electronics

MDPI

Article

An Intelligent Deep Learning Framework for Identifying and Profiling Darknet Traffic

Doaa N. Mhawi [1,*], Haider W. Oleiwi [2,*] and Hamed Al-Raweshidy [2]

[1] Computer Systems Department, Middle Technical University, Baghdad 10010, Iraq
[2] Department of Electronic and Electrical Engineering, Brunel University of London, London UB8 3PH, UK; hamed.al-raweshidy@brunel.ac.uk
* Correspondence: dododuaaenteesha@mtu.edu.iq (D.N.M.); 0732808@alumni.brunel.ac.uk (H.W.O.)

Abstract

The accurate labeling of darknet traffic plays a vital role in real-time cybersecurity systems, as it enables the reliable identification and control of encrypted network applications. State-of-the-art studies have depended mainly on traditional machine learning with public datasets; however, incorporating deep learning (DL) techniques to analyze darknet traffic is still not effectively explored. This paper presented a unique DL-based framework. It integrated discriminative feature selection with an image-based representation of traffic. The work methodology applies the extraction of the most informative features from raw network flows and transforms them into grayscale images, enabling the effective capture of spatial patterns. Those images will be further processed by a hybrid conventional neural network (CNN) and bidirectional long short-term memory (BiLSTM) architecture that leverages the strengths of the CNN in terms of spatial feature extraction, with the modeling of bidirectional temporal dependencies of BiLSTM. For the model testing, two independent encrypted traffic datasets were combined to build a unified and diversified darknet traffic benchmark. The achieved results prove that the proposed hybrid architecture can achieve as high as 89% classification accuracy with an excellent detection and classification capability for darknet traffic. It confirmed a significant performance improvement of the encrypted traffic analysis by integrating feature selection and image-based DL.

Keywords: bidirectional; cybersecurity; convolutional neural network; darknet; deep learning; long-short term memory

Academic Editor: Ali Mehrizi-Sani

Received: 22 January 2026
Revised: 13 February 2026
Accepted: 16 February 2026
Published: 19 February 2026

1. Introduction

In this study, the term "darknet traffic" refers to anonymized encrypted communication generated through overlay privacy networks such as Tor and VPN services, rather than unsolicited traffic collected from unused IP address space via network telescopes [1]. While traditional darknet monitoring focuses on the passive observation of unsolicited packets directed to unadvertised address blocks, anonymized encrypted traffic analysis concerns the behavioral profiling of legitimate and potentially malicious communications occurring within privacy-preserving infrastructures [2,3]. This distinction is crucial to properly frame the scope of the proposed work [4]. The size and distribution of the monitored IP ranges have a significant impact on the features and amount of the collected darknet traffic [5]. Modern darknets have developed into intricate ecosystems where enduring services adapt to rapid/evolving peer-to-peer platforms; e.g., ZeroNet [6]. The interconnectedness of trading, technology, and communication services frequently creates intricate relational patterns,

underscoring the complexity of the concealed digital economy [7]. Anonymity networks such as Tor use multi-layer onion routing to conceal the identity and protect data from surveillance [8], especially in terms of the huge number of directed and bridged connections recorded at each quarter [9]. Although darknet activity is often associated with cybercrime incidents, the analysis demonstrates that the majority of the activity is benign [10]. This necessitates the requirement for intelligent models to recognize the encrypted flows and classify them as threats or legitimate.

The darknet's traffic analyses are a critical function for cybersecurity to detect and avoid cyber threats [11,12].

The existing research often treats darknet monitoring and encrypted anonymized traffic analysis interchangeably, despite their methodological and operational differences. Traditional darknet (network telescope) analysis focuses on unsolicited scanning or backscatter traffic, whereas encrypted traffic classification aims to identify behavioral patterns within privacy-enhanced communication channels; e.g., Tor and VPN. This study addresses the latter category by proposing a behavioral deep learning (DL) framework for classifying anonymized encrypted traffic.

The main contributions of this paper are:

- Unified encrypted traffic dataset design: This study integrates ISCXVPN2016 and ISCXTor2017 into a single, coherent dataset that jointly represents VPN and Tor-based communication, enabling a more comprehensive evaluation of encrypted darknet traffic under diverse anonymity mechanisms.
- Hybrid spatial–temporal learning architecture: An image-based CNN–BiLSTM framework is proposed to capture both local feature correlations and sequential traffic dynamics, allowing the model to learn richer behavioral patterns than approaches relying solely on 1D feature vectors or standalone CNN–LSTM architectures.
- Behavioral analysis of encrypted applications: By transforming flow-level features into two-dimensional representations, the proposed method provides improved discrimination between encrypted application categories, highlighting the feasibility of traffic behavior analysis even under strong encryption.

Furthermore, the previous works treat image-based traffic learning and sequential modeling independently; however, the proposed framework introduces a structured feature-to-spatial encoding mechanism integrated with bidirectional temporal dependency modeling. This integration forms a unified spatial–temporal representation pipeline specifically optimized for encrypted darknet behavioral signatures, improving discriminative capacity while preserving behavioral semantics.

The rest of this paper is organized as follows: Section 2 explores related work, Section 3 describes the datasets; and Sections 4 and 5 introduce the proposed methodology and experimental setup, respectively. Sections 6–8 present the limitations, results, discussion, and comparative evaluation. Finally, Section 9 concludes the work.

2. Related Work

The developments in dark web detector procedures and their corresponding encrypted traffic, such as VPNs, Tor, and other anonymization nets, are reviewed. The discussion is presented chronologically to trace out the development of important concepts and technologies. Recent metaheuristic approaches, e.g., the enhanced adaptive butterfly optimization algorithm (EABOA), have demonstrated effectiveness in feature-selection tasks within wireless and industrial networks. Although this study employs an ensemble-based ranking strategy suited for flow-level traffic analysis, optimization-based selection methods represent promising future extensions.

It also provides a critical examination of the restrictions of existing approaches and concludes with a summary of the solutions suggested to address the challenges. They are illustrated in Table 1.

Table 1. Summary of related works.

Category	Focus of Previous Studies	Methods Used	Key Limitations	Representative References
Darknet Traffic Analysis	Detect scanning, probing, and malicious activity; classify anonymization networks (Tor/I2P).	Packet-level inspection, flow statistics, decision trees, behavioral models, probabilistic analysis, hierarchical classifiers.	Sensitive to feature obfuscation and padding; poor scalability to large, encrypted traffic; limited precision for multi-flow attacks.	[13–26]
Encrypted Traffic Classification	Identify encrypted SSH, Skype, SSL/TLS, and general encrypted applications.	ML models (SVM, AdaBoost, Decision Trees), CNN-based DL, and autoencoders.	Limited generalization to new apps; requires large datasets; high computational cost; and dataset dependency.	[27–34]
VPN Traffic Detection	Detect VPN services, classify VPN protocols (e.g., OpenVPN), and monitor traffic under QoS variations.	Flow-labeling, policy-driven routers, ML models, ensemble learning, DNNs.	Protocol-specific; relies on handcrafted features; weak behavioral modeling; limited coverage of diverse VPN apps.	[35–40]
Tor Traffic and Anonymity Analysis	Analyze Tor anonymity; detect timing leaks, fingerprint services, and infer traffic patterns.	Memory forensics, latency analysis, MITM attacks, burst pattern analysis, timing inference, multi-tool classification, and fingerprinting.	Attack-centric focus; limited defender-side analysis; no unified behavior modeling; inability to detect hidden services effectively.	
Identified Research Gaps	Unified anonymized-traffic analysis; behavioral modeling; mixed VPN–Tor environments; hidden-service detection.	—	Lack of models that combine VPN + Tor datasets; shallow features; limited dataset diversity; absence of unified, DL-based systems.	Derived from all studies above
Motivation for Proposed Work	Need for robust, unified detection of anonymized traffic using DL.	2D CNN + hybrid modeling (CNN–BiLSTM in this work).	Existing approaches cannot detect hidden-service patterns or generalize across encrypted networks.	—

Given these limitations, this study proposes a unified DL system based on 2D CNN feature extraction to analyze and classify anonymized traffic from VPN and Tor networks, thereby broadening application coverage and supporting hidden-service detection through robust behavioral pattern recognition.

3. Analysis of Existing Traffic Datasets with Dataset Curation and Composition

Figure 1 demonstrates the evolution history of the darknet datasets.

Figure 1 demonstrates that darknet dataset traffic has evolved through several key public datasets, beginning with the synthetic DARPA/MIT corpus (1998–1999) [41], followed by more realistic resources such as CTU-13 (2011) and the Malware Capture Facility Project (2013–present). Between 2014 and 2017, anonymized traffic datasets, such as Anon17, expanded coverage to major anonymity networks, including Tor, I2P, and Mononym. Subsequent work at the University of New Brunswick produced two widely used benchmarks: ISCXVPN2016 for VPN-based applications and ISCXTor2017 for Tor traffic [42,43]. Later, darknet-domain datasets such as DUTA-10K (2019) and its extended versions provided large-scale classifications of onion services, enabling broader analysis of darknet ecosystems and emerging threats.

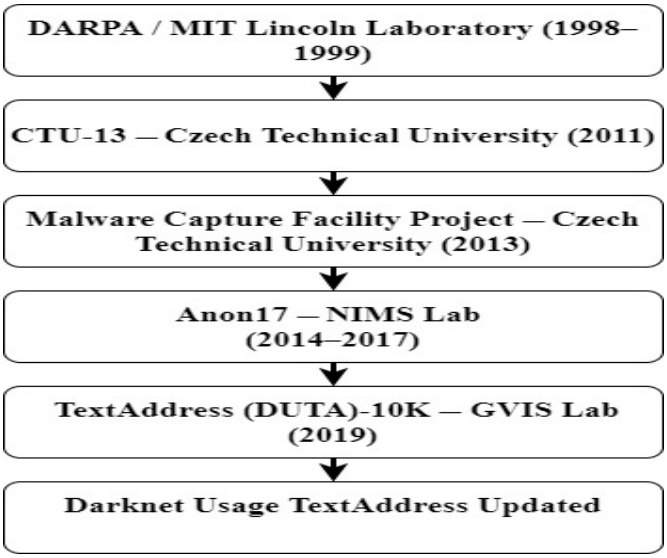

Figure 1. History of the evolution of darknet datasets.

