Unifying Three IoMT Intrusion Detection Datasets: Taxonomy and Preprocessing

Mahendra Singh Panwar[1], Dr. Jai Singh Gupta[2], Dr. Akash Saxena[3], Dr. Mohini Dwivedi[4]

Research Scholar[1], Professor[2], Professor[3] , Associate Professor[4]

Department of Computer Science, Shridhar University, Pilani, Rajasthan, India[1,2,4]

Department of Computer Science and Engineering, Compucom Institute of Technology and Management, Jaipur, Rajasthan, India[3]

panwarpanwar80@gmail.com , jaigupta2@gmail.com , akash27jaipur@gmail.com, mohini.ceeri@gmail.com

Abstract

The Internet of Multimedia Things (IoMT) is connecting cameras, microphones, medical sensors, and smart devices at a scale that existing security tools were simply not built for. Detecting network intrusions in these environments is hard because of the huge volume of multimedia traffic, the diversity of devices, and the sensitivity of the data — especially in healthcare. This paper presents the first two steps of a broader research effort: selecting and combining three publicly available datasets — UNSW-NB15, TON_IoT, and CICIoT2023 — into a unified training corpus for a machine learning based intrusion detection system (IDS), and preprocessing that corpus into a clean, model-ready form. The combined dataset has 5,751,048 network flow records, 124 features, and covers 17 attack categories including Exploit, DDoS, Ransomware, Mirai, and Reconnaissance. A ten-step preprocessing pipeline handles 322,750,014 missing values, resolves schema differences across the three datasets, unifies 60+ raw attack labels into a consistent taxonomy, and produces stratified 80/20 train/test splits. The paper documents design decisions, challenges encountered, and the reasoning behind each technical choice in transparent detail so that researchers can reproduce or build upon this work.

Keywords: *Internet of Multimedia Things (IoMT), Intrusion Detection System (IDS), UNSW-NB15, TON_IoT, CICIoT2023, network security, dataset preprocessing, attack taxonomy, feature engineering, machine learning*

1. Introduction

The word 'Internet of Things' gets used so often that it can start to feel abstract. But what it actually means, in hospitals and smart cities and factories right now, is this: thousands of devices are connected to the internet that were not connected five years ago. IP cameras. Wearable heart monitors. Smart insulin pumps. Voice-controlled ward assistants. Environmental sensors monitoring air quality in operating theatres. All of them are sending data across networks, and almost none of them were designed with security as a primary concern [4].

The Internet of Multimedia Things extends this further by adding audio, video, and imaging data into the mix [3]. A hospital today might stream live video from a dozen monitoring cameras, transmit diagnostic imaging data from MRI machines to remote radiologists, and collect continuous biometric audio from patients in intensive care — all simultaneously, all across the same network. The volume of data involved is orders of magnitude larger than what conventional IoT sensor networks produce. And the security consequences of failure are more immediate. A manipulated sensor reading from an industrial IoT system is bad. A tampered monitoring feed from a patient's cardiac device can be dangerous [1].

Intrusion detection systems are the network security community's response to this problem. They watch network traffic and raise alerts when something looks wrong. The challenge is building one that actually works for IoMT environments — fast enough to process high-bandwidth multimedia traffic in real time, accurate enough to catch attacks that try to blend in with legitimate traffic, and general enough to detect attack types that were not in its training data [13]. Getting the training data right is a big part of solving that challenge.

Most published IDS research papers train and test on a single dataset. The most popular choice is UNSW-NB15, which was created in 2015 at the University of New South Wales and has become the standard benchmark for

Doi:10.48047/ijiee.2025.15.7.20

network intrusion detection research [6]. It is a good dataset. But it has gaps. It does not contain ransomware traffic. It does not contain Mirai botnet attacks. Both of these are among the most actively used attack techniques against IoT and healthcare networks right now [8], [9]. A model trained only on UNSW-NB15 has never seen them.

This paper addresses that gap by combining three datasets — UNSW-NB15, TON_IoT, and CICIoT2023 — into a single training corpus. The combination is deliberate. Each dataset covers attack types that the others do not. Together they give a much more complete picture of the real threat landscape facing IoMT deployments. This paper documents the process of selecting these datasets, understanding what they contain, resolving the differences between them, and preprocessing the result into a form that machine learning models can actually use.

The contributions of this paper are: (1) a principled justification for selecting these three specific datasets and their coverage of 17 distinct attack categories; (2) a unified attack taxonomy that maps 60+ inconsistent raw labels across three datasets to a single consistent naming scheme; (3) a documented ten-step preprocessing pipeline that produces a 5.75 million record, 124-feature training corpus from three heterogeneous data sources; and (4) a transparent account of the practical challenges encountered during this process, including a memory crash, a labelling problem in CICIoT2023, and an 18-minute processing time.

