Data Engineering and Applications

Research Paper

LightGBM-Based Encrypted Traffic Classification: Interpretable Feature Fusion and Robust Evaluation for Modern Networks

Sławomir Feliks Jarosz[1, *]

[1] Faculty of Computer Science and Telecommunications, Tadeusz Kościuszko Cracow University of Technology, Kraków, 31-155, Poland
*Corresponding author: slawomir.j@pk.edu.pl

Abstract. As services expand, encrypted network traffic and fine-grained classification become necessary to meet quality of service and security requirements. First, this paper will introduce the shortcomings of traditional classification methods. Then, a multi-class encrypted traffic processing framework based on LightGBM will be proposed. It aims to improve detection accuracy while maintaining the model's practicality and interpretability. The three directions of package-level, flow-level, and entropy-driven approaches have all undergone rigorous cross-validation and comparison with other ensemble and kernel methods. The overall accuracy and recall of the LightGBM model surpassed the baseline classifier, using a large real-world traffic dataset for experimentation. The performance on long-tail and minority traffic categories has also improved. Feature ablation analysis indicates that combining multiple features can improve performance; SHAP and LIME can provide clear and interpretable explanations for classification decisions. Experiments show that this is an excellent predictor that can be applied in real-world environments with hostile conditions and low latency. In order to meet the needs of the next generation of security infrastructure, this study will support the construction of a traffic analysis system that is scalable, interpretable, and stable.

Keywords: *Network Security, Encrypted Traffic, Machine Learning, Feature Engineering, Model Interpretability, Real-Time Detection*

Received on 10 October 2024, Accepted on 14 April 2025, Published on 21 April 2025

Introduction

With the widespread adoption of encryption technology, HTTPS has become increasingly secure in recent years, and many networks are now encrypted. According to a recently released report, over 90% of internet traffic is currently encrypted using the TLS protocol to protect user privacy and reduce network visibility [1]. Although this change will protect data, it will also hinder the methods previously used to monitor intrusions and malicious traffic for legal review [2]. Complex cyber attackers are increasingly using encrypted channels for secret communication, data breaches, and preventing malware operations [3]. This has led to greater risks for government and corporate information systems and critical infrastructure. As the heterogeneity and scale of HTTPS applications increase, some technical issues have emerged, such as feature sparsity and reduced traffic visibility. Real-time encrypted traffic analysis is both necessary and challenging [4]. Packet padding, traffic multiplexing, and certificate pinning are also widely used to increase the difficulty of deep packet inspection and undermine traditional security analysis [5].

Research in both academia and the business sector is currently focused on developing advanced analytical techniques that can provide secure information while protecting privacy. Subsequently, traditional machine learning models such as support vector machines and random forests were used, with manually extracted statistical features from the data in the initial experiments, such as packet size distribution, flow duration, and

arrival interval time [6]. The aforementioned methods are effective to some extent, but they cannot adapt to changes in application models, as well as changes in encryption protocols and other obfuscation techniques [7]. With the development of deep learning and the emergence of structures such as convolutional networks and recurrent networks, automatically extracting features from encrypted streams has also become a reality [8]. Solutions often face issues of interpretability, high computational costs, and limitations in generalizing to different types of traffic, although they show some potential in extracting hidden representations without explicit load inspection [9]. Most methods do not leverage the complementary advantages of deep feature space and handcrafted feature space to use them within a single model [10].

In light of the aforementioned issues, this paper proposes a new traffic classification framework. This framework systematically integrates domain-driven handcrafted features and high-order representations learned by deep neural networks. By strategically integrating various feature sets, a high-performance, interpretable LightGBM ensemble model is constructed. To address these issues, this paper proposes a new traffic classification framework. The framework combines domain-driven handcrafted features with high-level representations learned by deep neural networks. In order to create a universal traffic feature descriptor, this study employed a dual-path feature extraction and fusion method. The descriptor combines deep learning methods and statistical methods. Using a set of features, an efficient and scalable ensemble classifier was constructed, which is both accurate and easy to interpret, based on high-performance gradient boosting. In many experiments, multiple real-world datasets were used, and both the classification accuracy and the ease of implementation were satisfactory. On this basis, the supplementary model identifies the driving factors behind these decisions. Based on the research findings and their practical applications, this paper identifies the current shortcomings of encrypted traffic analysis techniques and proposes new research directions.