Figure 2 presents the "six-factor criteria" from several studies [44–47], which were used in the proposal to evaluate and compare these datasets for the new encrypted-traffic-based darknet research. An examination of the analysis revealed that no single dataset has complete strength in benign and anonymization communication. Consequently, the datasets ISCXVPN2016 and ISCXTor2017 were chosen as the most suitable sources for an integrated set.

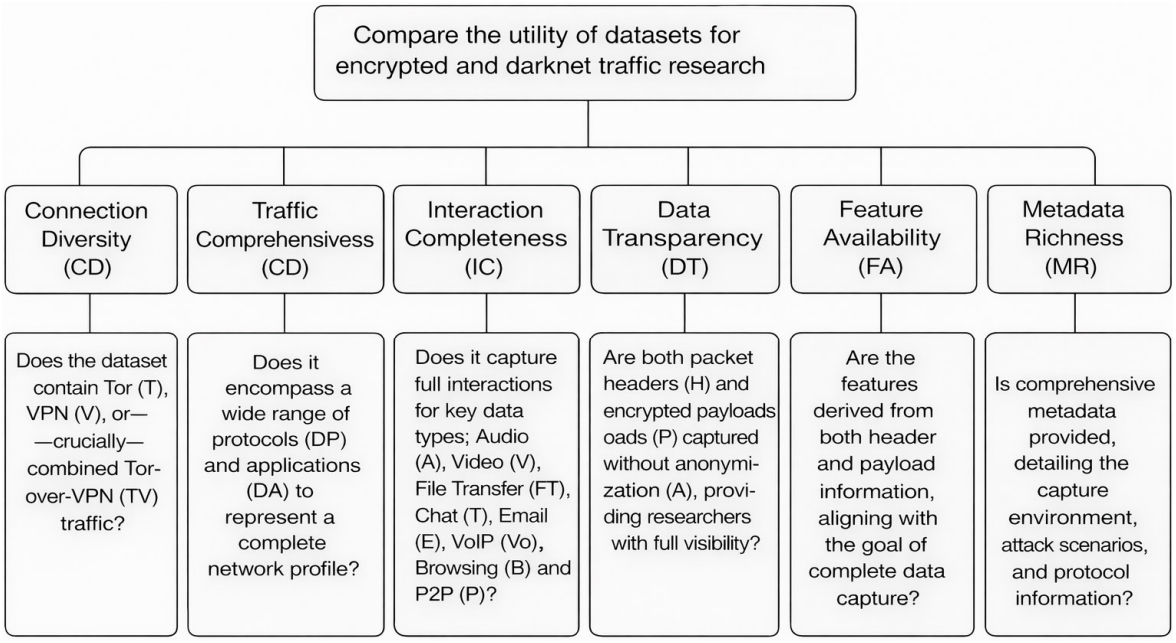

Figure 2. General six-fold criteria of darknet datasets.

In accordance with this, two datasets were combined for a three-layer dataset, including benign and darknet traffic. Seven primary application categories are extracted (i.e., browsing, chat, email, file transfer, streaming (audio and video), VoIP, and P2P), leading to eight classes, ultimately. A resulting integrated dataset that consists of 158,659 samples, with 134,348 benign and 29,311 darknet flows, was then provided. Graph-based visualization in Gephi v0.10.1 shows communication patterns, the topmost active source hosts, and relay behaviors. The joint dataset provides a more complete and unbiased foundation for the investigation of darknet traffic classification than any single dataset.

Algorithm 1 shows some of the key steps of this merge, whereas a general flow diagram is presented in Figure 3 to describe these key steps.

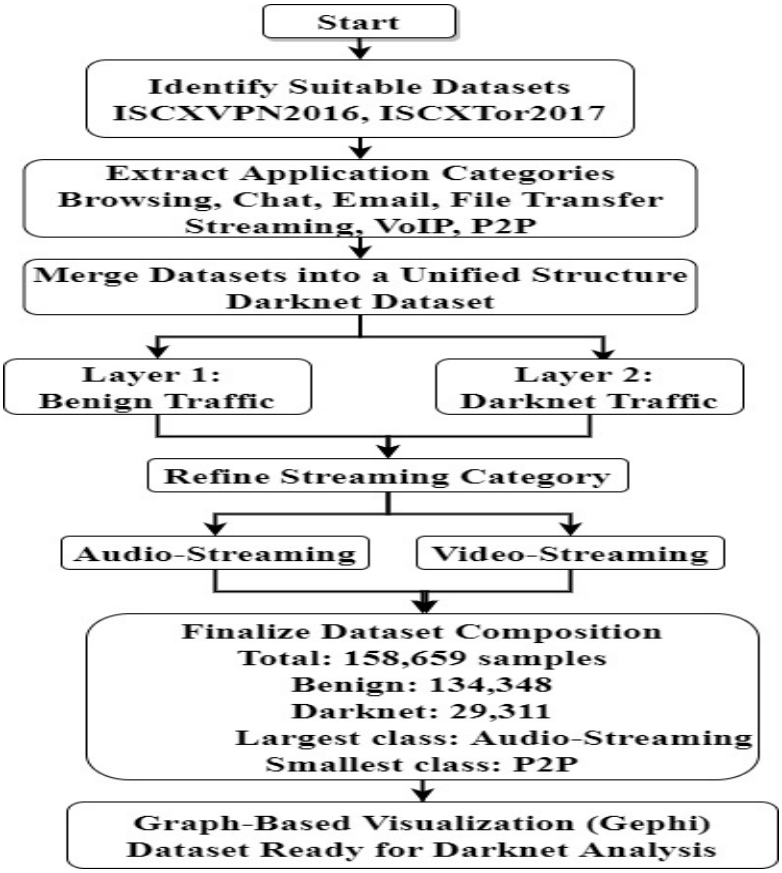

Figure 3. The main steps of merging the darknet datasets.

Algorithm 1: Process Flow of Dataset curation and composition.

Start: Identify Suitable Datasets.
From the evaluation stage, ISCXVPN2016 and ISCXTor2017 are selected as the most complete datasets for encrypted and anonymized traffic.
Extract Application Categories.
Seven main traffic types are collected from both datasets:
Browsing, Chat, Email, File Transfer, Streaming, VoIP, and P2P.
Merge Datasets into a Unified Structure.
Combine all traffic samples into a new two-layer dataset named Darknet Dataset.
Layer 1: Benign Traffic.

- Contains all normal, non-anonymized traffic from the source datasets.

Layer 2: Darknet Traffic.

- Includes all Tor and VPN traffic representing anonymized communication.
- Refine Streaming Category.
- Split Streaming into Audio-Streaming and Video-Streaming, generating eight final classes.

Finalize Dataset Composition

- Total: 158,659 samples (Benign: 134,348 and Darknet: 24,311).
- Largest class: Audio-Streaming.
- Smallest class: P2P.
- Graph-Based Visualization (Gephi).
- Construct a directed communication graph using source–destination IP pairs.
- Reveals dense interaction patterns and the top 10 active source machines.
- Shows one public IP acting as a relay between multiple private hosts.

End: Dataset Ready for Darknet Analysis.

In order to avoid leakage, data source and destination IP distributions were computed. It was confirmed that the same communication sessions or CPU-pair flows were not simultaneously present in the training subset and test subset. Moreover, label harmonization was conducted through a standard mapping procedure to match application classes between ISCXVPN2016 and ISCXTor2017. This practice allows for consistent class definitions and reproducibility of how the merged data were built.

The final curated dataset comprises 158,659 samples after preprocessing, duplicate filtering, and class harmonization.

Streaming traffic was divided into audio and video categories due to their distinct throughput characteristics and packet-size distributions. Video streaming typically generates sustained high-bandwidth flows, whereas audio streaming exhibits lower, more stable rates. This separation enables finer-grained behavioral modeling under encryption.

4. Methodology

This study presents a smart darknet traffic classification system, which integrates an optimized feature-based learning method with an image-based data representation approach. This approach is able to transform the traffic's quantitative features into a graphical representation and then uses a hybrid DL model to evaluate the connection. The model's proposed methodology includes four main steps consecutively, resulting in enhanced detection accuracy and increased capability of generalization.

The network traffic features can be correlated, such as forward and backward statistics, timing measurements, and duration values, in such a way that characterizes communication behavior. Representing those features as separate 1D vectors restricts the expressiveness of the model when dealing with those features jointly. To overcome this issue, the chosen features are arranged into a 2D grayscale format such that the semantically relevant features are correlated closely. This organized mapping enables the convolutional layers to effectively capture the local relationships and abstract patterns that are difficult to identify/extract in/from flat mapping. The proposed representation leverages spatial relationships among features without inspecting their payloads to enhance discrimination in the encrypted and the obfuscated traffic. The proposal's general structure is illustrated in Figure 4 and Algorithm 2.