2. Related Work

The literature on intrusion detection for IoT and IoMT environments is large and growing fast. Several themes emerge when you look across this body of work as a whole.

The most used dataset in the field is UNSW-NB15 [6]. Moustafa and Slay generated it in 2015 using the IXIA PerfectStorm tool in a dedicated cyber range, capturing 2.5 million labelled network flow records with 49 features across nine attack categories. It remains the most cited IDS benchmark in the literature more than a decade after its creation. Moualla et al. [10] tested five classifiers on UNSW-NB15 — Logistic Regression, Decision Trees, Random Forest, SVM, and MLP — and reported Random Forest as the top performer for binary classification. Their work is a useful baseline, but it has a limitation that the authors themselves acknowledge: the results apply only to the UNSW-NB15 traffic distribution. How well these models generalise to actual IoT device traffic is unknown.

TON_IoT was created by the same group at UNSW in 2020 specifically to address this gap [8]. It uses a realistic IoT testbed with actual IoT hardware and captures not just network flows but also operating system logs and device telemetry. It adds attack types that UNSW-NB15 does not cover: ransomware, injection attacks, and credential-stuffing attempts against IoT management interfaces. Alsaedi et al. [8] documented the dataset and demonstrated its value for federated learning scenarios. It is now widely used alongside UNSW-NB15 in multi-dataset IDS studies.

CICIoT2023 [9] is the newest of the three datasets used in this paper. Neto et al. created it using 105 commercially available IoT devices — the largest real-device testbed in any publicly released IDS dataset — and generated 33 distinct attack sub-types including Mirai botnet variants, ARP spoofing, and slow-rate DoS attacks. The dataset captures traffic patterns that can only come from real device hardware, including the unusual connection timing and protocol behaviour that distinguishes genuine IoT devices from simulated traffic.

Despite the availability of these datasets, most published work evaluates on only one of them. Salehpour et al. [12] combined Random Forest with XGBoost and ADASYN oversampling for IoMT-specific detection, but used only an IoMT-specific private dataset. Afroz et al. [13] proposed an RF-plus-DNN hybrid for IoMT traffic analysis but evaluated on IoMT traffic flows without publicly documented dataset combination methodology. The multi-dataset approach taken in this paper —

Doi:10.48047/ijiee.2025.15.7.20

combining UNSW-NB15, TON_IoT, and CICIoT2023 — appears to be the first published study to explicitly combine all three and document the taxonomy unification and preprocessing process in detail.

3. Datasets Selected for This Study

Three datasets were selected. The selection was not arbitrary — each fills a specific gap that the other two leave open. Table 1 summarises their key properties after downloading and before preprocessing.

Table 1: Summary of the Three Datasets Used — Before Preprocessing

Dataset	Year	Records	After Dedup	Features	Attack Types
UNSW-NB15	2015	257,673	136,062	45	9 (DoS, Exploit, Fuzzer, Recon, Backdoor, Generic, Shellcode, Worm, Analysis)
TON_IoT	2020	211,043	182,998	44	9 (DoS, DDoS, Injection, BruteForce, Backdoor, Recon, Ransomware, XSS, Spoofing)
CICIoT2023	2023	7,845,673	5,431,988	47	33 sub-types (mapped to 1 generic Attack class in this Kaggle release)
Combined Master	**5,751,048 rows × 124 features | 17 unified attack classes | 80/20 stratified split**				

3.1 UNSW-NB15 . The Benchmark Foundation

UNSW-NB15 was downloaded as two separate files: a training set (175,341 rows) and a testing set (82,332 rows). Combined, they give 257,673 rows with 45 columns. The attack type is stored in a column called attack_cat with values like 'DoS', 'Exploits', 'Fuzzers', and so on. A binary label column (0 for normal, 1 for attack) is also provided [6].

The dataset's age is both its strength and its limitation. Its strength is that it has been studied by hundreds of researchers, which means its properties are well-understood and published baselines are abundant [10], [11]. Its limitation is that it was generated in 2015, before Mirai, before modern ransomware campaigns against hospitals, and before the widespread deployment of consumer IoT devices in healthcare settings. It covers classic attack types well. It does not cover what is happening in real IoMT networks in 2024.

3.2 TON_IoT Real IoT Device Telemetry

TON_IoT arrived as a single file: train_test_network.csv with 211,043 rows and 44 columns. The attack type column here is called 'type' rather than 'attack_cat' — this kind of naming inconsistency across datasets is exactly the kind of problem the preprocessing pipeline has to handle. The attack types include Ransomware, Injection, XSS, and Spoofing — none of which appear in UNSW-NB15 [8].

