pp. 420–426, Aug. 2016, doi: 10.1049/iet-smt.2015.0255.
- [119] B. Z. Rouso, M. Lambert, and J. Gong, "Smart water networks: A systematic review of applications using high-frequency pressure and acoustic sensors in real water distribution systems," *J. Clean. Prod.*, vol. 410, p. 137193, Jul. 2023, doi: 10.1016/j.jclepro.2023.137193.
- [120] Y. Gao, M. J. Brennan, P. F. Joseph, J. M. Muggleton, and O. Hunaidi, "On the selection of acoustic/vibration sensors for leak detection in plastic water pipes," *J. Sound Vib.*, vol. 283, no. 3–5, pp. 927–941, May 2005, doi: 10.1016/j.jsv.2004.05.004.
- [121] G. Yang and H. Wang, "Optimal Pressure Sensor Deployment for Leak Identification in Water Distribution Networks," *Sensors*, vol. 23, no. 12, p. 5691, Jun. 2023, doi: 10.3390/s23125691.
- [122] E. Raei, M. R. Nikoo, S. Pourshahabi, and M. Sadegh, "Optimal joint deployment of flow and pressure sensors for leak identification in water distribution networks," *Urban Water J.*, vol. 15, no. 9, pp. 837–846, Oct. 2018, doi: 10.1080/1573062X.2018.1561915.
- [123] P. Agrawal, S. Fong, D. Friesen, and S. Narasimhan, "Maximum likelihood estimation for leak localization in water distribution networks using in-pipe acoustic sensing," *J. Acoust. Soc. Am.*, vol. 153, no. 3_supplement, pp. A300–A300, Mar. 2023, doi: 10.1121/10.0018930.
- [124] M. Quinones-Grueiro, C. Verde, and O. Llanes-Santiago, "Multi-objective sensor placement for leakage detection and localization in water distribution networks," in *2019 4th Conference on Control and Fault Tolerant Systems (SysTol)*, IEEE, Sep. 2019, pp. 129–134. doi: 10.1109/SYSTOL.2019.8864746.
- [125] D. Alves, J. Blesa, E. Duviella, and L. Rajaoarisoa, "Multi-leak detection and isolation in water distribution networks," in *Proceedings - 2nd International Join*