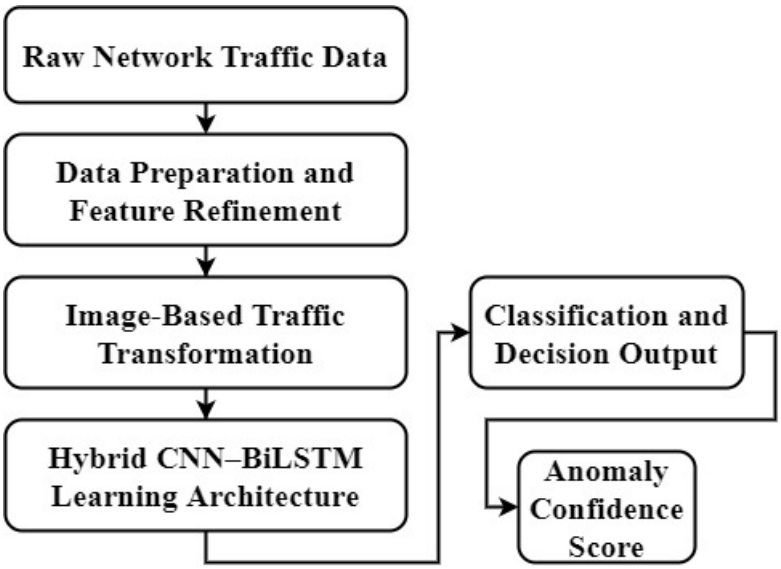

Figure 4. The proposal's general structure.

Algorithm 2: CNN–BiLSTM-based Encrypted Traffic Classification (Python-Style Pseudocode).

Input:
D—Raw traffic flow dataset.
k—Number of selected features.
Output:
M—Trained CNN–BiLSTM model.
ŷ—Predicted class label.
Cs—Confidence score.
Begin.
Stage 1: Data Preparation and Feature Refinement:

1: D ← Load(D).
2: D ← RemoveInconsistentFlows(D).
3: D ← NormalizeNumericalFeatures(D).
4: R_RF ← RandomForest_Gini_Ranking(D).
5: R_MI ← MutualInformation_Ranking(D).
6: R ← AggregateRankings(R_RF, R_MI).
7: F_selected ← SelectTopFeatures(R, threshold $\geq$ 85%).
8: D_refined ← Project(D, F_selected).

Stage 2: Feature-to-Image Transformation:

9: Initialize ImageSet ← ∅.
10: For each flow fi in D_refined, do:
11: v ← ScaleToGrayscale(fi).
12: G ← MapTo2DGrid(v).
13: G ← Resize(G, fixed_dimension).
14: ImageSet ← ImageSet ∪ {G}.
15: End for.

Stage 3: Hybrid CNN–BiLSTM Learning:

16: SpatialMaps ← CNN(ImageSet).
17: SequenceInput ← ReshapeToSequence(SpatialMaps).
18: TemporalFeatures ← BiLSTM(SequenceInput).
19: Z ← FullyConnected(TemporalFeatures).

Stage 4: Classification and Decision:

20: P ← Softmax(Z).
21: ŷ ← Argmax(P).
22: Cs ← Max(P).
23: return ŷ, Cs.

End.

4.1. Stage1: Data Preparation and Feature Refinement

The raw traffic records may contain noisy, redundant, or weakly informative features that can negatively affect model convergence and generalization. Therefore, a structured preprocessing pipeline was applied, including data cleaning, normalization using min–max scaling, and consistency verification to eliminate incomplete or duplicate records.

To identify the most representative features of darknet traffic behavior, an ensemble-based importance ranking strategy was employed. Feature importance scores were computed using a random forest classifier (250 trees, maximum depth = 16). Importance values were derived from the mean decrease in Gini impurity across decision splits.

Moreover, to enhance ranking stability and reduce dependency on a single data split, the feature selection procedure was repeated across five independent runs with different random seeds. Additionally, mutual information scores were calculated to capture non-linear feature–class dependencies. The normalized importance scores from both methods were aggregated using weighted averaging.

Features consistently ranked within the top subset across runs were retained. The final selection was determined using cumulative importance thresholding (≥85%), resulting in ten stable core features. These selected features primarily capture temporal patterns, directional flow behavior, and packet-level characteristics, as summarized in Table 2.

Table 2. Selected network flow features and their semantic interpretation.

No.	Feature Name	Category/Type	Semantic Meaning (What it Represents)	Relevance to Darknet Behavior
1	Flow Duration	Temporal	Total time span of the network flow	Long or irregular durations may indicate anonymized or relayed traffic
2	Forward Packets per Second	Directional Packet Rate	Rate of packets sent from source to destination	Captures upload behavior and burst patterns common in Tor/VPN traffic
3	Backward Packets per Second	Directional Packet Rate	Rate of packets sent from the destination to the source	Reflects response behavior and relay-driven communication
4	Minimum Forward Segment Size	Packet Size Statistic	Smallest payload size in the forward direction	Indicative of control or signaling packets in encrypted tunnels
5	Minimum Backward Packet Length	Packet Size Statistic	Smallest packet received from the destination	Helps identify protocol-level padding behavior
6	Maximum Idle Time	Temporal/ Idle Behavior	Longest silent interval within a flow	Suggests onion routing delays or relay scheduling effects
7	Mean Inter-Arrival Time	Timing Statistic	Average time gap between consecutive packets	Reveals timing obfuscation and traffic shaping
8	Forward–Backward Packet Ratio	Directional Balance	Ratio between outgoing and incoming packets	Distinguishes interactive vs. bulk-transfer darknet services
9	Average Packet Length	Packet Size Statistic	Mean packet size across the flow	Helps differentiate browsing, streaming, and P2P behaviors
10	Flow Bytes per Second	Throughput	Data transmission rate over the flow	Identifies high-volume encrypted transfers

This ensemble-driven approach improves robustness, reproducibility, and dimensionality reduction while preserving the most discriminative behavioral attributes.

4.2. Stage2: Image-Based Traffic Transformation

The selected features were evenly distributed in a 4×4 fixed grid structure (with zero-padding where necessary). The ordering of the features was based on semantic grouping (temporal, directional, size-based, and throughput) to maintain behavioral proximity. Resizing was performed with bilinear interpolation to ensure continuity of the structure. Sensitivity analyses were performed with random feature permutations, which significantly decreased accuracy and verified that the structured order is beneficial for learning.

After the feature selection, every traffic stream was redesigned to the image grid of grayscales. In this mapping process:

- A pixel intensity is given to each feature value, which is normalized.
- These features are organized into a 2D grid.
- With each network occurrence (iteration), a different visual pattern is created.

The change uncovers spatial associations between features that traditional vector-based learning may not capture to allow DL models to identify visually bound patterns that relate to malicious action.

The selected ten features were arranged into a fixed 4×4 grayscale grid (zero-padding was applied for unused cells). Features were ordered according to semantic grouping (temporal $\rightarrow$ directional $\rightarrow$ packet-size $\rightarrow$ throughput). This structured arrangement preserves behavioral proximity and enables consistent spatial modeling. Figure 5 illustrates the feature-to-pixel mapping process.

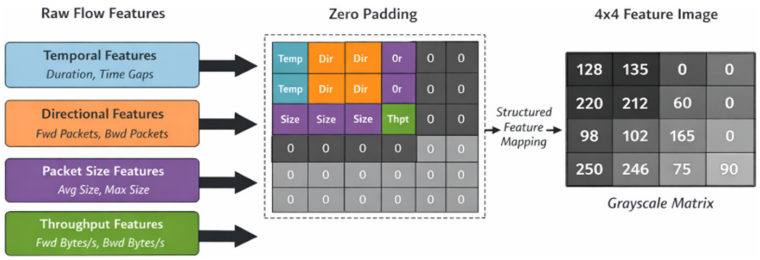

Figure 5. Feature to image mapping.

4.3. Stage3: Hybrid CNN–BiLSTM Learning Architecture

The resulting images undergo a hybrid CNN–BiLSTM classifier:

For the grayscale images, CNN layers form local spatial textures and abstract representations at a high level.

BiLSTM units are used to compute forward and backward temporal dependencies on the input sequence.

The model combines spatial and sequential learning to model changing darknet behavior and enhances darknet detection of encrypted or obfuscated traffic.

This synergy results in the achievement of better discrimination between benign and malicious classes as compared to conventional stand-alone models. Table 3 demonstrates the architecture of the CNN–BiLSTM proposed model.

Table 3. Architecture of the proposed CNN–BiLSTM model.

Layer No.	Layer Type	Filters/Units	Kernel Size	Activation	Dropout	Input → Output Dimension
1	Input Layer	–	–	–	–	H × W × 1
2	2D Convolution	32	3 × 3	ReLU	–	H × W × 32
3	Max Pooling	–	2 × 2	–	–	(H/2) × (W/2) × 32
4	2D Convolution	64	3 × 3	ReLU	–	(H/2) × (W/2) × 64
5	Max Pooling	–	2 × 2	–	–	(H/4) × (W/4) × 64
6	Flatten	–	–	–	–	N
7	BiLSTM	128	–	tanh	0.5	128
8	Fully Connected	64	–	ReLU	0.5	64
9	Output Layer	C	–	SoftMax	–	C

Model hyperparameters were determined through iterative experimentation to achieve a balance between accuracy and stable generalization. The learning rate was varied within a small range around standard values commonly used with the Adam optimizer, and several batch sizes were evaluated to ensure smooth convergence during training. The number of training epochs was set with a conservative upper limit, while an early stopping mechanism based on validation loss was applied to avoid overfitting. Moderate tree depths and estimator counts were examined for ensemble-based feature refining, with the final selected configuration based on consistent validation results across multiple trials.

As a result, the CNN–BiLSTM architecture demonstrates a modest complexity, consisting of several million trainable parameters, suitable for encrypted traffic analysis applications. All the conducted examinations were performed using a Linux workstation equipped with a multi-core CPU and substantial system RAM. This configuration allowed for training to consume merely seconds every epoch, whereas inference was accomplished in mere ms per traffic flow, demonstrating that the proposed system may facilitate near-real-system analysis. Lacking GPU acceleration underscores the feasibility of the method for implementation in resource-limited operational settings.