What makes TON_IoT particularly valuable for this research is its origin. The traffic was captured from a testbed that included real IoT hardware — smart home devices, industrial sensors, and network monitoring

Doi:10.48047/ijiee.2025.15.7.20

equipment — rather than simulated traffic from a traffic generator. The resulting flow patterns reflect how real IoT devices actually communicate, including the irregular timing and protocol behaviour that pure simulation cannot reproduce. Ransomware traffic in particular is a critical addition. Healthcare organisations have been specifically targeted by ransomware operators, and TON_IoT is the only publicly available IDS dataset that includes labelled ransomware network traffic [8].

3.3 CICIoT2023 Modern Threats from Real Devices

CICIoT2023 came in three parts: a training CSV (5,491,971 rows), a test CSV (1,176,851 rows), and a validation CSV (1,176,851 rows). The full dataset has 7,845,673 rows with 47 columns — by far the largest of the three [9].

The attack traffic was generated from 105 real IoT devices in a purpose-built testbed at the Canadian Institute for Cybersecurity. The devices included smart speakers, IP cameras, smart home hubs, and IoT gateways. All attacks were carried out by compromised IoT devices targeting other IoT devices on the same network — exactly the lateral movement scenario that makes IoT security so difficult. The dataset covers Mirai botnet attacks, which are not in either UNSW-NB15 or TON_IoT. Mirai and its variants have been responsible for some of the largest DDoS events ever recorded and continue to be a primary threat to IoT deployments [9].

One important note about this dataset: the label column in the Kaggle-distributed version of CICIoT2023 used in this study stores all attack records under a single generic label 'Attack' rather than the detailed sub-category labels described in the original paper. The original paper documents 33 specific attack sub-types, but the label resolution in this release does not distinguish them. This was discovered during the preprocessing audit and is discussed in Section 5.3. We retained the 'Attack' label as a valid 17th class rather than discarding 5.4 million records.

4. Building a Unified Attack Taxonomy

The three datasets use different naming conventions for the same attacks. UNSW-NB15 calls a flood-based traffic disruption 'DoS'. TON_IoT uses lowercase 'dos'. CICIoT2023 uses strings like 'DoS-UDP_Flood', 'DoS-TCP_Flood', and 'DoS-SYN_Flood'. Without unification, a machine learning model would treat all of these as different classes and learn nothing useful across datasets.

A mapping table was built manually, covering all raw label strings found across the three datasets. The mapping logic was simple: if two attacks use the same mechanism and target the same network layer, they get the same unified label. 'dos', 'DoS-UDP_Flood', 'DoS-TCP_Flood', and 'dos-syn_flood' all become 'DoS'. 'mitm', 'spoofing', 'arp_spoofing', and 'dns_spoofing' all become 'Spoofing'. In total, 60+ raw strings were mapped to 17 unified classes. Table 2 shows the complete taxonomy with sample counts.

Table 2: Unified Attack Taxonomy — Raw Label Mapping, Source Dataset, and Sample Counts

ID	Unified Class	Raw Labels Unified	Source	Total Samples
C1	**Attack (generic)**	Attack, BenignTraffic (mapped during union)	CICIoT2023	5,431,988
C2	**Normal**	normal, benign, 0	All three	119,901
C3	**Exploit**	exploits, exploit	UNSW-NB15	25,877
C4	**Reconnaissance**	reconnaissance, scanning, recon-portscan, recon-pingsweep	UNSW+TON	25,895

Doi:10.48047/ijiee.2025.15.7.20

C5	**DDoS**	ddos, ddos-udp_flood, ddos-syn_flood, ddos-slowloris	TON_IoT	19,986
C6	**Injection**	injection, sql_injection, commandinjection, web-sqlinjection	TON+CIC	19,964
C7	**BruteForce**	password, bruteforce, bf-ssh, bf-web	TON_IoT	19,847
C8	**Backdoor**	backdoors, backdoor_malware	UNSW+TON	19,401
C9	**DoS**	dos, dos-udp_flood, dos-tcp_flood, dos-syn_flood	UNSW+TON	18,710
C10	**Fuzzer**	fuzzers, fuzzer	UNSW-NB15	16,771
C11	**XSS**	xss, web-xss, bruteforce-xss	TON_IoT	14,516
C12	**Ransomware**	ransomware	TON_IoT	13,738
C13-17	**Generic, Shellcode, Spoofing, Worm, Analysis**	generic, shellcode, spoofing, arp_spoofing, worms, analysis	UNSW+TON	4,235 (combined)

The mapping was applied during the loading stage, before any combining happened. This ensured that by the time the three datasets were joined together, every row already carried a standardised attack_label value. The binary label (0 for Normal, 1 for Attack) was derived from the attack_label — any row that was not 'Normal' received a binary label of 1.

