

# INTRUSION DETECTION SYSTEM





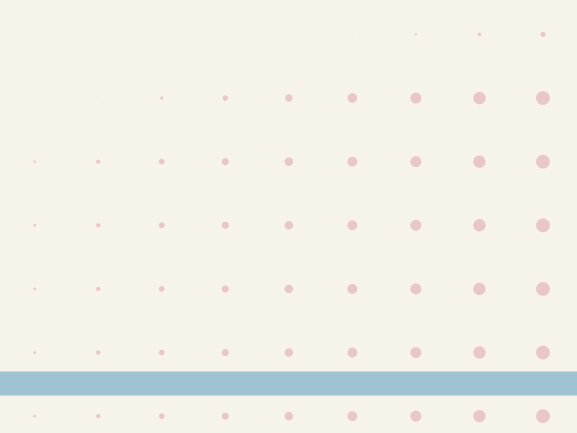
# ABSTRACT

- Cybersecurity is a critical area as digital infrastructures grow.

## **Traditional IDS (Intrusion Detection Systems) struggle with:**

- Imbalanced datasets
- Difficulty in identifying unknown attack types
- Lack of transparency in ML decisions

## **Our proposed system:**

- Uses Autoencoders for unsupervised anomaly detection
  - Employs Random Forests for multi-class classification
  - Applies SMOTE to handle data imbalance
  - Integrates LIME to make model predictions interpretable
  - Features a React-based real-time dashboard for user interaction
- 

# INTRODUCTION

## What is IDS?

- Intrusion Detection Systems are tools that monitor network or system activities for malicious actions.

## Two main types:

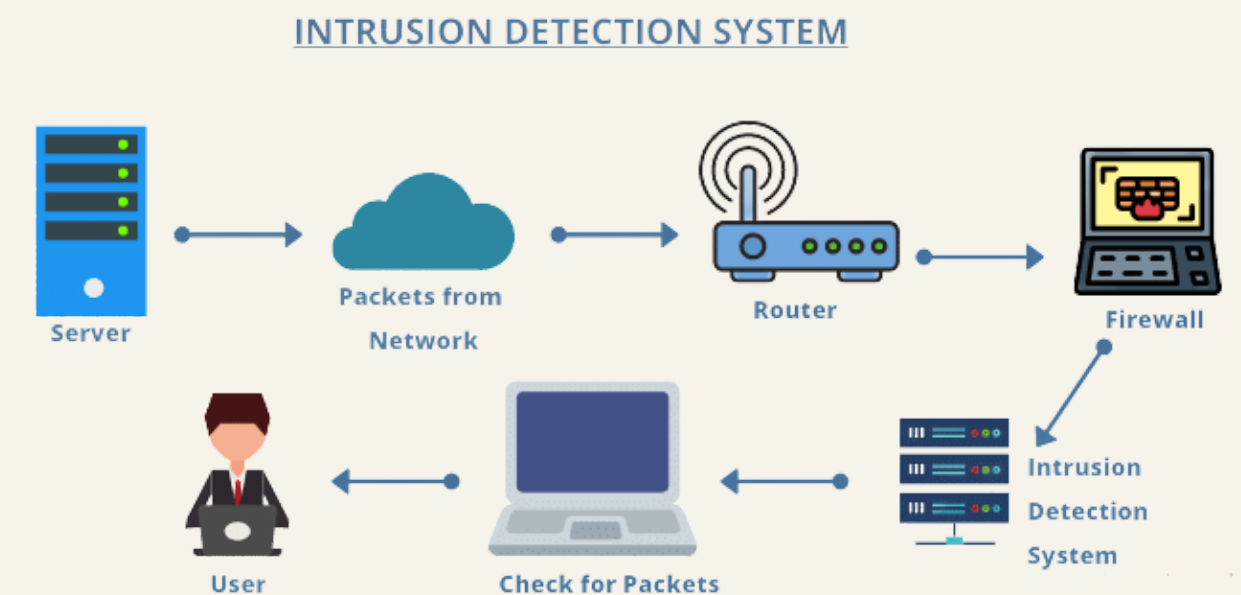
- Signature-based IDS: Detects known patterns
- Anomaly-based IDS: Detects deviations from normal behaviour