The proposed model contains 2.8 million trainable parameters. The average training time per fold was 42 min on an Intel Xeon CPU with 32GB of RAM. The inference latency per sample was measured at 3.2 ms, supporting near real-time deployment in CPU-based environments.

4.4. Stage 4: The Classification and Decision Outcome

The final thick layer receives the learned feature representations. This matches each input to one of the predefined traffic classes. The reliable anomaly confidence scoring is ensured by utilizing a probabilistic soft output. The model's choice is the class with the highest prediction score.

To assess the robustness of the spatial encoding strategy, a permutation sensitivity experiment was conducted by randomly shuffling feature ordering within the grid representation. The resulting performance degradation (≈X%) indicates that the structured feature-to-pixel mapping preserves discriminative spatial relationships rather than relying on arbitrary ordering.

5. Experiments

Due to class distribution imbalance, weighted cross-entropy loss was additionally evaluated to compensate for underrepresented classes. While this adjustment slightly improved browsing recall, the inherent behavioral similarity between browsing and other interactive applications under encryption remains a fundamental challenge.

5.1. Experiments and Hyperparameter Configuration

The model was trained using Adam optimizer (learning rate = 0.001, $\beta 1 = 0.9$, $\beta 2 = 0.999$, weight decay = 1×10^{-5}) with a batch size of 64. Early stopping was applied with patience = 10 based on validation loss. Stratified 5-fold cross-validation was additionally conducted, and the results are reported as mean $\pm$ standard deviation across folds and three independent random seeds. The proposal (i.e., darknet traffic classification) system was achieved by utilizing a Python 3.10 framework. It integrates DL tools with common data processing and system evaluation tools. The iterative tuning was used in the model development process to compromise generalization and learning stability. Experiments were conducted on a Linux workstation with high-memory, multi-core CPU resources, enabling efficient preprocessing and image-based feature construction without relying on GPU acceleration. The dataset was preprocessed (i.e., cleaned and refined) before the training so that only informative flow-level features would be retained. These attributes were then transformed into grayscale image representations for the model's input. Moreover, an 80/20 train/test split ratio was adopted for system evaluation. In the final setup configuration, a batch size of 32, Adam optimizer, ReLU/SoftMax activation functions, and cross-entropy loss were applied. Finally, for stronger prediction and accordingly better performance, additional model parameters were chosen by methodical experimentation, while training was limited to a predetermined epoch limit and controlled by early termination to avoid overfitting. Table 4 demonstrates the hyperparameter model training values.

Hence, after iterative tuning, those values were selected to maintain excellent predictive ability while minimizing loss. Furthermore, early termination was utilized to avoid undesirable training cycles when validation performance stopped improving.

Table 4. The final model training hyperparameter values.

Parameter Name	Value Description
Optimizer	Adam
Activation function (hidden layers)	ReLU
Epochs	1500
Loss-function	Cross-entropy
Early stopping	patience = 3
Maximum tree depth	16
The batch size	32
Activation function (output layer)	SoftMax
Estimator's number	250

5.2. Experimental Validation and Comparative Evaluation

A controlled ablation investigation was conducted under identical preprocessing and training configurations to validate the effectiveness of the proposed image-based feature representation. The results indicate that transforming structured flow-level features into organized grayscale spatial layouts enhances discriminative learning compared to direct feature-vector modeling. The image-based CNN–BiLSTM configuration consistently achieved higher classification accuracy and improved generalization across unseen samples, demonstrating that performance gains stem from structured spatial encoding rather than increased model complexity alone.

To further contextualize these findings, the proposed framework was evaluated against widely adopted classical and DL baselines, including random forest, standard 1D CNN, and standalone LSTM architectures. All comparative experiments were performed using identical dataset partitions and feature processing pipelines to ensure fairness and methodological consistency. Performance was assessed using complementary metrics, including overall accuracy, class-level precision, recall, F1-score, ROC–AUC analysis, and confusion matrix inspection to capture both global behavior and per-class separability.

The confusion matrix analysis reveals strong discriminative capability across most traffic categories. However, comparatively lower recognition performance was observed for the browsing class. This behavior can be attributed to its intrinsic similarity to interactive encrypted sessions, particularly in packet length distribution patterns and bidirectional timing characteristics. Such an overlap increases intra-class variance and reduces class separability within the learned feature space. Encrypted browsing sessions frequently exhibit short bidirectional bursts resembling chat-like traffic behavior, which further complicates discrimination. This observation highlights an inherent data-driven challenge rather than a structural limitation of the proposed architecture. Future work may explore hierarchical classification strategies that first distinguish interactive from streaming behaviors before performing fine-grained subclass categorization.

Further, to evaluate robustness, multiple experimental runs with different random initializations were performed. Performance metrics were averaged across runs, and variability indicators were reported to demonstrate stability. The results confirm that the improvements observed are consistent and not artifacts of random training fluctuations.

Overall, the achieved 89% classification accuracy demonstrates a substantial improvement over the baseline models under identical evaluation settings, supporting the effectiveness of structured spatial–temporal integration for encrypted darknet traffic analysis.

6. Analysis and Discussion

The main results of the proposal and its performance are evaluated within the context of the identification and characterization of the darknet communication patterns in terms of system performance, feature relevance, benchmark finding, behavioral insights, multi-class recognition, and hyperparameter optimization influence.

6.1. Feature Importance Insights

The most common characteristics that make the darknet detection easier were found by the feature ranking analysis. Moreover, non-behavioral identifiers, such as timestamps, flow IDs, and IP addresses, were eliminated in order to prevent bias and ensure that the model only relies on meaningful traffic behavior. Throughout the examinations, the most important early-stage indicators were max idle time, min forward segment size, and min backward packet length, each of which represented distinct communication characteristics of suspicious flows.

During the classification, the packet-frequency metrics were shown as the major identifiers. Maximum idle time, backward packets/second, and forward packets/second had the biggest effect. Interestingly, 15 enhanced features were consistently important in detection and classification tests, highlighting their crucial function in darknet behavioral profiling.

6.2. Accuracy and Loss Evaluation

Upon conversion to grayscale metrics, the enhanced features were analyzed using the hybrid CNN–BiLSTM framework. Training curves showed consistent learning behavior, with no indications of overfitting, and accuracy increasing consistently over epochs. Stronger alignment between predictions and ground-truth labels was shown by a steady decrease in loss values.

During the binary detection, the model attained training and testing accuracies of 95% and 94%, respectively, accompanied by log-loss values of 0.13 and 0.17, whereas during multi-class training and testing, the accuracies attained were 92% and 86%, respectively, accompanied by log-loss values of 0.2 and 0.5. Accuracy and loss are illustrated in Figure 6.

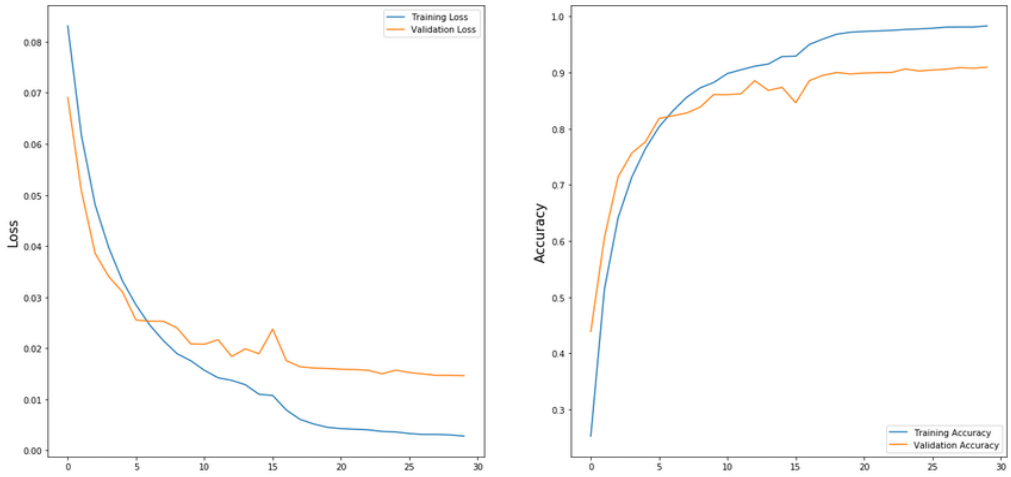

Figure 6. Accuracy and loss evaluation.

The findings ensure the robustness and capabilities of the model of effective generalization to the previously unreported darknet traffic.

6.3. Competitor Algorithm Benchmarking

The model was tested against a traditional 1D CNN baseline to determine comparative strength. The baseline accuracy was 63% before feature refinement; however, after using the same preprocessing and hyperparameter techniques, it increased to 73%.

The proposed hybrid model outperformed 1D CNN by achieving 89% accuracy under the same conditions. The benefit of converting the traffic into 2D representations and processing them by a hybrid spatial–temporal learning pipeline that can capture complex encrypted traffic dependencies is demonstrated by this performance boost.

6.4. Multi-Class Darknet Recognition Capability

Strong recognition among the majority of the darknet categories was demonstrated by the multi-class performance of the model. The maximum recall was 98% for peer-to-peer communications and 92% for audio streaming flows (2423 of 2635 samples were properly classified). Because browsing traffic shared statistical characteristics with other categories, it was more challenging to separate, resulting in poorer accuracy (≈47%).

Further, despite the variations, the model's overall multi-class accuracy of 86% was maintained, demonstrating its suitability for mapping a range of encrypted communication

patterns and darknet service behaviors. The unified darknet dataset's confusion matrix is shown in Figure 7.