Related Work and Motivation

Classical Machine Learning in Traffic Analysis

Traditional machine learning methods, such as support vector machines, random forests, and k-nearest neighbors, are used in the initial study of encrypted traffic analysis to classify network traffic based on collected statistical features [11]. These methods are usually based on identifiable parameters in encrypted sessions, such as packet size, flow speed, and arrival interval time [12]. The aforementioned methods are not always effective in complex or covert traffic [13]. Due to insufficient understanding of the subject, the feature selection process is very simple and may not capture all types of network traffic [14]. The ability to generalize in dynamic environments is still very weak, especially with the introduction of new protocols or evasion methods [15]. With the increase in encrypted traffic, the limitations of traditional machine learning are becoming more apparent, necessitating models that are more expressive and adaptive [16].

Deep Learning Approaches for Encrypted Traffic

Recurrent architectures and convolutional neural networks have recently been used to study encrypted traffic classification, with the development of deep learning [17]. By using these models, the need for manual feature engineering can be reduced, thereby automatically learning the hierarchical and temporal relationships between the data [18]. In deep learning, many classification tasks of encrypted streams perform well [19]. Typically, large labeled datasets and considerable computational resources are required, making these models potentially unsuitable for real-time use [20]. The complexity of complex neural networks is not suitable for security-sensitive applications [21]. It has been found that performance significantly declines when encountering previously unseen categories or network conditions. The robustness and transferability of current deep learning solutions are also being questioned [22].

Limitations and Gaps in Current Research

Despite the recent publication of numerous results, there are still some issues with encrypted traffic analysis. Many studies still use handcrafted statistical features and deep representations, ignoring the potential benefits of integration [23]. Since this method only uses one type of feature extraction, its performance and adaptability decline in complex traffic environments [24]. There is a trade-off between achieving high classification accuracy and real-time prediction efficiency, which is not very effective for safe operation [25]. The aforementioned

serious shortcomings provide a strong rationale for the current research on hybrid frameworks, which combine the advantages of both domains to build robust, large-scale solutions for modern network environments.

Hybrid Feature Fusion Framework

Handcrafted Feature Engineering

In order to handle the temporal and structural properties of encrypted network sessions at different scales, it is necessary to add statistical properties to the model to extract intrinsic features. For any session, consider a sequence of packets $S = \{(t_i, s_i, d_i)\}_{i=1}^{N}$, where t_i is the timestamp, s_i the packet size, and d_i the direction (+1 for outbound, -1 for inbound). A good representation starts with the entropy of packet sizes from the session, which can be expressed as:

$$H_{\text{size}} = -\sum_{k} p_k \log(p_k) \quad \text{Eq.(1)}$$

where p_k is the empirical probability mass function of the k-th observed packet size. Entropy can be used to distinguish between normal application flows and potentially hidden transmissions.

The normalized average arrival time fluctuations indicate temporal burstiness and regularity:

$$\phi_{IAT} = \frac{1}{N-2}\sum_{i=2}^{N-1}\left|\frac{(t_{i+1} - t_i) - (t_i - t_{i-1})}{\sigma_{IAT}}\right| \quad \text{Eq.(2)}$$

where σ_{IAT} is the standard deviation of all inter-arrival times within the session. The volatility of a session can be used to determine whether it is scripted or automated.

The directional variance ratio can show the directional asymmetry of the session, which is usually related to application logic or data leakage.

$$R_{\text{dirvar}} = \frac{\text{Var}(\mathcal{S}_{fwd})}{\text{Var}(\mathcal{S}_{fwd}) + \text{Var}(\mathcal{S}_{\text{bwd}})} \quad \text{Eq.(3)}$$

where $\mathcal{S}_{fwd}$ and $\mathcal{S}_{bwd}$ are sets of outbound and inbound packet sizes respectively.