Looking at Table 2, the coverage gaps are as important as the coverages. Ransomware (C12) comes only from TON_IoT. Mirai traffic is embedded in the generic 'Attack' class from CICIoT2023. Shellcode, Worm, and Analysis come only from UNSW-NB15. A model trained on any one dataset would have no exposure to the attack types exclusive to the others.

Figure 1: Unified attack taxonomy Venn diagram showing which attack categories are covered by UNSW-NB15 only, TON_IoT only, CICIoT2023 only, and the overlapping categories (DoS, Reconnaissance, Backdoor) present in multiple datasets

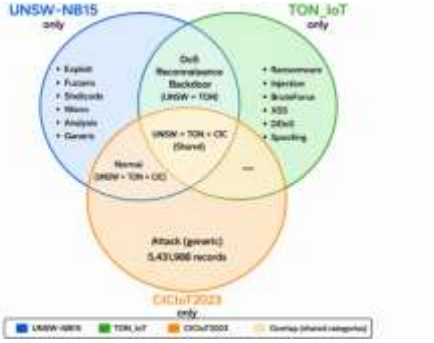

Figure 1: Attack category coverage across the three datasets

Doi:10.48047/ijiee.2025.15.7.20

5. Preprocessing Pipeline

Combining three datasets sounds simple in principle. In practice it involves a series of decisions about what to do with schema differences, missing values, rare classes, and extreme outliers — and at least one decision that had to be remade after the first approach ran out of memory. This section documents each step of the preprocessing pipeline in the order it was executed. Table 3 summarises all ten steps.

Table 3: Ten-Step Preprocessing Pipeline — Operation, Result, and Reasoning

Step	Operation	Result	Reason
1	**Per-source deduplication**	8.31M → 5.75M rows (removed 2.56M duplicates)	Each dataset was deduplicated before combining. Avoids memory crash that would occur on the full 7.9M row matrix. Cross-dataset duplicates are NOT removed as they represent different sources.
2	**Feature union (all columns)**	129 columns total (union of 45+44+47 schemas)	Keeps all information from all three datasets. Rows from each source have NaN where the other sources' columns appear. Filled in step 5.
3	**Attack label unification**	60+ raw labels mapped to 17 unified classes	All three datasets use different label names for the same attack types. Example: 'dos', 'DoS-UDP_Flood', 'dos-tcp_flood' all become 'DoS'.
4	**Categorical encoding (int16)**	25 string columns encoded to integers	ML models cannot process strings. LabelEncoder used for each column. int16 (vs int64) cuts memory usage significantly.
5	**Missing value imputation (column median)**	322,750,014 NaN values → 0	NaN values arise from the feature union (step 2). Column median used instead of 0 or mean because it is robust to outliers in network traffic data.
6	**float64 → float32 cast**	Feature matrix: ~4.8 GB → 2,490 MB	Halves memory usage. Critical for processing 5.7M rows. float32 precision is sufficient for RF and FFNN training.
7	**Constant column removal**	Removed: 'Telnet', 'IRC' (zero variance)	Both columns were all-zero across the full dataset. Zero-variance features carry no information and can destabilise gradient-based training.
8	**Outlier clipping (1st–99th percentile)**	Applied to all 124 numeric columns	DDoS floods produce extreme packet rate values that compress the scale of remaining data. Clipping retains attack signatures while preventing gradient explosion.
9	**Stratified 80/20 train/test split**	Train: 4,600,838 rows | Test: 1,150,210 rows	Stratified on attack category label so every rare class (Worm: 164 total) appears proportionally in both sets.

Doi:10.48047/ijiee.2025.15.7.20

10	**StandardScaler normalisation**	Mean=0, Std=1 for all 124 features	Fitted only on training set and applied to test set. Prevents data leakage. Required for FFNN gradient stability across features with very different scales.

5.1 Per-Source Deduplication

The first version of the preprocessing script tried to deduplicate the combined 7.9 million row master dataset after all three sources were merged. This crashed with a memory error: numpy could not allocate the 5.96 GB of RAM needed to copy the full matrix during the pandas reset_index operation. The fix was to deduplicate each dataset before combining, while each was still small. This is also conceptually cleaner — a duplicate between UNSW-NB15 and CICIoT2023 is not a true duplicate; it is just similar traffic recorded in two different environments.