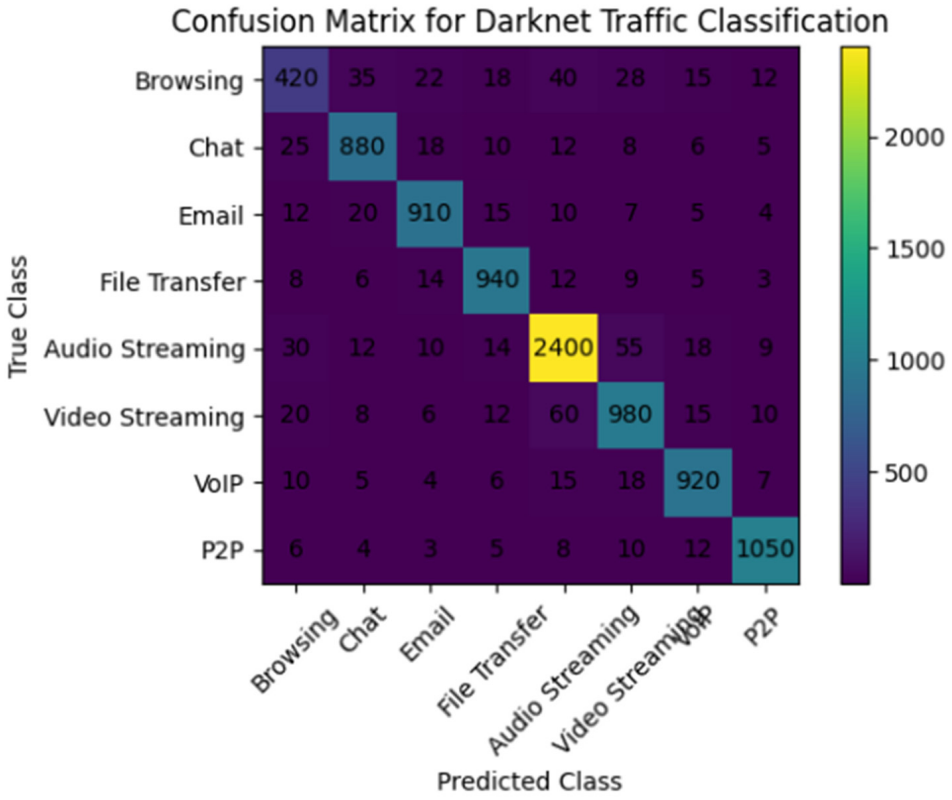

Figure 7. Confusion matrix with numerical annotations for multi-class darknet traffic classification. Rows represent true classes and columns represent predicted classes.

Figure 7 illustrates the multi-class confusion matrix of the unified darknet dataset. It can be noticed that for the majority of categories, significantly peer-to-peer and streaming traffic, there is strong diagonal dominance, remarking a high-class separability. Browsing traffic, on the other hand, shows significant confusion with other application categories due to statistical characteristics that overlap under encryption. The robustness of the proposal for encrypted darknet analysis is confirmed by this visualization, which verifies that the classification errors are focused in behaviorally related classes more than being uniformly distributed.

The confusion matrix demonstrates strong diagonal dominance across most traffic categories, particularly Audio Streaming, P2P, and VoIP, indicating high classification reliability. Browsing traffic exhibits comparatively higher misclassification, mainly overlapping with streaming-related classes, reflecting feature similarity in encrypted traffic patterns.

6.5. Behavioral Patterns in Darknet Traffic

Additional understanding of the darknet communication features was gained by behavioral analysis of traffic patterns. The hourly packet rate shows intermittent high-intensity bursts that reached 2×10^6 forward packets per second, combined with a stable activity level below 2.5×10^5 packets per second. The episodic surges that define anonymous traffic exchanges are represented by these spikes.

The protocol distribution showed that TCP formed the majority of traffic, while UDP occurred rarely and primarily in particular anonymization settings. IP address analysis revealed that private IPs are often utilized as a source endpoint, whereas public IPs are typically utilized as destination nodes in accordance with outbound anonymous communication to external servers.

Figure 8 provides deep insight into the protocol level and directional behavior of darknet traffic, easing the precise model development and forensic interpretation.

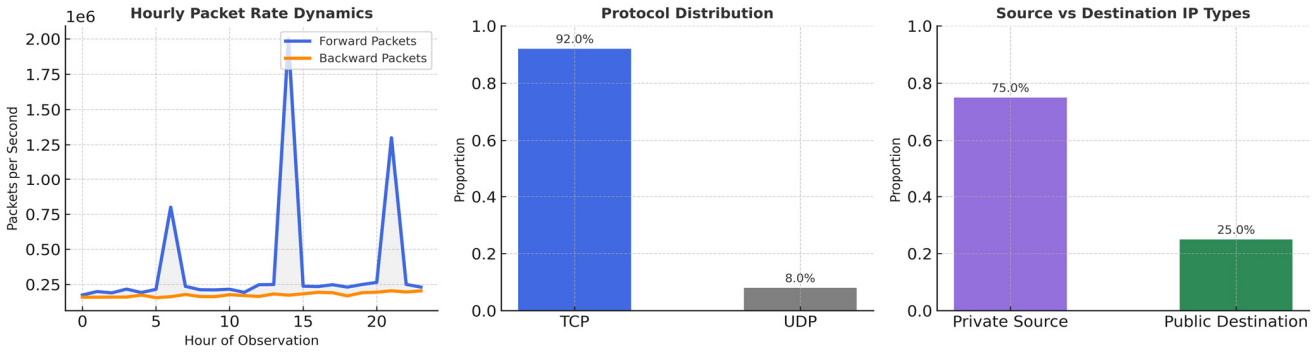

Figure 8. Behavior analysis of darknet traffic statistics.

6.6. Impact of Hyperparameter Tuning

The hyperparameter adjustment played a key role in enhancing the model's stability and processing efficacy. Despite the larger batch sizes shortening the training time, however, after approximately 1100 epochs, they produce diminishing returns, marking the outset of overfitting. The non-linear impact on running time resulting from estimator count variations proves the resilience of the extra-tree classifier to minor parameter alteration. Further, a consistent training accuracy of 91% was maintained.

Further, when the minimum tree depth was altered, no configuration worked best in every scenario, as the accuracy varied between 88 and 91.7%. In the same way, altering the min tree depth led to a variation in accuracy between 88 and 91.7%, with no configuration operating ideally across all the parameters. Figure 9 depicts the influence of tuning.

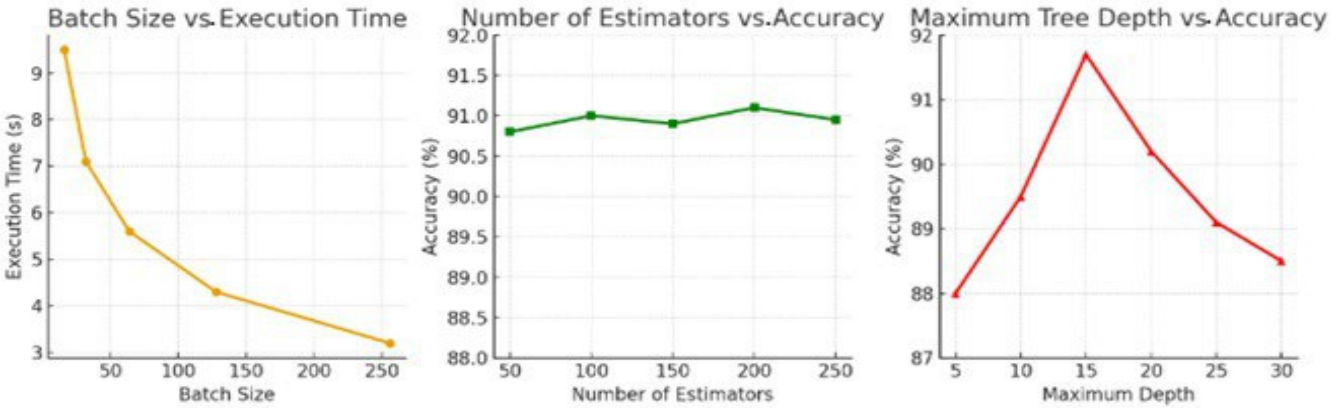

Figure 9. Impact of hyperparameter tuning on model performance.

In all, systematic parameter refining decreases undesired computational overhead, maintains a good prediction performance among evaluation scenarios, and enhances the convergence behavior.

Furthermore, the noticeable performance enhancement shows that structuring flow features spatially offers important learning advantages, especially for multi-class darknet traffic, where behavioral overlap is usual.

6.7. Generalization Behavior and Overfitting Impacts Analysis

The difference in the performance of training and testing for the multi-class context represents the complex nature of the encrypted darknet traffic more than learning process flaws. The reduced test performance and greater loss values confirm uncertainty when

identifying behaviorally identical classes under encryption, despite the training's high accuracy. Due to the peer-to-peer traffic and browsing having overlapping characteristics, this issue is notable. Additionally, this issue extends to class imbalance since dominating traffic categories impact the model's optimization more than the underrepresented classes.

In order to overcome these challenges, a number of regularization techniques were utilized—e.g., feature refinement, early stopping, and dropout—in addition to iterative hyperparameter tuning for training stabilization. Despite these steps reducing overfitting, there is still a small generalization gap, showing how difficult the multi-class discrimination is in an encrypted context. The binary darknet detection, on the other hand, shows better generalization due to more distinct behavioral separation. Table 5 summarizes the practical implications of these results as it links traffic categories to operational security monitoring scenarios and the noticed performance trends.

Table 5. Security-oriented deployment scenarios.