Quantile normalization makes features follow a normal distribution across different sessions.

$$Q_f^* = \frac{Q_{f,0.75} - Q_{f,0.25}}{Q_{f,0.50}} \quad \text{Eq.(4)}$$

where $Q_{f,q}$ denotes the q-th quantile of feature f in a batch window. This normalization is not very sensitive to changes in traffic distribution or outliers.

Deep Feature Extraction via Neural Networks

Due to the almost complete lack of manually designed features, deep neural networks can easily learn the complex and nonlinear relationships in encrypted network data. First, each network session is preprocessed and formatted into a sequence matrix. Normalize attributes such as packet size, arrival distance, and transmission direction to generate the rows of the matrix. Then, the sequentially input data is transmitted to a multi-layer one-dimensional convolutional neural network (1D-CNN). Convolutional filters are used to scan the packet sequence, hierarchically extracting local and global temporal patterns relevant for distinguishing traffic types.

The system uses an attention-based pooling mechanism to prevent information loss during aggregation. It also automatically assigns different weights to each session area. Don't simply use pooling or averaging to summarize the sequence. Instead, use learned attention scores to create a weighted representation, which is more distinctive for the next classification.

The global session embedding vector, denoted as z_{deep}, is generated using the following attention-weighted integration over all sequence positions:

$$z_{\text{deep}} = \sum_{j=1}^{L} \frac{\exp\left(v^{\top}\tanh\left(W_{\text{att}}F_j + b_{\text{att}}\right)\right)}{\sum_{k=1}^{L}\exp\left(v^{\top}\tanh\left(W_{\text{att}}F_k + b_{\text{att}}\right)\right)} F_j \qquad \text{Eq.(5)}$$

In this expression, F_j is the contextual deep feature vector at position j output by the final convolutional layer, W_{att} and b_{att} are attention-specific learnable parameters, and v is a projection vector that produces a scalar compatibility score. The Softmax normalization coefficient ensures that the combined embeddings are differentiable and fully adapt to the data; in other words, significant time intervals will receive higher weights rather than being treated equally. Due to this attention mapping, the model tends to favor protocol handshake, encryption handshake anomalies, or frequent time bursts related to service-specific signatures or anomalies.

Backpropagation is used to simultaneously update the parameters of the deep convolutional extractor and the attention mechanism to enhance feature discrimination. Deep sessions indicate that it is possible to learn high-order patterns that are difficult to obtain manually. In addition to manual features, these patterns also need to analyze encrypted, variable, and potentially adversarial network traffic. By combining the above methods with rich contextual embeddings, the system can be applied to new domains and resist adversarial protocol imitation.

Feature Fusion Strategies and Pipeline

Comprehensive representation is a combination of domain statistical awareness and deep perception. Here, adaptive calibration based on sample data is used. The fusion vector is as follows:

$$z_{\text{fusion}} = \gamma \cdot z_{\text{stat}} + (1-\gamma) \cdot z_{\text{deep}} \qquad \text{Eq.(6)}$$

where z_{stat} is the handcrafted feature vector, z_{deep} the neural embedding, and the adaptive fusion parameter γ is dynamically learned to optimize validation loss. This coupling leverages the expressiveness of latent neural encoding while maintaining the interpretability of structural features.

This hybrid model can dynamically adapt, which is advantageous compared to traditional methods of simple feature concatenation or using fixed linear weights (without considering the current traffic distribution and operating environment). In contrast, this framework has context-aware capabilities, allowing it to dynamically adjust fusion calibration and network parameters during inference. In the case of adversarial perturbations or sudden changes in operating conditions, increase the weight of robust handcrafted features, or rely more on deep features to identify new and unknown behaviors. Compared to single-path or coarse ensemble models, it has a certain degree of flexibility and fault tolerance.