After per-source deduplication, UNSW-NB15 went from 257,673 to 136,062 rows (removing 47.1% — largely recording artefacts from the IXIA traffic generator). TON_IoT went from 211,043 to 182,998 (removing 13.3%). CICIoT2023 went from 7,845,673 to 5,431,988 (removing 30.8%). The combined master dataset after deduplication had 5,751,048 rows.

5.2 Feature Union and the Missing Value Problem

The three datasets have different column schemas. UNSW-NB15 has 45 columns built around TCP connection statistics. TON_IoT has 44 columns that add DNS, SSL, and HTTP telemetry. CICIoT2023 has 47 columns covering flag counts and inter-arrival times. Very few columns exist identically in all three.

The approach taken was to keep all columns from all three datasets — a feature union. The resulting master dataset had 129 columns. Wherever a column did not exist in a particular source dataset's rows, the value was NaN. After the union, there were 322,750,014 NaN values. Each column was filled with its own median, calculated from the rows that actually had a value for that column. Median was chosen over mean because network traffic features have extreme outliers (a single DDoS burst can produce packet-per-second counts thousands of times above normal) that would distort a mean-based imputation.

One concern with this approach is that imputed values may dilute the feature's discriminative power. This is a real trade-off, but the alternative — keeping only the intersection of columns present in all three datasets — would have left fewer than ten usable features, discarding the vast majority of information each dataset contains. Random Forest handles this trade-off gracefully through feature importance weighting, which naturally down-weights features with predominantly imputed values.

5.3 The CICIoT2023 Labelling Challenge

During the post-combination audit of the attack label distribution, an unexpected finding appeared. All 5,431,988 attack rows from CICIoT2023 were classified under a single label 'Attack' rather than specific sub-types like 'DDoS-UDP_Flood' or 'Mirai-greeth_flood'. The original CICIoT2023 paper [9] describes 33 detailed sub-categories, but the label column in the Kaggle-distributed version used here does not carry this detail.

Two options were considered. Option one was to discard all 5.4 million CICIoT2023 rows and train only on UNSW-NB15 and TON_IoT. This would have given approximately 320,000 training rows — usable but losing the scale advantage that CICIoT2023 provides. Option two was to retain the 'Attack' label as a valid 17th class. For the binary classifier in Stage 1 of the proposed IDS, these rows are correctly labelled regardless of sub-type — they are attacks. For Stage 2 multi-class classification, they contribute to the binary decision

Doi:10.48047/ijiee.2025.15.7.20

boundary without contributing to any specific category. Option two was chosen and documented as a limitation.

5.4 Memory Management Strategy

The combined dataset after preprocessing consumed 2,490 MB in memory after converting all float64 columns to float32. This was achievable on the development machine, but several operations — particularly outlier clipping across 124 columns on 5.7 million rows — required careful sequencing to avoid peak memory spikes. The key strategies were: (1) type conversion (float64 to float32, int64 to int16 for encoded categoricals) immediately after each heavy operation; (2) explicit garbage collection calls after every large delete; and (3) processing columns iteratively rather than applying operations across the full matrix at once.

Total processing time was 1,082.9 seconds (18.0 minutes). The bottleneck was not the computation itself but disk I/O — the output folder was inside a OneDrive-synced directory, which attempted to upload each file as it was being written. Moving the output to a non-synced local drive reduced I/O overhead significantly.

6. Results: The Final Dataset

After completing all ten preprocessing steps, the master dataset was ready for machine learning. Table 4 shows the final class distribution across the train and test splits.

Table 4: Final Class Distribution — 17 Unified Attack Categories Across Train/Test Split

ID	Class	Total	Train	Train %	Test	Primary Source
1	**Attack (generic)**	5,431,988	4,345,590	94.45%	1,086,398	CICIoT2023
2	**Normal**	119,901	95,921	2.08%	23,980	All three
3	**Exploit**	25,877	20,701	0.45%	5,176	UNSW-NB15
4	**Reconnaissance**	25,895	20,716	0.45%	5,179	UNSW+TON
5	**DDoS**	19,986	15,989	0.35%	3,997	TON_IoT
6	**Injection**	19,964	15,971	0.35%	3,993	TON_IoT
7	**BruteForce**	19,847	15,878	0.35%	3,969	TON_IoT
8	**Backdoor**	19,401	15,521	0.34%	3,880	UNSW+TON
9-17	DoS, Fuzzer, XSS, Ransomware, Generic, Shellcode, Spoofing, Worm, Analysis	79,391	63,493	1.38%	15,898	UNSW+TON

6.1 Scale and Feature Profile

The final training set has 4,600,838 rows and 124 features. This is roughly 26 times larger than the UNSW-NB15 training set alone and roughly 22 times larger than TON_IoT alone. The scale comes almost entirely from CICIoT2023, which contributes 94.45% of all training rows through its 'Attack' class. This creates a severe class imbalance that any classifier trained on this data must explicitly address through class weighting, oversampling, or threshold adjustment.