## Challenges in IDS:

- High false positives in anomaly-based systems
- Poor generalisation to unknown attack types
- The black-box nature of ML models causes trust issues

## Our Contribution:

- A hybrid system combining anomaly detection and supervised classification
- Real-time risk monitoring with visual explanations





# TOOLS & TECHNOLOGIES USED

## **Dataset: CICIDS-2018**

- Contains realistic network traffic with normal and various attack types (DDoS, BruteForce, PortScan, etc.)

## **Preprocessing Tools:**

- Pandas, Scikit-learn, Imbalanced-learn (SMOTE)

## **ML Models:**

- Autoencoder (Keras/TensorFlow)
- Random Forest Classifier (Scikit-learn)
- LIME (Interpretability, Local Model Explanation)

## **Frontend Development:**

- React.js (Real-time dashboard)
- 



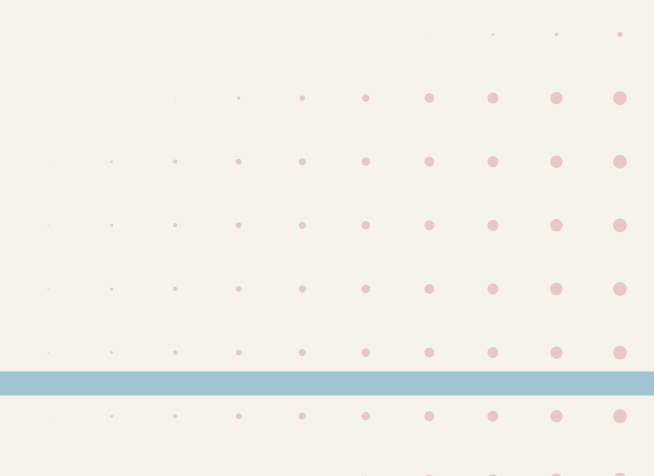
# Dataset Description – CICIDS-2018

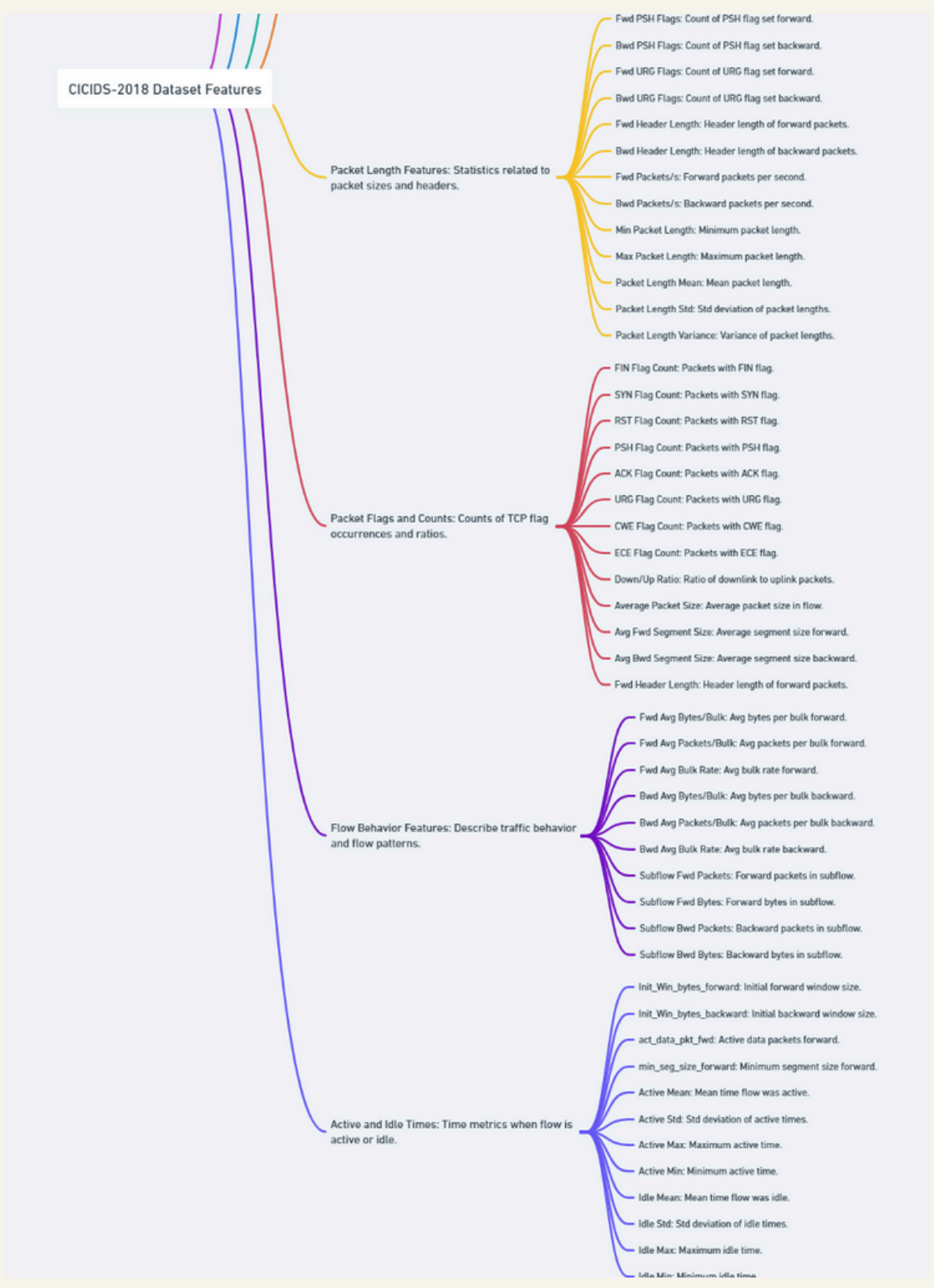
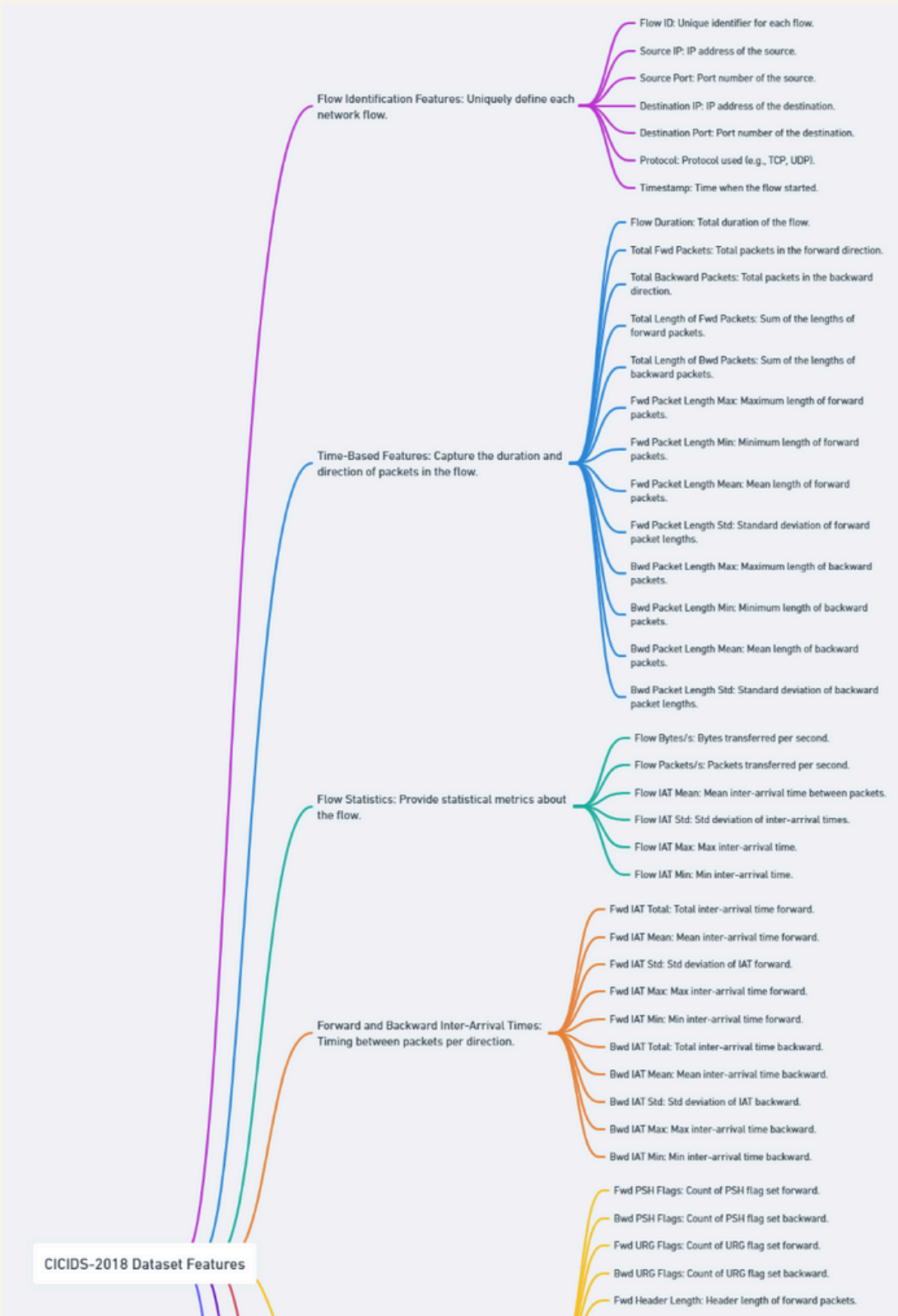
## About the Dataset:

- **Developed by:** Canadian Institute for Cybersecurity (CIC)
- **Purpose:** Benchmark dataset for evaluating Intrusion Detection Systems (IDS)
- Captures real-world traffic with up-to-date attack types

## Dataset Details

Attribute	Details
Total Records	~16 million flow-based network records
Duration	10 days of traffic (June 2–15, 2018)
Data Format	CSV (after preprocessing from PCAPs)
Total Features	<b>80 features + 1 label column</b>
Label Classes	Normal + 14 attack types







Label	Attack Category	Description
BENIGN	Normal Traffic	Legitimate, non-malicious network activity.
FTP-Patator	Brute Force	Brute-force login attempts on an FTP server using tools like Patator.
SSH-Patator	Brute Force	Brute-force login attempts on SSH (Secure Shell) service using Patator.
DoS Hulk	DoS (Denial of Service)	Overwhelms the web server by flooding it with HTTP GET/POST requests.
DoS GoldenEye	DoS	Sends large volumes of HTTP requests to exhaust server resources.
DoS slowloris	DoS	Sends partial HTTP headers slowly to keep connections open and exhaust threads.
DoS Slowhttptest	DoS	Simulates slow HTTP attacks to exhaust server resources using few connections.
Heartbleed	Exploit	Exploits a vulnerability in OpenSSL to read sensitive memory from the server.
Web Attack – Brute Force	Web-Based Attack	Repeated login attempts on web applications (e.g., admin login pages).
Web Attack – XSS	Web-Based Attack	Cross-site scripting attacks via input fields to inject malicious JavaScript.
Web Attack – Sql Injection	Web-Based Attack	Attempts to inject SQL queries via web inputs to manipulate database queries.
Infiltration	Infiltration	Intruder gains access from outside and performs internal scanning or control.
Bot	Botnet	System acts as a zombie (part of a botnet) under the attacker's control.
PortScan	Reconnaissance	Scanning of multiple ports to identify open/active services.
DDoS	Distributed DoS	Large-scale coordinated attack from multiple sources to flood the target.