- Conference on Water Distribution System Analysis (WDSA)& Computing and Control in the Water Industry (CCWI)*, València: Editorial Universitat Politècnica de València, Jul. 2022. doi: 10.4995/WDSA-CCWI2022.2022.14056.
- [126] E. Raei, M. E. Shafiee, M. R. Nikoo, and E. Berglund, "Placing an ensemble of pressure sensors for leak detection in water distribution networks under measurement uncertainty," *J. Hydroinformatics*, vol. 21, no. 2, pp. 223–239, Mar. 2019, doi: 10.2166/hydro.2018.032.
 - [127] S. Das, A. Tariq, T. Santos, S. S. Kantareddy, and I. Banerjee, "Recurrent Neural Networks (RNNs): Architectures, Training Tricks, and Introduction to Influential Research," 2023, pp. 117–138. doi: 10.1007/978-1-0716-3195-9_4.
 - [128] C. Capinha, A. Ceia-Hasse, A. M. Kramer, and C. Meijer, "Deep learning for supervised classification of temporal data in ecology," *Ecol. Inform.*, vol. 61, p. 101252, Mar. 2021, doi: 10.1016/j.ecoinf.2021.101252.
 - [129] W. Hashim *et al.*, "Optimizing the Clinical Decision Support System (CDSS) by Using Recurrent Neural Network (RNN) Language Models for Real-Time Medical Query Processing," *Comput. Mater. Contin.*, vol. 81, no. 3, pp. 4787–4832, 2024, doi: 10.32604/cmc.2024.055079.
 - [130] N. Manchev and M. Spratling, "Learning multi-modal recurrent neural networks with target propagation," *Comput. Intell.*, vol. 40, no. 4, Aug. 2024, doi: 10.1111/coin.12691.
 - [131] S. M. Al-Selwi *et al.*, "RNN-LSTM: From applications to modeling techniques and beyond—Systematic review," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 36, no. 5, p. 102068, Jun. 2024, doi: 10.1016/j.jksuci.2024.102068.
 - [132] D. Peral-García, J. Cruz-Benito, and F. J. García-Peñalvo, "Systematic literature review: Quantum machine learning and its applications," *Comput. Sci. Rev.*, vol. 51, p. 100619, Feb. 2024, doi: 10.1016/j.cosrev.2024.100619.
 - [133] F. Landi, L. Baraldi, M. Cornia, and R. Cucchiara, "Working Memory Connections for LSTM," *Neural Networks*, vol. 144, pp. 334–341, Dec. 2021, doi: 10.1016/j.neunet.2021.08.030.
 - [134] Z. Zou, X. Yan, Y. Yuan, Z. You, and L. Chen, "Attention mechanism enhanced LSTM networks for latency prediction in deterministic MEC networks," *Intell. Syst. with Appl.*, vol. 23, p. 200425, Sep. 2024, doi: 10.1016/j.iswa.2024.200425.
 - [135] A. Thakkar and K. Chaudhari, "A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions," *Expert Syst. Appl.*, vol. 177, p. 114800, Sep. 2021, doi: 10.1016/j.eswa.2021.114800.
 - [136] S. Wang, C. Shao, J. Zhang, Y. Zheng, and M. Meng, "Traffic flow prediction using bi-directional gated recurrent unit method," *Urban Informatics*, vol. 1, no. 1, p. 16, Dec. 2022, doi: 10.1007/s44212-022-00015-z.
 - [137] S. Khan *et al.*, "Comparative analysis of deep neural network architectures for renewable energy forecasting: enhancing accuracy with meteorological and time-based features," *Discov. Sustain.*, vol. 5, no. 1, p. 533, Dec. 2024, doi: 10.1007/s43621-024-00783-5.
 - [138] E. Oluwasakin *et al.*, "Minimization of high computational cost in data preprocessing and modeling using MPI4Py," *Mach. Learn. with Appl.*, vol. 13, p. 100483, Sep. 2023, doi: 10.1016/j.mlwa.2023.100483.
 - [139] M. M. Abdelgwad, T. H. A. Soliman, A. I. Taloba, and M. F. Farghaly, "Arabic

- aspect based sentiment analysis using bidirectional GRU based models,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 9, pp. 6652–6662, Oct. 2022, doi: 10.1016/j.jksuci.2021.08.030.
- [140] M. Trabelsi, Z. Chen, B. D. Davison, and J. Heflin, “Neural ranking models for document retrieval,” *Inf. Retr. J.*, vol. 24, no. 6, pp. 400–444, Dec. 2021, doi: 10.1007/s10791-021-09398-0.
- [141] L. W. Rizkallah, “Enhancing the performance of gradient boosting trees on regression problems,” *J. Big Data*, vol. 12, no. 1, p. 35, Feb. 2025, doi: 10.1186/s40537-025-01071-3.
- [142] H. Ahmetoglu and R. Das, “A comprehensive review on detection of cyber-attacks: Data sets, methods, challenges, and future research directions,” *Internet of Things*, vol. 20, p. 100615, Nov. 2022, doi: 10.1016/j.iot.2022.100615.
- [143] J. Tanha, Y. Abdi, N. Samadi, N. Razzaghi, and M. Asadpour, “Boosting methods for multi-class imbalanced data classification: an experimental review,” *J. Big Data*, vol. 7, no. 1, p. 70, Dec. 2020, doi: 10.1186/s40537-020-00349-y.
- [144] T.-T.-H. Le, S. Heo, J. Cho, and H. Kim, “DDoSBERT: Fine-tuning variant text classification bidirectional encoder representations from transformers for DDoS detection,” *Comput. Networks*, vol. 262, p. 111150, May 2025, doi: 10.1016/j.comnet.2025.111150.
- [145] N. Gunasekara, B. Pfahringer, H. Gomes, and A. Bifet, “Gradient boosted trees for evolving data streams,” *Mach. Learn.*, vol. 113, no. 5, pp. 3325–3352, May 2024, doi: 10.1007/s10994-024-06517-y.
- [146] H. Shi, H. Wang, Y. Huang, L. Zhao, C. Qin, and C. Liu, “A hierarchical method based on weighted extreme gradient boosting in ECG heartbeat classification,” *Comput. Methods Programs Biomed.*, vol. 171, pp. 1–10, Apr. 2019, doi: 10.1016/j.cmpb.2019.02.005.
- [147] A. A. Khan, O. Chaudhari, and R. Chandra, “A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation,” *Expert Syst. Appl.*, vol. 244, p. 122778, Jun. 2024, doi: 10.1016/j.eswa.2023.122778.
- [148] T. Chen and C. Guestrin, “XGBoost,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [149] G. Ke *et al.*, “LightGBM: A Highly Efficient Gradient Boosting Decision Tree,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf
- [150] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, “A comparative analysis of gradient boosting algorithms,” *Artif. Intell. Rev.*, vol. 54, no. 3, pp. 1937–1967, Mar. 2021, doi: 10.1007/s10462-020-09896-5.
- [151] J. Korstanje, “Gradient Boosting with XGBoost and LightGBM,” in *Advanced Forecasting with Python*, Berkeley, CA: Apress, 2021, pp. 193–205. doi: 10.1007/978-1-4842-7150-6_15.
- [152] W. Gong *et al.*, “Gradient boosting decision tree algorithms for accelerating