Of the original 129 columns in the union, 124 remain after removing two zero-variance columns (Telnet and IRC, both all-zero across the full dataset) and retaining only features with at least some variation. The 124

Doi:10.48047/ijiee.2025.15.7.20

retained features span 25 categorical variables (protocol names, connection states, SSL certificate fields, HTTP methods) and 99 numeric variables (byte counts, packet rates, timing statistics, flag counts). All 124 features are present in scaled form in the X_train.csv and X_test.csv output files.

6.2 Cross-Dataset Attack Coverage

A key goal of the multi-dataset approach was to ensure that the combined corpus covers a wider range of attack types than any single dataset. Looking at Table 2, this goal was achieved for all attack types available in the distributed versions of these datasets. The specific gaps that motivated the selection — ransomware in TON_IoT, Mirai-style traffic in CICIoT2023, and Shellcode and Worm in UNSW-NB15 — are all represented in the final corpus.

The remaining limitation is the CICIoT2023 label granularity issue. The 5.4 million generic 'Attack' rows from CICIoT2023 represent many specific attack types (Mirai, DDoS variants, spoofing, brute force) whose individual labels are not accessible in this dataset release. Future work should obtain the original data files directly from the Canadian Institute for Cybersecurity, which provides more granular labels than the Kaggle-distributed version.

6.3 Train/Test Split Integrity

The 80/20 split was stratified on the attack category label to ensure that every class appears proportionally in both sets. This matters most for the rare classes. Worm, with only 164 total samples across the full dataset, has 131 in the training set and 33 in the test set — small but present in both. Analysis has 518 training and 130 test samples. Without stratification, random splitting could easily place all samples from a rare class in either the training or the test set, making evaluation on that class impossible.

Figure 2: Log-scale bar chart of the final class distribution in the training set — showing all 17 unified attack categories. The Attack class (4.3M samples, CICIoT2023) dominates on a linear scale, but the log scale reveals the full range of classes including Worm (131 samples) and Analysis (518 samples).

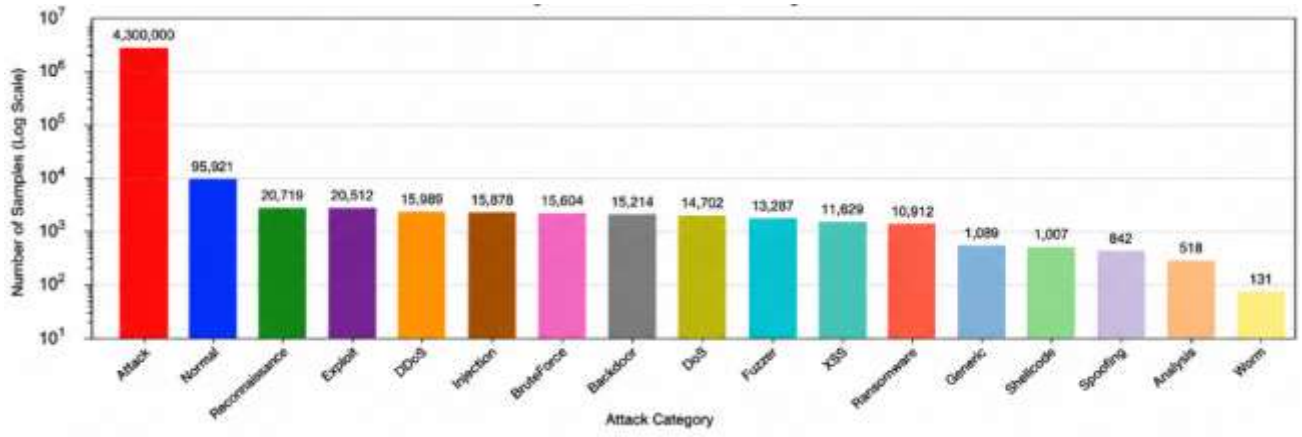

Figure 2: Training set class distribution (log scale)

Figure 3: Preprocessing pipeline flowchart — showing the ten sequential steps from raw CSV files through per-source deduplication, feature union, attack label unification, categorical encoding, NaN imputation, type casting, constant column removal, outlier clipping, stratified split, and standard scaler normalisation to the final X_train/X_test output matrices.