Figure 1 shows the interaction between the entire process and its components. Network session data is processed by engineered statistical logic and deep neural encoders, then smoothed and merged through dynamic calibration, and finally classified. The aforementioned explicit architectural arrangement is designed to demonstrate how the functions operate and to provide basic standards for the expansion and interpretation of the network monitoring environment.

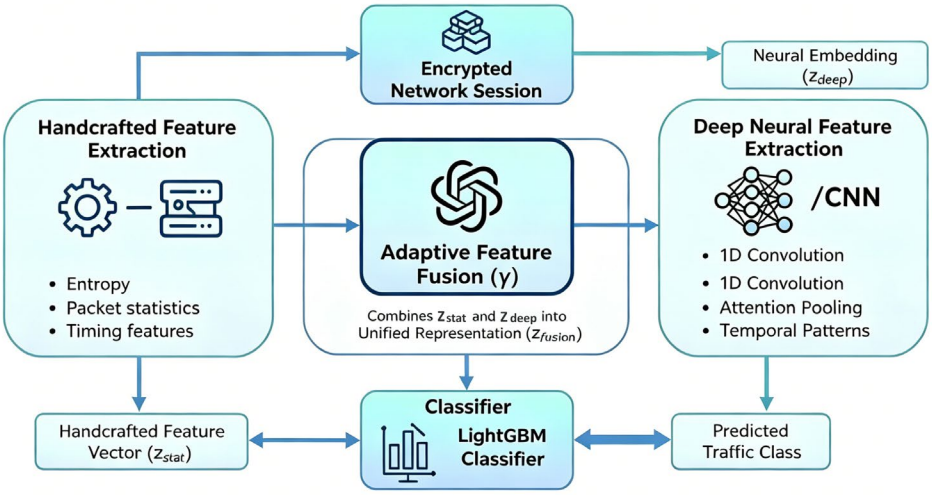

Figure 1. Hybrid Feature Fusion Architecture

https://doi.org/10.64972/dea.2025.v4i2.1994d:43-56

Proposed LightGBM-Based Classification

Algorithm Design and Parameter Optimization

LightGBM has redesigned gradient boosting to classify encrypted traffic features in a scalable and efficient manner. Let the training set be $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, with $\mathbf{x}_i$ denoting feature vectors and y_i representing class labels. At each improvement stage, model integration is recursively defined as

$$F_t(\mathbf{x}) = F_{t-1}(\mathbf{x}) + \nu h_t(\mathbf{x}) \quad \text{Eq.(7)}$$

where F_t is the prediction function at round t, ν the shrinkage (learning rate), and $h_t(\mathbf{x})$ the newly added decision tree.

Tree construction in LightGBM is governed by maximizing an objective-based gain metric derived from the sum of first (G) and second (H) derivatives (gradient, Hessian) of the loss with respect to predicted values. The gain for splitting a node into left (L) and right (R) children at candidate split s is:

$$\text{Gain}_s = \frac{1}{2}\left(\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda}\right) - \gamma \quad \text{Eq.(8)}$$

where λ is the regularization term and γ a minimum gain threshold for split acceptance.

In order to adapt to traffic encryption, grid search and Bayesian optimization are used for hyperparameter tuning. To verify the minimal performance loss, grid search directly searches the parameter combinations θ:

$$\theta^* = \arg\min_{\theta \in \mathcal{S}} \mathcal{L}_{val}(\theta) \quad \text{Eq.(9)}$$

where $\mathcal{S}$ is the configuration space. Bayesian approaches model $\mathcal{L}_{\text{val}}$ as a stochastic process, allowing efficient exploration and convergence.

By using leaf-wise growth and histogram binning, LightGBM aims for large-scale deployment, thereby reducing computational costs and memory overhead while lowering the risk of overfitting. Empirical results from many streaming experiments indicate that runtime and memory usage have decreased, and model complexity has been reduced. Very suitable for current real-time and adversarial network engineering problems.