Deployment Context	Detected Activity	Relevant Classes	Observed Performance
Enterprise Network	Suspicious encrypted browsing	Browsing	Moderate recall due to behavioral overlap
Enterprise Network	P2P-based covert communication	P2P	High detection reliability
ISP Monitoring	Darknet access via Tor	Tor-based traffic	High binary detection accuracy
ISP Monitoring	Encrypted streaming vs. darknet	Streaming	Stable classification

7. Limitations

The primary limitations of the proposed approach are summarized in Table 6 to avoid lengthy textual descriptions.

Table 6. Summary of identified limitations.

Aspect	Limitation
Dataset Scope	Restricted to specific time periods and services.
Generalization	Not yet validated on emerging protocols.
Traffic Obfuscation	Performance may degrade under advanced padding or morphing.
Class Similarity	Browsing and P2P remain challenging.

The lower recognition rate of browsing traffic (47%) is attributed to overlapping temporal and packet-size characteristics with streaming and encrypted background flows. This overlap reduces inter-class separability within the selected feature space, suggesting the need for higher-order behavioral features in future work.

8. Comparison with Other Related Studies

Table 7 presents a detailed comparative evaluation between the proposed CNN–BiLSTM model and three representative baselines (random forest, standard CNN, and LSTM), reporting accuracy, precision, recall, and F1-scores under identical evaluation settings.

Furthermore, Table 8 summarizes the comparative performance analysis of encrypted and darknet traffic classification methods.

Table 7. Comparative performance evaluation.

Model/Reference	Dataset	Architecture Type	Accuracy (%)	Precision	Recall	F1-Score	Key Strength	Limitation
1D CNN [48]	Unified Dataset	Deep CNN	82.1	0.81	0.79	0.80	Automatic feature learning	Ignores sequential dependencies
DeepPacket [49]	ISCXVPN2016	1D CNN + SAE	85 *	–	–	–	Raw encrypted traffic modeling	Dataset-specific
DIDarknet [50]	Darknet Image Dataset	2D CNN	86.5	0.85	0.84	0.84	Image-based representation	No hybrid temporal modeling
Proposed CNN–BiLSTM	Unified VPN–Tor Dataset	2D CNN + BiLSTM	89.0	0.88	0.86	0.87	Unified dataset + spatial–temporal modeling	Higher computational cost

Table 8. Benchmarking against Related Studies.

Ref. No.	Dataset Used	Techniques/Algorithms	Measurements/Evaluation	Pros	Cons	Limitations
[13]	Not specified (early darknet traces)	Initial-packet-based detection using packet size, direction, and early connection features	Early-phase packet analysis	Low overhead; fast screening	Vulnerable to packet padding and spoofing	Works only on initial packet phases; limited scalability
[14]	Early darknet datasets	Decision trees	Classification accuracy	Easy to interpret; baseline method	Limited robustness vs. evolving attacks	Struggles with encrypted or modern anonymized traffic
[19]	Single-flow datasets	Single-flow behavioral analysis	Flow-level detection	Fast, computationally lightweight	Ignores multi-flow behavior	Ineffective for coordinated/complex attacks
[18]	Session-level darknet traces	Multi-packet/session flow modeling	Temporal pattern extraction	Captures richer temporal behavior	More resource-intensive	Requires full session data, often unavailable
	Rule-based classification datasets	Rule-based threat categorization	Threat grouping	Easy to apply; structured	Static rules degrade	Cannot detect novel or hybrid attacks
[51]	Forensic artefacts generated from controlled deep and dark web browsing scenarios across multiple platforms (Windows, Linux, Android, iOS) using TOR and privacy-preserving browsers	Proposed D2WFP protocol combining host-based digital forensics, memory forensics, browser artifact analysis, network traffic inspection, and artefact correlation	Quantitative comparison of artifacts recovered using D2WFP versus standard automated forensic tools; qualitative validation across multiple scenarios and operating systems	Provides a structured and comprehensive forensic protocol; improves artefact recovery compared to conventional tools; supports cross-validation and timeline reconstruction; applicable across different OS platforms	Not designed for real-time detection; relies on post-incident forensic acquisition; focuses primarily on host-side evidence rather than live network monitoring	Limited generalization to large-scale operational environments; evaluation conducted on simulated scenarios; does not integrate machine learning or automated classification for traffic analysis
[15]	Large-scale darknet packet captures	2D features; clustering; signature matching	Accuracy for known malware	High precision for known threats	Fails vs. new malware	Localized dataset; poor generalization
[20]	Aggregate darknet traffic	Packet freq., unique IP counts	Anomaly spotting	Good for mass scans	Low precision for low traffic	Cannot separate benign vs. malicious anomalies
[17]	Time series darknet logs	Attack clustering; temporal modeling	Pattern periodicity	Detects repeated attack waves	Poor with irregular attacks	Needs continuous and stable data
[26]	Survey (no dataset)	Honeyd environments; time series overview	Conceptual mapping	Broad methodological overview	No experiments	No new detection models

Table 8. *Cont.*

Ref. No.	Dataset Used	Techniques/Algorithms	Measurements/Evaluation	Pros	Cons	Limitations
[24]	Leaked darknet data	Identifier extraction (names, domains)	Profile extraction	Strong malicious-user insights	Privacy concerns	Dependent on the availability of leaked data
[16]	Darknet market text	Text mining: threat dictionary	Threat discovery	Detects emerging threats	Heavy noise in text	Fails when markets disappear/migrate
[22]	Attacker behavior logs	Stochastic/ probabilistic modeling	Attack likelihood estimation	Quantitative attacker modeling	Inefficient for distributed attacks	Cannot model large-scale probing reliably
[25]	Tor/I2P/ JonDonym datasets	Hierarchical classification	F-score ≈ 75.56%	Good cross-network separation	Moderate accuracy	Overlaps due to encryption uniformity
[27]	SSH/Skype encrypted traffic	AdaBoost, GP, C4.5	Protocol recognition	Accurate without payloads	Algorithms vary by traffic type	Limited to older protocols
[28]	Skype flows	Lightweight online classification	Real-time detection	High accuracy, low cost	Single-application focus	Not general-purpose
[29]	SSL traffic	SSL decryption	Content-based visibility	High inspection accuracy	Breaks privacy	Not scalable; heavy overhead
[30]	Encrypted traffic	Blind-box metadata analysis	Non-decryption classification	Protects privacy; no crypto overhead	Weak vs. heavy obfuscation	Fails when metadata is restricted
[31]	SSL/TLS	Certificate-based bigram model; Markov chain	TPR↑ 29%, FPR↓ 25%	High detection accuracy	Complex; preprocessing heavy	Depends on certificate visibility
[32]	ISCXVPN2016	1D and 2D CNN, C4.5	VPN vs. non-VPN accuracy (92%/85%)	Strong feature learning	Computationally expensive	Dataset-specific tuning needed
[33]	Encrypted traffic	Feature elimination: SVM, RF, XGBoost	Reduced model complexity	Efficient; low overhead	May lose fine-grained patterns	Poor with unseen traffic
[34]	Raw encrypted traffic	DeepPacket (1D CNN + SAE)	App ID (98%), Traffic type (93%)	Automatic feature extraction	Requires large training sets	Sensitive to encryption updates
[35]	Policy-driven filtering	Device-level identification rules	Access control	Fine-grained control	Requires endpoint integration	Not scalable for large networks
[36]	VPN traffic	Dual-certificate VPN handshake	Key-exchange-based classification	Maintains encryption security	Complex deployment	Requires endpoint cooperation
[37]	ISCXVPN2016	Time-based flow features; C4.5; kNN	~80% accuracy	Benchmark dataset	Basic ML only	Dataset aging; limited apps
[38]	ISCXVPN2016	Ensemble models (RF, GBT)	Higher VPN discrimination	Better accuracy	Higher computational cost	Depends on hand-crafted features
[39]	OpenVPN traces	MLP neural network	>92% accuracy	Effective for OpenVPN	Not multi-protocol	Retraining is needed for new protocols
[40]	QoS-marked VPN	PHB/QoS classification	94% non-VPN, 92% VPN	Very high accuracy	Requires QoS integration	Breaks under traffic shaping
[1]	Device memory	Forensic memory analysis	Tor trace extraction	Reveals sensitive metadata	Requires device seizure	Not remote-applicable

Table 8. *Cont.*

Ref. No.	Dataset Used	Techniques/Algorithms	Measurements/Evaluation	Pros	Cons	Limitations
[2]	Tor routing	LASTor modified path selection	Latency leakage mitigation	Reduces timing attacks	Partial protection	Not universal vs. timing threats
[52]	Tor HTTP	MITM attack via the exit node	Real-world feasibility	Demonstrates attack paths	Requires exit-node control	Only non-HTTPS traffic
[53]	Tor protocol	Malformed Tor cells	Protocol weakness exposure	Insights into Tor internals	Disruptive, detectable	Applies to older versions
[49]	Tor circuits	Latency-based side-channel	Traffic and circuit inference	Non-invasive	Noise-sensitive	Accuracy drops with congestion
[54,55]	Tor circuits	Burst/timing features	App inference	Effective on burst patterns	Padding defeats it	Weak vs. uniform encrypted traffic
[56]	Tor exit nodes	TorWard IDS	Large-scale malicious detection	Captures botnets, spam	Deployment overhead	Exit-node only; cannot see onion layers
[57]	Tor/I2P/JonDonym	Feature-based hierarchical model	Cross-anonymity classification	Good multi-tool recognition	Dataset reliance	Not robust to new tools
[58]	Literature survey	Systematic Tor review	Research landscape mapping	Comprehensive	No new detection	Limited experimental insights
[59]	Tor website traffic	Adaptive stream mining	Website fingerprinting	High accuracy	Privacy-invasive	Breakable with defenses
[60]	Mobile devices	Battery consumption analysis	Traffic inference	Works without network access	Device-dependent	Not generalizable
[61]	Tor traces + image leaks	Image deanonymization	Multimedia identification	Shows privacy leakage	Needs leaked images	Limited to image-heavy traffic
[62]	Bitcoin + Tor	Blockchain correlation	Hidden service deanonymization	Financial linkage detection	Requires blockchain visibility	Fails with mixers/privacy coins
Proposed System	**Merged ISCXVPN2016 + ISCXTor2017 (Unified Dataset)**	**Hybrid 2D CNN + feature refinement + behavior analysis**	**Binary: 94% acc.; multi-class: 86%; Loss: 0.17/0.50; Statistical behavior analysis**	**Unified VPN + Tor + Darknet detection; deep feature extraction; hidden-service behavior discovery; robust generalization**	**Requires image transformation; higher computational load than classic ML**	**Results depend on dataset diversity, browsing category is still weaker (≈47% recall)**