- nanofiltration membrane design and discovery,” *Desalination*, vol. 592, p. 118072, Dec. 2024, doi: 10.1016/j.desal.2024.118072.
- [153] J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, Apr. 2001, [Online]. Available: <http://www.jstor.org/stable/2699986>
- [154] H. Alshari, A. Y. Saleh, and A. Odaba\cs, “Comparison of gradient boosting decision tree algorithms for CPU performance,” *J. Institue Sci. Technol.*, vol. 37, no. 1, pp. 157–168, 2021.
- [155] L. Breiman, “Random Forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
- [156] G. Biau, “Analysis of a Random Forests Model,” *J. Mach. Learn. Res.*, vol. 13, no. 38, pp. 1063–1095, 2012, [Online]. Available: <http://jmlr.org/papers/v13/biau12a.html>
- [157] C. Musanase, A. Vodacek, D. Hanyurwimfura, A. Uwitonze, and I. Kabandana, “Data-Driven Analysis and Machine Learning-Based Crop and Fertilizer Recommendation System for Revolutionizing Farming Practices,” *Agriculture*, vol. 13, no. 11, p. 2141, Nov. 2023, doi: 10.3390/agriculture13112141.

Vitae

Name : Ahmad Fares Al-Nouti
Nationality : Jordanian
Date of Birth : 12/30/1999
Email : afalnouti@gmail.com
Address : Kufr Saum 21941, Bani Kenanah, Irbid, Jordan
Academic Background : Master of Science in Civil Engineering.

As a Master of Science candidate in Civil Engineering with a specialization in Water and Environmental Engineering, I have actively led research focused on the application of machine learning and intelligent data-driven models to solve complex problems in environmental systems and water resources management. I am the first author of several peer-reviewed journal articles. My publications include a comprehensive review titled “Reservoir Operation Based Machine Learning Models: Comprehensive Review for Limitations, Research Gap, and Possible Future Research Direction” in the Knowledge-Based Engineering and Sciences (KBES) Journal. I also authored “Integration of Hybrid Machine Learning Model with Differential Evolution Algorithm for Underground Pipeline Corrosion Prediction”, published in the Engineering Applications of Artificial Intelligence (EAAI) Journal. Furthermore, my recent work, “On the Prediction of Plant Root Uptake Emergent Contaminants: From Laboratory Experiment to Data-Intelligent Modelling”, featured in the Microchemical Journal.