Model Training and Validation

To reliably assess generalization, the full dataset $\mathcal{D}$ is divided into training ($\mathcal{D}_{tr}$) and validation ($\mathcal{D}_{\text{val}}$) portions. K-fold cross-validation is used, partitioning $\mathcal{D}$ into folds $\mathcal{D}_1, \ldots, \mathcal{D}_K$, and cyclically evaluating model performance across all folds,

$$\text{CV} = \frac{1}{K}\sum_{k=1}^{K} \mathcal{M}_{val}^{(k)} \quad \text{Eq.(10)}$$

where $\mathcal{M}_{\text{val}}^{(k)}$ is the metric (e.g., loss, accuracy) for fold k. Early stopping halts boosting iterations when validation loss does not improve beyond tolerance ϵ over a patience window T_p :

$$\mathcal{L}_{val}^{(t)} > \min_{0 \le s < t} \mathcal{L}_{val}^{(s)} + \epsilon \quad \text{Eq.(11)}$$

for $t \geq t_0$. Overall validation accuracy is:

$$\text{Accuracy} = \frac{1}{N_{\text{val}}}\sum_{i=1}^{N_{val}} \mathbb{I}(\hat{y}_i = y_i) \quad \text{Eq.(12)}$$

where $\mathbb{I}$ is the indicator and N_{val} is validation set size. The calculation methods for auxiliary metrics (AUC, F1) used in comprehensive bias-variance analysis are the same.

Comparative Baseline Experiments

All experiments were conducted in a highly standardized manner and were directly and algorithmically compared with existing classifiers such as Random Forest, Support Vector Machine (SVM), and traditional GBDT to support the advantages of the proposed LightGBM model. To avoid confounding factors and maintain empirical reliability, the algorithm uses the same mixed feature inputs, stratified training-validation splits, and standardization processes.

https://doi.org/10.64972/dea.2025.v4i2.1994d:43-56

The first module of the comparative experimental process is the modular feature extraction stage, as shown in Figure 2. This stage uses raw session data as input. Deep neural network embeddings and statistical metrics are combined into a feature vector. All competing classifiers undergo the training process simultaneously. In the experiment, the architecture flow ensures that each classifier's branch receives exactly the same preprocessed feature representation. The data flow and validation checkpoints are synchronized, so the observed result differences are valid.