9. Conclusions and Future Directions

This work extended prior research on encrypted traffic classification by jointly addressing dataset construction and model design. Existing image-based approaches focus on a single anonymization technique or static feature representations; however, the proposed framework leveraged a unified VPN–Tor dataset and a hybrid CNN–BiLSTM architecture to model spatial feature relationships and temporal traffic evolution. The achieved results showed that integrating sequence modeling with image-based representations enhances system robustness in the context of multi-class darknet traffic classification. The proposed framework offers a strong platform for secure monitoring systems in encrypted environments in the future.

The conducted experiments prove that the image-based method outperformed the traditional one-dimensional models, significantly when capturing the encrypted traffic's spatial feature. Further, peer-to-peer detection remains a challenging issue. Hence, the model achieved good recognition with only a few false negatives. Furthermore, hyperparameter adjustment increased the model stability, and a number of behavioral indicators (particularly directional packet rates) proved useful for distinguishing the darknet traffic patterns.

Moreover, this paper addressed the shortcomings in the existing datasets, as they failed to adequately capture the complexity of actual darknet activity, multi-layer encryption configurations, or upgradable behavior of hidden services.

Future directions may include the development of larger, varied, and anonymized traffic collections that represent different tunneling setups and new encrypted protocols. This would strengthen model generalization and support more effective forensic and cybersecurity applications.

The lower recognition rate of browsing traffic (47%) is attributed to overlapping temporal and packet-size characteristics with streaming and encrypted background flows. This overlap reduces inter-class separability within the selected feature space, suggesting the need for higher-order behavioral features in future work.

Author Contributions: Conceptualization, D.N.M.; methodology, D.N.M. and H.W.O.; software, D.N.M.; validation, D.N.M., H.W.O. and H.A.-R.; formal analysis, D.N.M.; investigation, D.N.M.; resources, H.W.O.; data curation, D.N.M.; writing—original draft preparation, D.N.M.; writing—review and editing, H.W.O. and H.A.-R.; visualization, D.N.M.; supervision, H.A.-R.; project administration, H.A.-R.; funding acquisition, H.A.-R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets analyzed during the current study are publicly available from their original repositories cited within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mirea, M.; Wang, V.; Jung, J. The not so dark side of the darknet: A qualitative study. *Secur. J.* **2019**, *32*, 102–118. [CrossRef]
2. Svilicic, B.; Kamahara, J.; Rooks, M.; Yano, Y. Maritime Cyber Risk Management: An Experimental Ship Assessment. *J. Navig.* **2019**, *72*, 1108–1120. [CrossRef]
3. Cynthia, J. A Survey on Deep Learning Techniques for Darknet Traffic Malware Detection. *Int. J. Sci. Res. Eng. Manag.* **2023**, *7*, 24715. [CrossRef]
4. Niranjana, R.; Kumar, V.A.; Sheen, S. Darknet Traffic Analysis and Classification Using Numerical AGM and Mean Shift Clustering Algorithm. *SN Comput. Sci.* **2020**, *1*, 16. [CrossRef]
5. Safaei Pour, M.; Mangino, A.; Friday, K.; Rathbun, M.; Bou-Harb, E.; Iqbal, F.; Samtani, S.; Crichigno, J.; Ghani, N. On data-driven curation, learning, and analysis for inferring evolving internet-of-Things (IoT) botnets in the wild. *Comput. Secur.* **2020**, *91*, 101707. [CrossRef]

6. Cabana, O.; Youssef, A.M.; Debbabi, M.; Lebel, B.; Kassouf, M.; Atallah, R.; Agba, B.L. Threat Intelligence Generation Using Network Telescope Data for Industrial Control Systems. *IEEE Trans. Inf. Forensics Secur.* **2021**, *16*, 3355–3370. [CrossRef]
7. Spitters, M.; Verbruggen, S.; Staalduinen, M. Van Towards a comprehensive insight into the thematic organization of the tor hidden services. In Proceedings of the 2014 IEEE Joint Intelligence and Security Informatics Conference, JISIC 2014, The Hague, The Netherlands, 24–26 September 2014; pp. 220–223.
8. Li, R.; Chen, S.; Yang, J.; Luo, E. Edge-based detection and classification of malicious contents in tor darknet using machine learning. *Mob. Inf. Syst.* **2021**, *2021*, 8072779. [CrossRef]
9. Beshiri, A.S.; Susuri, A. Dark Web and Its Impact in Online Anonymity and Privacy: A Critical Analysis and Review. *J. Comput. Commun.* **2019**, *7*, 30–43. [CrossRef]
10. Al-Nabki, M.W.; Fidalgo, E.; Alegre, E.; Fernández-Robles, L. ToRank: Identifying the most influential suspicious domains in the Tor network. *Expert. Syst. Appl.* **2019**, *123*, 212–226. [CrossRef]
11. Hashimoto, N.; Ozawa, S.; Ban, T.; Nakazato, J.; Shimamura, J. A Darknet Traffic Analysis for IoT Malwares Using Association Rule Learning. *Procedia Comput. Sci.* **2018**, *144*, 118–123. [CrossRef]
12. Kanemura, K.; Toyoda, K.; Ohtsuki, T. Identification of Darknet Markets' Bitcoin Addresses by Voting Per-Address Classification Results. In Proceedings of the ICBC 2019—IEEE International Conference on Blockchain and Cryptocurrency, Seoul, Republic of Korea, 14–17 May 2019; pp. 154–158.
13. Habibi Lashkari, A.; Kaur, G.; Rahali, A. DIDarknet: A contemporary approach to detect and characterize the darknet traffic using deep image learning. In Proceedings of the 15th International Conference on Availability, Reliability and Security (ARES), Dublin, Ireland, 25–28 August 2020; pp. 1–13. [CrossRef]
14. Acar, A.; Liu, W.; Beyah, R.; Akkaya, K.; Uluagac, A.S. A privacy-preserving multifactor authentication system. *Secur. Priv.* **2019**, *2*, e88. [CrossRef]
15. Nishikaze, H.; Ozawa, S.; Kitazono, J.; Ban, T.; Nakazato, J.; Shimamura, J. Large-scale monitoring for cyber attacks by using cluster information on darknet traffic features. *Procedia Comput. Sci.* **2015**, *53*, 175–182. [CrossRef]
16. Dong, F.; Yuan, S.; Ou, H.; Liu, L. New Cyber Threat Discovery from Darknet Marketplaces. In Proceedings of the 2018 IEEE Conference on Big Data and Analytics, ICBDA 2018, Langkawi, Malaysia, 21–22 November 2018; pp. 62–67.
17. Choudhary, M.; Tiwari, V.; Uduthalapally, V. Iris presentation attack detection based on best-k feature selection from YOLO inspired RoI. *Neural Comput. Appl.* **2021**, *33*, 5609–5629. [CrossRef]
18. Saleem, J.; Islam, R.; Islam, M.Z. Darknet Traffic Analysis: A Systematic Literature Review. *IEEE Access* **2024**, *12*, 42423–42452. [CrossRef]
19. Ji, I.H.; Lee, J.H.; Kang, M.J.; Park, W.J.; Jeon, S.H.; Seo, J.T. Artificial Intelligence-Based Anomaly Detection Technology over Encrypted Traffic: A Systematic Literature Review. *Sensors* **2024**, *24*, 898. [CrossRef]
20. Gadhia, F.; Choi, J.; Cho, B.; Song, J. Comparative analysis of darknet traffic characteristics between darknet sensors. In Proceedings of the International Conference on Advanced Communication Technology, ICACT, PyeongChang, Republic of Korea, 1–3 July 2015; pp. 59–64.
21. Akiyoshi, R.; Kotani, D.; Okabe, Y. Detecting Emerging Large-Scale Vulnerability Scanning Activities by Correlating Low-Interaction Honeypots with Darknet. In *Proceedings of the International Computer Software and Applications Conference, Tokyo, Japan, 23–27 July 2018*; IEEE: New York, NY, USA, 2018; Volume 2, pp. 658–663.
22. Safaei Pour, M.; Bou-Harb, E. Theoretic derivations of scan detection operating on darknet traffic. *Comput. Commun.* **2019**, *147*, 111–121. [CrossRef]
23. Fachkha, C.; Bou-Harb, E.; Debbabi, M. Inferring distributed reflection denial of service attacks from darknet. *Comput. Commun.* **2015**, *62*, 59–71. [CrossRef]
24. Wang, M.; Wang, X.; Shi, J.; Tan, Q.; Gao, Y.; Chen, M.; Jiang, X. Who are in the darknet? Measurement and analysis of Darknet person attributes. In Proceedings of the 2018 IEEE 3rd International Conference on Data Science in Cyberspace, DSC 2018, Guangzhou, China, 18–21 June 2018; pp. 948–955.
25. Georgoulias, D.; Yaben, R.; Vasilomanolakis, E. Cheaper than you thought? A dive into the darkweb market of cyber-crime products. In *Proceedings of the ACM International Conference Proceeding Series*; Association for Computing Machinery (ACM): New York, NY, USA, 2023.
26. Fachkha, C.; Debbabi, M. Darknet as a Source of Cyber Intelligence: Survey, Taxonomy, and Characterization. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 1197–1227. [CrossRef]
27. Alshammari, R.; Zincir-Heywood, A.N. Can encrypted traffic be identified without port numbers, IP addresses and payload inspection? *Comput. Netw.* **2011**, *55*, 1326–1350. [CrossRef]
28. Gu, C.; Zhang, S.; Sun, Y. Real-time encrypted traffic identification using machine learning. *J. Softw.* **2011**, *6*, 1009–1016. [CrossRef]
29. Alzahrani, A.O.; Alenazi, M.J.F. Designing a network intrusion detection system based on machine learning for software defined networks. *Future Internet* **2021**, *13*, 111. [CrossRef]
30. Sherry, J.; Lan, C.; Popa, R.A.; Ratnasamy, S. BlindBox: Deep Packet Inspection over Encrypted Traffic. *Comput. Commun. Rev.* **2015**, *45*, 213–226. [CrossRef]