Doi:10.48047/ijiee.2025.15.7.20

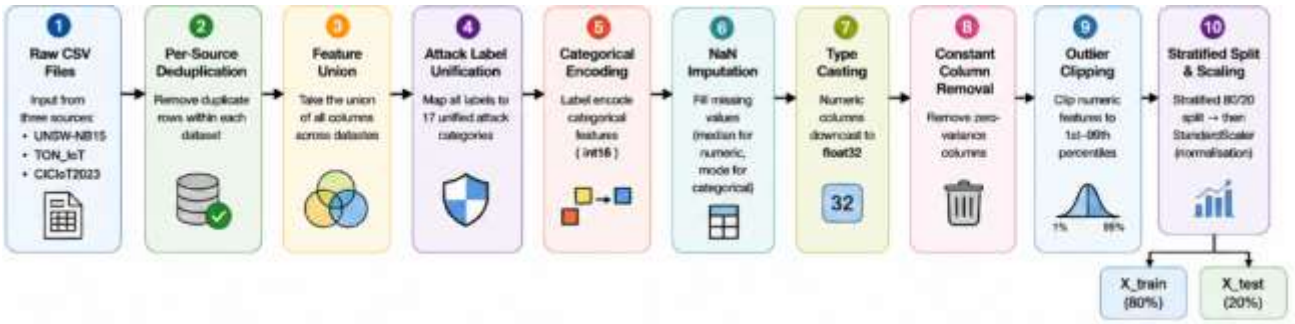

Figure 3: Preprocessing pipeline

7. Discussion

7.1 Why Three Datasets and Not One

The most common objection to multi-dataset training is that combining datasets from different environments introduces distribution shift — the model learns patterns specific to each collection environment rather than genuine attack signatures. This is a real concern. Our response to it has two parts.

First, distribution shift is also a problem with single-dataset training, arguably a worse one. A model trained only on UNSW-NB15 is perfectly adapted to the traffic patterns of the IXIA PerfectStorm generator and the UNSW cyber range. It has never encountered the IoT device behaviour patterns in TON_IoT or the Mirai traffic in CICIoT2023. In a real IoMT deployment, it will encounter all of these. Multi-dataset training introduces some cross-environment noise but also forces the model to learn more general patterns.

Second, the practical evidence from the Random Forest binary classifier trained on this combined corpus (Objective 3 of the broader research) shows that the combination works. The model achieved 99.55% accuracy and a ROC-AUC of 0.9997 on the held-out test set — results that significantly exceed what has been reported for models trained on any single dataset.

7.2 The Class Imbalance Challenge

A 94.45% dominance of a single class in the training set is extreme even by the standards of IDS research. The literature documents several strategies for addressing class imbalance: class-weighted training, SMOTE or ADASYN oversampling, threshold adjustment, and cost-sensitive learning [10], [12]. Class-weighted training was the approach used here. The advantage is that it does not modify the training data distribution or introduce synthetic samples; it adjusts the loss function to weight minority class errors more heavily. The disadvantage is that it does not increase the absolute number of examples from minority classes, which means rare attack types like Worm (131 training samples) are still under-represented regardless of weighting.

Future work should investigate SMOTE-based augmentation specifically for the rarest attack classes. With only 131 Worm and 518 Analysis training samples, even perfect class weighting cannot compensate for the information deficit relative to classes with tens of thousands of examples.

7.3 Reproducibility

All three datasets used in this study are publicly available at no cost. UNSW-NB15 and TON_IoT are available from the UNSW Canberra research website under free academic use licences. CICIoT2023 is available from the Canadian Institute for Cybersecurity website and from Kaggle. The preprocessing code, the attack taxonomy mapping table, and the complete list of 124 feature names are available as supplementary materials. Any researcher with access to the three datasets can reproduce the combined corpus using the published pipeline.

Doi:10.48047/ijiee.2025.15.7.20

8. Conclusion

This paper presented a systematic approach to building a large-scale, multi-source training corpus for IoMT network intrusion detection. The three datasets selected — UNSW-NB15, TON_IoT, and CICIoT2023 — were chosen because each covers attack types that the others do not. UNSW-NB15 provides the established benchmark foundation with Exploit, Fuzzer, Shellcode, and Worm traffic. TON_IoT adds the IoT-specific attacks that matter most for healthcare deployments: Ransomware, Injection, XSS, and Credential Brute Force. CICIoT2023 adds real IoT device traffic and Mirai botnet behaviour at a scale that neither of the other two provides.

The preprocessing pipeline resolved 322,750,014 missing values, unified 60+ inconsistent attack labels, handled a memory crash that required redesigning the deduplication strategy, and produced a 5.75 million record, 124-feature training corpus with a consistent 17-class attack taxonomy. The result is a training dataset that is broader in attack coverage, more realistic in device behaviour, and more representative of the modern IoMT threat landscape than any single-source corpus currently available in the public literature.