# LIME(LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS)

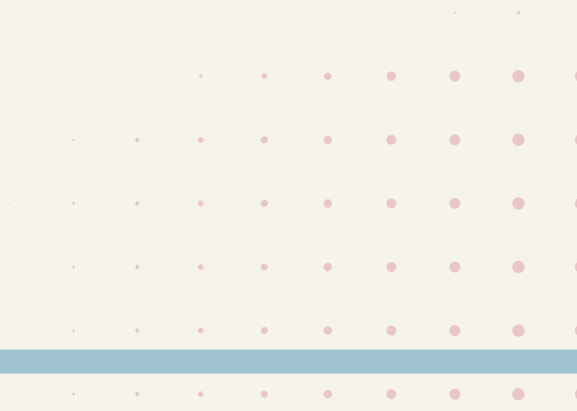
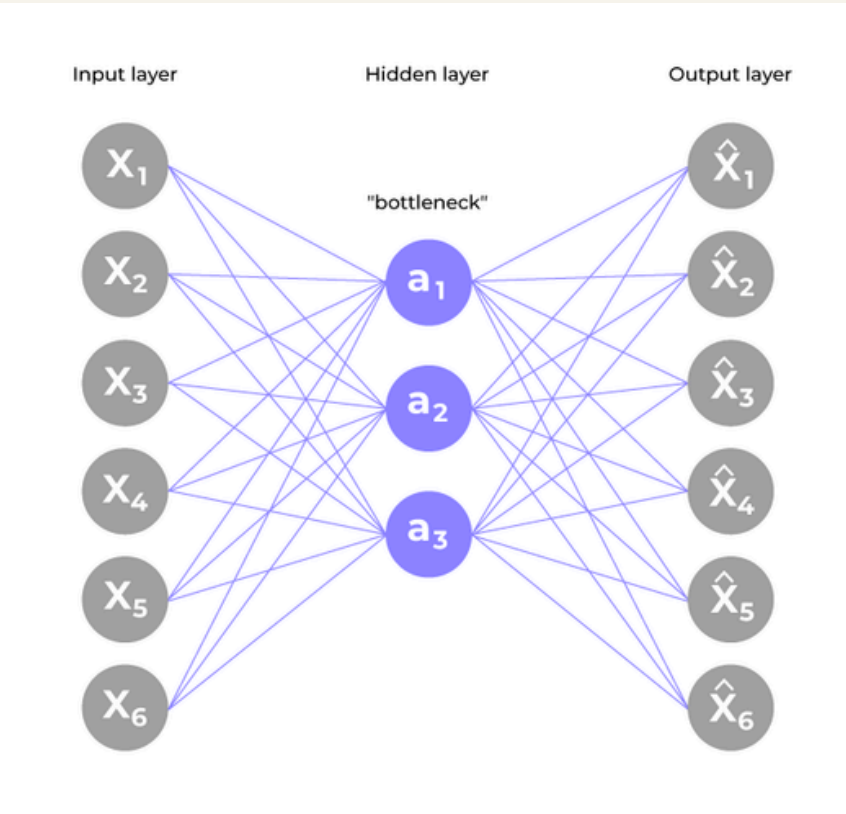
LIME Means	Like This
Local	It explains just <b>one prediction</b> , not the whole model.
Interpretable	It gives an <b>easy-to-understand</b> reason.
Model-agnostic	It works with <b>any model</b> – decision trees, neural networks, anything
Explanation	It tells you <b>why</b> the model made a decision.

## LIME Makes Tiny Changes

- LIME takes your original message:
- "Win ₹10000 now!"
- Then makes many small changes, like:
- "Win ₹10000"
- "₹10000 now"
- "Win now"
- "Free ₹10000 now!"

These are called **perturbations**.