31. Shen, M.; Wei, M.; Zhu, L.; Wang, M. Classification of Encrypted Traffic with Second-Order Markov Chains and Application Attribute Bigrams. *IEEE Trans. Inf. Forensics Secur.* **2017**, *12*, 1830–1843. [CrossRef]
32. Wang, W.; Zhu, M.; Wang, J.; Zeng, X.; Yang, Z. End-To-end encrypted traffic classification with one-dimensional convolution neural networks. In Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics: Security and Big Data, ISI 2017, Beijing, China, 22–24 July 2017; pp. 43–48.
33. Shekhawat, A.S.; Di Troia, F.; Stamp, M. Feature analysis of encrypted malicious traffic. *Expert Syst. Appl.* **2019**, *125*, 130–141. [CrossRef]
34. Lotfollahi, M.; Jafari Siavoshani, M.; Shirali Hossein Zade, R.; Saberian, M. Deep packet: A novel approach for encrypted traffic classification using deep learning. *Soft Comput.* **2020**, *24*, 1999–2012. [CrossRef]
35. Javanmardi, E.; Liu, S. Exploring grey systems theory-based methods and applications in analyzing socio-economic systems. *Sustainability* **2019**, *11*, 4192. [CrossRef]
36. Islam, F.U.; Liu, G.; Liu, W.; ul Haq, Q.M. A deep learning-based framework to identify and characterise heterogeneous secure network traffic. *IET Inf. Secur.* **2023**, *17*, 294–308. [CrossRef]
37. Balachandran, A.; Amritha, P.P. VPN Network Traffic Classification Using Entropy Estimation and Time-Related Features. In *Proceedings of the Smart Innovation, Systems and Technologies*; Springer Nature: Singapore, 2022; Volume 251, pp. 509–520.
38. Bagui, S.; Fang, X.; Kalaimannan, E.; Bagui, S.C.; Sheehan, J. Comparison of machine-learning algorithms for classification of VPN network traffic flow using time-related features. *J. Cyber Secur. Technol.* **2017**, *1*, 108–126. [CrossRef]
39. Miller, S.; Curran, K.; Lunney, T. Multilayer perceptron neural network for detection of encrypted VPN network traffic. In Proceedings of the 2018 International Conference on Cyber Situational Awareness, Data Analytics and Assessment, CyberSA 2018, Glasgow, UK, 11–12 June 2018.
40. Caicedo-Muñoz, J.A.; Ledezma Espino, A.; Corrales, J.C.; Rendón, A. QoS-Classifier for VPN and Non-VPN traffic based on time-related features. *Comput. Netw.* **2018**, *144*, 271–279. [CrossRef]
41. Rezaei, S.; Liu, X. Deep Learning for Encrypted Traffic Classification: An Overview. *IEEE Commun. Mag.* **2019**, *57*, 76–81. [CrossRef]
42. Draper-Gil, G.; Lashkari, A.H.; Mamun, M.S.I.; Ghorbani, A.A. Characterization of encrypted and VPN traffic using time-related features. In *Proceedings of the ICISSP 2016—Proceedings of the 2nd International Conference on Information Systems Security and Privacy*; SciTePress: Setúbal, Portugal, 2016; pp. 407–414.
43. Lashkari, A.H.; Gil, G.D.; Mamun, M.S.I.; Ghorbani, A.A. Characterization of tor traffic using time based features. In *Proceedings of the ICISSP 2017—Proceedings of the 3rd International Conference on Information Systems Security and Privacy; 2017*; SciTePress: Setúbal, Portugal, 2017; pp. 253–262.
44. Mhawi, D.N.; Oleiwi, H.W.; Al-Taie, H.L. Generating Encrypted Document Index Structure Using Tree Browser. *J. Tech.* **2023**, *5*, 114–122. [CrossRef]
45. Ferreira, E.W.T.; Shinoda, A.A. The development and evaluation of a dataset for testing of IDS for wireless networks. *IEEE Lat. Am. Trans.* **2016**, *14*, 404–410. [CrossRef]
46. Cermak, M.; Jirsik, T.; Velan, P.; Komarkova, J.; Spacek, S.; Drasar, M.; Plesnik, T. Towards Provable Network Traffic Measurement and Analysis via Semi-Labeled Trace Datasets. In Proceedings of the TMA 2018—Proceedings of the 2nd Network Traffic Measurement and Analysis Conference, Vienna, Austria, 26–29 June 2018.
47. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the ICISSP 2018—Proceedings of the 4th International Conference on Information Systems Security and Privacy*; SciTePress: Setúbal, Portugal, 2018; pp. 108–116.
48. Wang, Y.; Yan, W.; Oates, T. Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 1578–1585. [CrossRef]
49. Lashkari, H.; Draper-Gil, A.; Joshi, L.; Ghorbani, R. DIDarknet: A Deep Learning-Based Network Intrusion Detection System for Darknet Traffic. *arXiv* **2020**, arXiv:2004.03065.
50. Ghanem, M.C.; Mulvihill, P.; Ouazzane, K.; Djemai, R.; Dunsin, D. D2WFP: A Novel Protocol for Forensically Identifying, Extracting, and Analysing Deep and Dark Web Browsing Activities. *J. Cybersecur. Priv.* **2023**, *3*, 808–829. [CrossRef]
51. Wang, X.; Luo, J.; Yang, M.; Ling, Z. A potential HTTP-based application-level attack against Tor. *Future Gener. Comput. Syst.* **2011**, *27*, 67–77. [CrossRef]
52. Ling, Z.; Luo, J.; Yu, W.; Fu, X.; Jia, W.; Zhao, W. Protocol-level attacks against Tor. *Comput. Netw.* **2013**, *57*, 869–886. [CrossRef]
53. Liška, T.; Sochor, T.; Sochorová, H. Comparison between normal and TOR-anonymized web client traffic. *Procedia Comput. Sci.* **2011**, *3*, 888–892. [CrossRef]
54. Shahbar, K.; Zincir-Heywood, A.N. Benchmarking two techniques for Tor classification: Flow level and circuit level classification. In Proceedings of the IEEE SSCI 2014: 2014 IEEE Symposium Series on Computational Intelligence—CICS 2014: 2014 IEEE Symposium on Computational Intelligence in Cyber Security, Proceedings, Orlando, FL, USA, 9–12 December 2014.

55. He, G.; Yang, M.; Luo, J.; Gu, X. Inferring Application Type Information from Tor Encrypted Traffic. In Proceedings of the 2014 2nd International Conference on Advanced Cloud and Big Data, CBD 2014, Huangshan, China, 20–22 November 2014; pp. 220–227.
56. Ling, Z.; Luo, J.; Wu, K.; Yu, W.; Fu, X. TorWard: Discovery, Blocking, and Traceback of Malicious Traffic over Tor. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2515–2530. [CrossRef]
57. Montieri, A.; Ciuonzo, D.; Aceto, G.; Pescape, A. Anonymity Services Tor, I2P, JonDonym: Classifying in the Dark (Web). *IEEE Trans. Dependable Secur. Comput.* **2020**, *17*, 662–675. [CrossRef]
58. Saleh, S.; Qadir, J.; Ilyas, M.U. Shedding Light on the Dark Corners of the Internet: A Survey of Tor Research. *J. Netw. Comput. Appl.* **2018**, *114*, 1–28. [CrossRef]
59. Attarian, R.; Abdi, L.; Hashemi, S. AdaWFPA: Adaptive Online Website Fingerprinting Attack for Tor Anonymous Network: A Stream-wise Paradigm. *Comput. Commun.* **2019**, *148*, 74–85. [CrossRef]
60. Yang, Q.; Gasti, P.; Balagani, K.; Li, Y.; Zhou, G. USB side-channel attack on Tor. *Comput. Netw.* **2018**, *141*, 57–66. [CrossRef]
61. Fidalgo, E.; Alegre, E.; Fernández-Robles, L.; González-Castro, V. Classifying suspicious content in tor darknet through Semantic Attention Keypoint Filtering. *Digit. Investig.* **2019**, *30*, 12–22. [CrossRef]
62. Al Jawaheri, H.; Al Sabah, M.; Boshmaf, Y.; Erbad, A. Deanonymizing Tor hidden service users through Bitcoin transactions analysis. *Comput. Secur.* **2020**, *89*, 101684. [CrossRef]