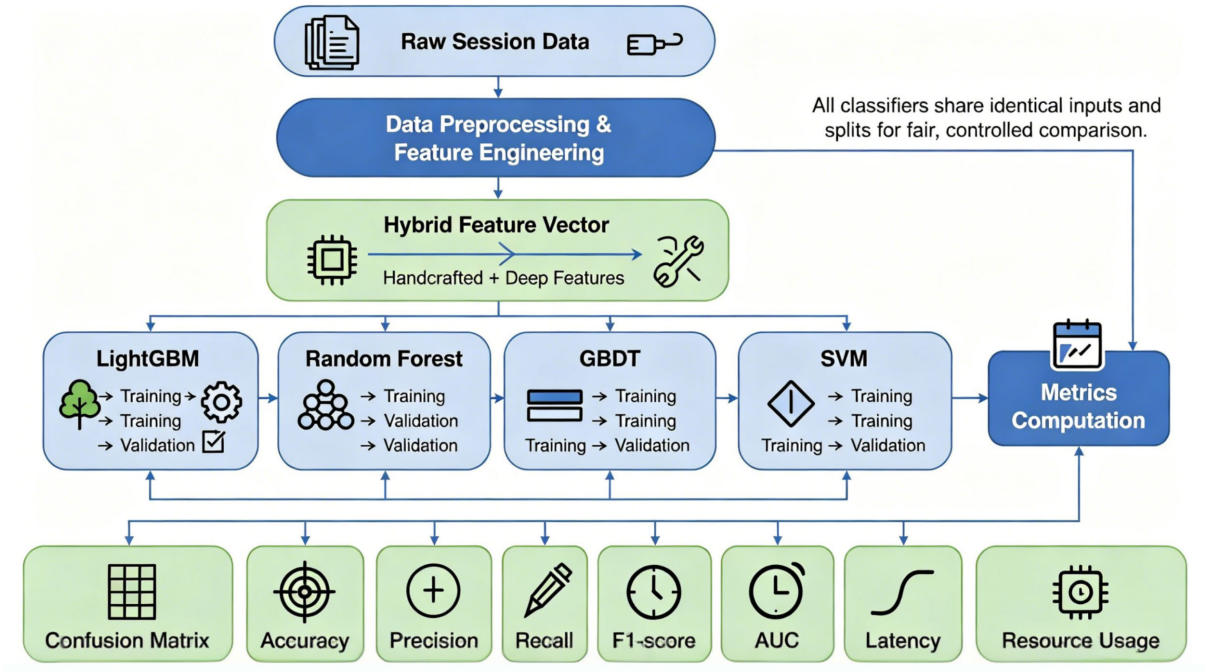

Figure 2. Classification Algorithm Flowchart

True positives (TP), false positives (FP), and false negatives (FN) are the confusion matrix used to calculate classification metrics for each evaluation stream. Based on the above content, the important comparison metrics have been determined. The recall rate and accuracy are as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Eq.(13)}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \qquad \text{Eq.(14)}$$

and the harmonic mean-the F1-score-is

$$\text{F1-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \qquad \text{Eq.(15)}$$

The area under the ROC curve can represent the general discriminative ability of all methods,

$$\text{AUC} = \int_0^1 \text{TPR}(f) d\text{FPR}(f) \qquad \text{Eq.(16)}$$

where $\text{TPR}(f)$ and $\text{FPR}(f)$ track true and false positive rates as a function of classification threshold. Each classifier branch consistently records various fine-grained statistics, including training time, memory usage, and inference latency. The results are normalized based on the size of the dataset to provide a system-level overview.

The above experimental results indicate that the empirical landscape of LightGBM is more suitable. Due to its advanced optimization, feature binning, and leaf growth mechanism, it outperforms other methods in terms of accuracy and recall for minority classes. In various traffic scenarios, it outperforms other methods in terms of accuracy and recall for minority classes under class imbalance conditions. In real-world high-traffic, adversarial, and encrypted network environments, the entire ship will exhibit better performance.

https://doi.org/10.64972/dea.2025.v4i2.1994d:43-56

Evaluation and Discussion

Dataset and Experimental Setup

Through a comprehensive analysis of backbone-level capture datasets from multiple urban areas. The high accuracy of the labels was manually verified for over 1.2 million sessions using DPI real data, port-based heuristic methods, and active probing.

The distribution of these session categories is shown in Figure 3(a). Due to its maximum, web browsing traffic accounts for approximately 36.7% of the total traffic. Although streaming and file transfer traffic is relatively low, it still accounts for a portion of the total traffic, approximately 19.4% and 14.2%, respectively. The proportion of system updates and VoIP categories in the total is less than 4%. The aforementioned issues indicate that to address the bias and underrepresentation of minority classes, it is necessary to improve the classification model and carefully stratify the experimental data.

Figure 3(b) shows the box plots of packet lengths for each category. For example, file transfers and streaming sessions have significantly larger interquartile ranges and higher upper whiskers, indicating a tendency for data blocks to have greater variability and burstiness. Due to the relatively uniform distribution of flows associated with social media and web browsing, they are more predictable and application-driven in a batch mode. Due to the differences in packet length distribution affecting feature engineering choices, a hybrid statistical-deep learning descriptor is needed.

According to kernel density estimation, Figure 3(c) shows that the time profile has different packet interval distributions. Streaming media and VoIP sessions show sharp, dense registration clusters below 60 milliseconds, indicating real-time and low-latency characteristics. Long-tail, multimodal density exhibits bursty and sporadic transmission patterns, characterizing file transfer and network categories. For creating a time-aware classifier capable of handling various network conditions, the aforementioned classification is necessary.