LIME shows these changed messages to the model again and again





# LITERATURE SURVEY

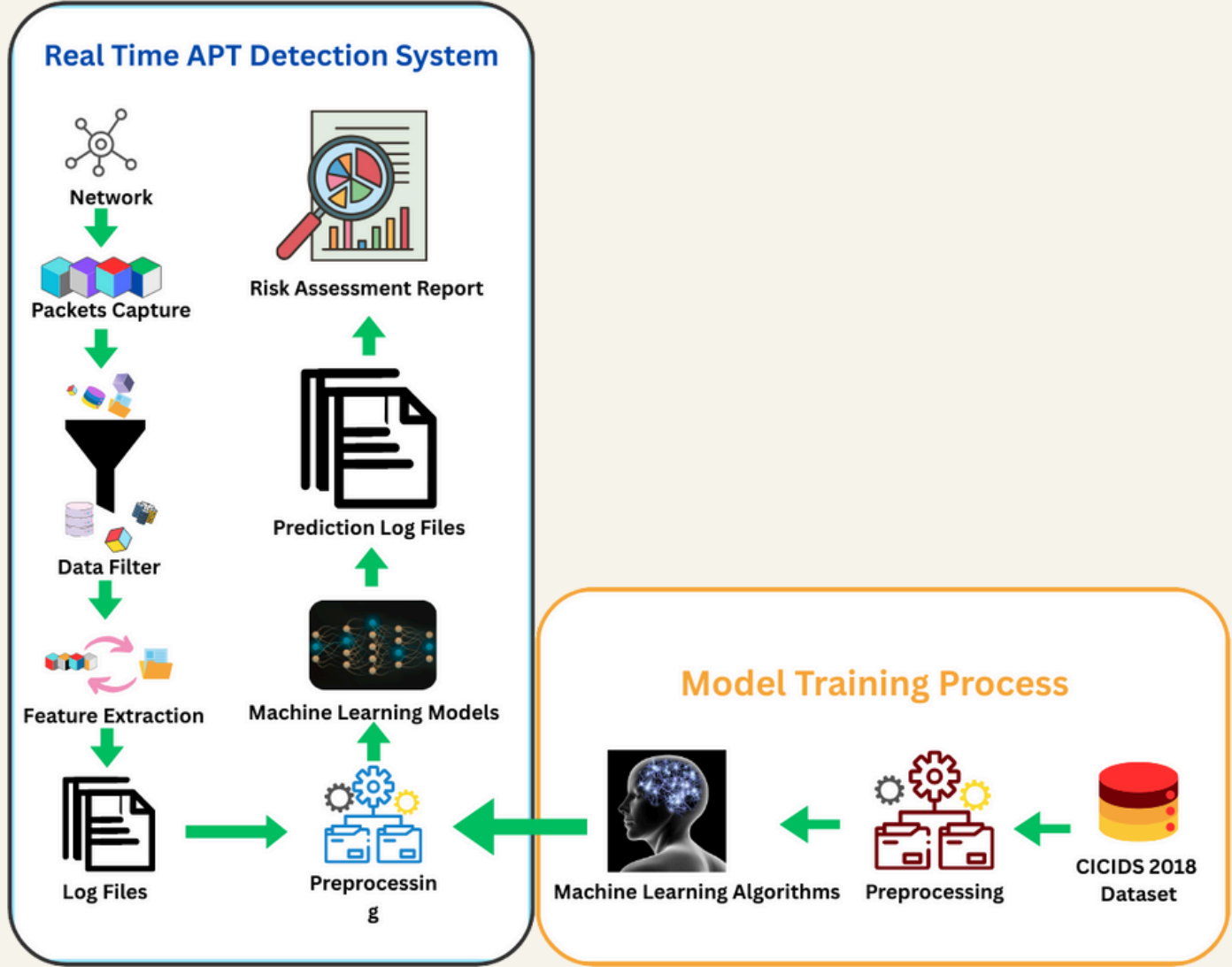
Paper Title (with source)	Year	Method Used	What They Did	Weaknesses (vs. user’s project)	Accuracy (if given)
“Intrusion detection system based on ML using least square SVM”	2025	Quantum-inspired Least Squares SVM (LS-SVM)	Developed a supervised IDS by applying an LS-SVM classifier (with exhaustive feature selection) on three benchmark datasets (NSL-KDD, CIC-IDS-2017, UNSW-NB15)	No anomaly-detection (only classification), no explainability tools (no LIME/SHAP), and no front-end/dashboard support – just offline model training.	NSL-KDD: 99.3%; CIC-IDS-2017: 99.5%; UNSW-NB15: 93.3%
“IDS based on improved Random Forest with double feature selection” <a href="https://www.tasr.scione.com">tasr.scione.com</a>	2024	Random Forest (+ “double” feature selection)	Used an enhanced RF classifier with two-stage feature selection (variance threshold + score-based selection) on NSL-KDD <a href="https://www.tasr.scione.com">tasr.scione.com</a> .	Only classification on one dataset, no anomaly detection stage, no model explanations, and no interactive UI – limited to offline detection.	99.81% (on NSL-KDD) <a href="https://www.tasr.scione.com">tasr.scione.com</a>
“Enhanced IDS performance with UNSW-NB15 data analysis” <a href="https://www.mdpi.com">mdpi.com</a>	2024	Random Forest (plus LR, SVM, DT)	Evaluated several ML models (logistic regression, SVM, decision tree, Random Forest) on UNSW-NB15 with correlation-based feature selection <a href="https://www.mdpi.com">mdpi.com</a> .	Focused on classification only (no separate anomaly detection), did not use interpretability tools, and provided no real-time or dashboard component – purely offline analysis.	98.63% (UNSW-NB15) <a href="https://www.mdpi.com">mdpi.com</a>
“Network intrusion detection using RF with Extra Trees” <a href="https://www.romanpub.com">romanpub.com</a>	2022	Random Forest (with Extra Trees)	Built a multi-class IDS on the UNSW-NB15 dataset using a Random Forest augmented by an Extra-Tree-based component (for feature evaluation) <a href="https://www.romanpub.com">romanpub.com</a> .	Limited to basic classification (no anomaly-detection module), no explainability techniques, and no UI/dashboard – just standard offline training on one dataset.	96.2% (UNSW-NB15) <a href="https://www.romanpub.com">romanpub.com</a>
“NSL-KDD dataset with Naive Bayes algorithm” <a href="https://www.internationalpubs.com">internationalpubs.com</a>	2025	Naïve Bayes (with logistic regression)	Tested a simple Naïve Bayes (and logistic) classifier on NSL-KDD data <a href="https://www.internationalpubs.com">internationalpubs.com</a> (no data-imbalance handling).	Very basic approach – only 85% accuracy, no anomaly-detection or advanced learning, no interpretability, no live monitoring or dashboards – much weaker than hybrid IDS.	85% (NSL-KDD) <a href="https://www.internationalpubs.com">internationalpubs.com</a>