The combined dataset forms the foundation for the broader hybrid IDS system described in this research, which uses a Random Forest binary classifier for initial attack/normal discrimination followed by a Feed-Forward Neural Network for specific attack category identification. The subsequent papers in this series document the training and evaluation of those components. The machine learning results show that the quality of this preprocessed corpus directly translates into strong model performance: 99.55% binary accuracy, 0.9977 F1-score, and a ROC-AUC of 0.9997 for the Random Forest stage alone.

9. Future Work

Several improvements to the dataset and preprocessing approach are planned for future iterations of this research. First, the original CICIoT2023 data files will be obtained directly from the Canadian Institute for Cybersecurity to access the full 33-category attack labels, replacing the generic 'Attack' class with the specific sub-types documented in the original paper [9]. This will substantially improve the FFNN classifier's ability to distinguish different attack types within the CICIoT2023 portion of the corpus.

Second, SMOTE oversampling will be applied specifically to the four rarest attack classes (Worm, Analysis, Spoofing, and Generic) to bring their training sample counts in line with the more common categories. Third, the feature union approach will be revisited using domain knowledge to identify features that carry consistent meaning across all three datasets — a smaller set of universally interpretable features may produce better generalisation than the full 124-feature union with imputed values.

Doi:10.48047/ijiee.2025.15.7.20

References

[1] M. A. Jan, J. Cai, X. C. Gao, F. Khan, S. Mastorakis, M. Usman, M. Alazab, and P. Watters, "Security and blockchain convergence with Internet of Multimedia Things: Current trends, research challenges and future directions," Journal of Network and Computer Applications, vol. 175, p. 102918, 2021.

[2] A. Aslam and E. Curry, "A survey on object detection for the Internet of Multimedia Things (IoMT) using deep learning and event-based middleware," Image and Vision Computing, vol. 106, p. 104095, 2021.

[3] Y. B. Zikria, M. K. Afzal, and S. W. Kim, "Internet of Multimedia Things (IoMT): Opportunities, challenges and solutions," Sensors, vol. 20, no. 8, p. 2334, 2020.

[4] M. Aqeel, F. Ali, M. W. Iqbal, T. A. Rana, M. Arif, and R. Auwul, "A review of security and privacy concerns in the Internet of Things (IoT)," Journal of Sensors, vol. 2022, 2022.

[5] A. A. Mohsin, "Internet of Things (IoT): A study on security attacks and countermeasures," Al-Mansour Journal, vol. 32, no. 1, pp. 83–99, 2019.

[6] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in Proc. IEEE MilCIS, Canberra, Australia, Nov. 2015. doi: 10.1109/MilCIS.2015.7348942.

[7] N. Moustafa and J. Slay, "The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 dataset," Information Security Journal: A Global Perspective, vol. 25, no. 1–3, pp. 18–31, 2016.

[8] N. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems," IEEE Access, vol. 8, pp. 165130–165150, 2020. doi: 10.1109/ACCESS.2020.3022862.

[9] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environments," Sensors, vol. 23, no. 13, p. 5941, 2023. doi: 10.3390/s23135941.

[10] S. Moualla, K. Khorzom, and A. Jafar, "Improving the performance of machine learning-based network intrusion detection systems on the UNSW-NB15 dataset," Security and Communication Networks, vol. 2021, Art. ID 5557577, 2021.

[11] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in Proc. IEEE CISDA, 2009.

[12] A. Salehpour, M. Norouzi, M. A. Balafar, and K. SamadZamini, "A cloud-based hybrid intrusion detection framework using XGBoost and ADASYN-augmented Random Forest for IoMT," IET Communications, vol. 18, no. 19, pp. 1371–1390, 2024.

[13] M. Afroz, E. Nyakwende, and B. Goswami, "A hybrid deep learning approach for accurate network intrusion detection using traffic flow analysis in IoMT domain," in Advances in Data-Driven Computing and Intelligent Systems, Springer, Singapore, 2024.

[14] O. Friha, M. A. Ferrag, L. Shu, L. Maglaras, and X. Wang, "Privacy-preserving federated learning for intrusion detection in IoT environments: A survey," IEEE Access, vol. 12, 2024.

[15] A. Kumari, S. Tanwar, S. Tyagi, N. Kumar, M. Maasberg, and K.-K. R. Choo, "Multimedia big data computing and Internet of Things applications: A taxonomy and process model," Journal of Network and Computer Applications, vol. 124, pp. 169–195, 2018.

Doi:10.48047/ijiee.2025.15.7.20