# SUMMARY OF LITERATURE SURVEY

Feature / Aspect	our Project	Other Papers (Reviewed)
Hybrid Model	✔ Anomaly Detection + Classification	✗ Only basic classification (no hybrid detection)
Data Imbalance Handling	✔ Used sampling techniques (e.g., SMOTE)	✗ Mostly ignored imbalance issue
Explainability (LIME/SHAP)	✔ Added interpretability layer	✗ No use of explainable AI tools
UI / Dashboard	✔ Real-time interface for monitoring	✗ Offline model only, no dashboard or interaction
Dataset Variety	✔ Tested on 2+ datasets (e.g., NSL-KDD, CIC-IDS)	⚠ Most used only one dataset (usually NSL-KDD or UNSW-)
Accuracy	✔ High (typically >98% with balanced focus)	⚠ Some are high, but lacking robustness (e.g., no anomaly
Innovation Level	✔ Multi-stage, real-world inspired architecture	✗ Simple ML approaches with limited deployment readiness

# DESIGN (SYSTEM ARCHITECTURE)

Component	Description
1. Real-Time Data Input	Capture live network data streams or logs continuously from the environment.
2. Preprocessing & Imbalance Handling	Clean data, extract features, and use SMOTE to balance data classes before detection.
3. Hybrid ML Model	Apply a combination of anomaly detection and supervised classification ( Random Forest ) for accurate detection.
4. Explainability Module	Use LIME to generate human-understandable explanations for each detected intrusion or anomaly.
5. Result Visualization Dashboard	Show real-time detection results, attack type, confidence score, and explanation summary in an interactive UI.





# METHODOLOGY

## **Data Collection & Preprocessing:**

- Load and clean the CICIDS-2018 dataset, handling missing values , converting categorical to numeric
- Normalise features (Min–Max scaling) and remove irrelevant columns to streamline inputs.

## **Class Imbalance Handling:**

- Split data into training and test sets (80/20).
- Apply SMOTE to the training set to generate synthetic samples for underrepresented attack classes.

## **Anomaly Detection (Autoencoder):**

- Train an unsupervised deep autoencoder on benign traffic to learn normal behaviour.
- Flag flows as anomalies when reconstruction error exceeds a threshold.

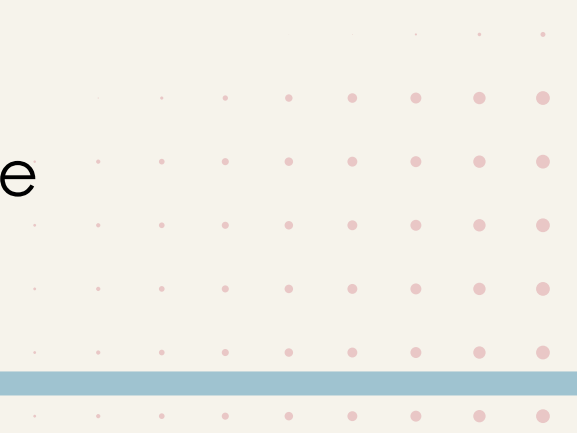
## **Attack Classification (Random Forest):**

- Feed anomalies into a Random Forest classifier trained on balanced, labelled attack data.
- Configure with 100 trees, max depth 30, balanced class weights for robust multi-class prediction.

## **Explainability (LIME):**

- For each classified event, use LIME to identify and rank the top contributing features.
- Generate human-readable explanations to build analyst trust and support audits.

## **Real-Time Visualisation (React Dashboard):**

- Expose a Flask API serving predictions and LIME explanations.
  - Display live alerts, risk scores, feature contributions, and metadata (IPs, timestamps) in an interactive React Interface
- 





# ANALYSIS & COMPARATIVE STUDY

Aspect	Our Project (Proposed System)	Basic/Other IDS Papers
Dataset	CICIDS-2018 (modern, diverse attack types)	Often older/smaller datasets or no public datasets
Handling Imbalance	Uses SMOTE to balance data for better detection accuracy	Basic classifiers with no or simple imbalance handling
Anomaly Detection	Autoencoder for detecting unknown attacks without labels	Mostly rule-based or supervised methods requiring labeled data
Classification	Random Forest for multi-class attack classification + risk score	Simple classifiers like decision trees or SVM without scoring
Interpretability	LIME Explainer provides explanation for each classification	Little or no interpretability/explanation methods
Real-time Capability	React dashboard for live monitoring and explanations	Mostly offline analysis or static reports

## Key\_points:

- Your system uses advanced imbalance handling (SMOTE) and unsupervised anomaly detection (Autoencoder), making it more robust.
- Adding LIME increases trust by explaining decisions, missing in simpler systems.
- Real-time visualization makes it practical for actual deployment compared to static outputs.





# CONCLUSION & FUTURE WORK

This IDS project improves over basic systems by combining modern techniques to handle real-world challenges:

- SMOTE effectively tackles data imbalance, improving detection rates.
- Autoencoder detects novel, unknown attacks without relying solely on labeled data.
- Random Forest accurately classifies attack types and assigns meaningful risk scores.
- LIME ensures transparency, helping users understand why alerts are raised.
- The React dashboard enables real-time monitoring, crucial for timely security responses.

Overall, this system offers a practical, interpretable, and adaptive solution for intrusion detection, suitable for dynamic network environments.



The background features three vertical stripes on the left: a wide red stripe, a medium blue stripe, and a narrow beige stripe. The rest of the background is a light beige color with two rectangular areas of a pink dot pattern, one in the top right and one in the bottom right.

**THANK YOU**