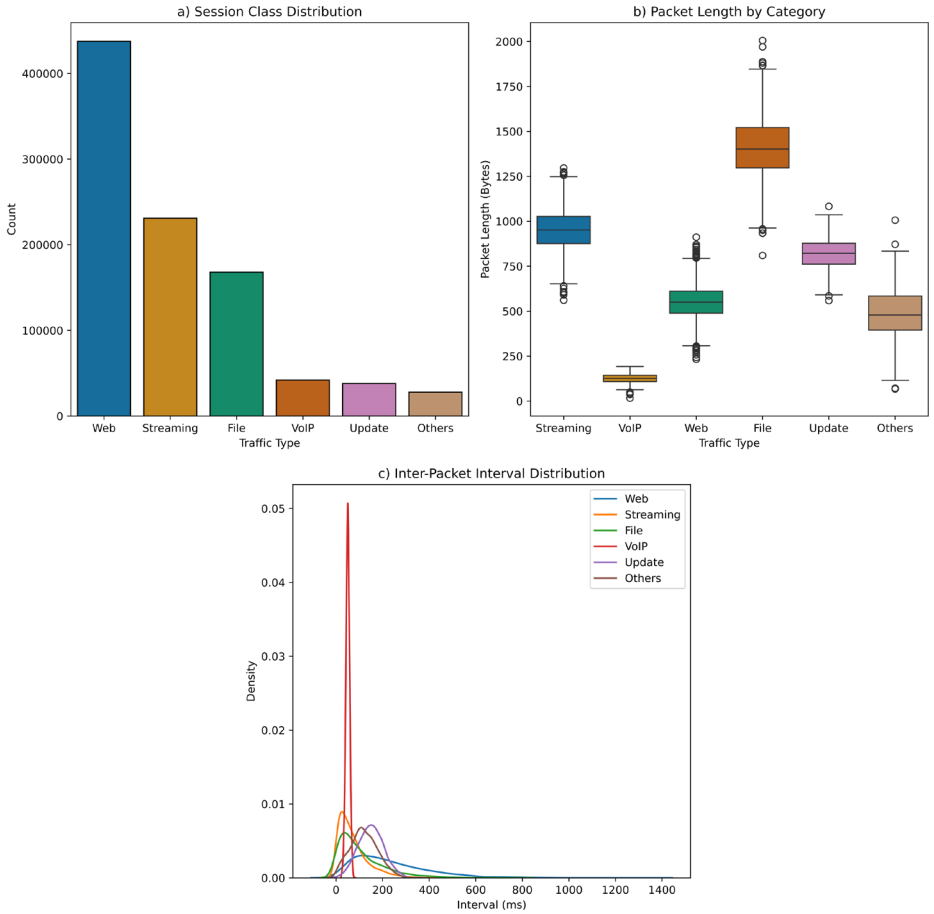

Figure 3. Dataset analysis and traffic diversity: (a) Session class distribution; (b) Packet length distribution; (c) Inter-packet interval density.

https://doi.org/10.64972/dea.2025.v4i2.1994d:43-56

All experiments were conducted on a platform equipped with 512GB of memory, four NVIDIA A100 GPUs, Ubuntu 22.04, Python 3.10, LightGBM 3.3.5, and controlled CUDA/driver versions on a 40-core Intel Xeon Gold 6248 to ensure the reproducibility of the results. After completing all preprocessing and outlier removal steps, a clean benchmark was obtained, with less than 0.3% of sessions excluded due to label ambiguity or data corruption.

Performance Analysis and Visualization

Evaluate the accuracy and robustness of the classifier, as well as its effectiveness in real encrypted traffic. At each stage of the analysis, various results were carefully examined to ensure that LightGBM demonstrated better specific details in overall accuracy, class-specific recall, and resource usage compared to the previous baseline models. This method can be used to thoroughly and meticulously demonstrate the reasons and areas where the new approach performs better. Naturally connects to a more quantitative analysis of the differences in performance and error characteristics.

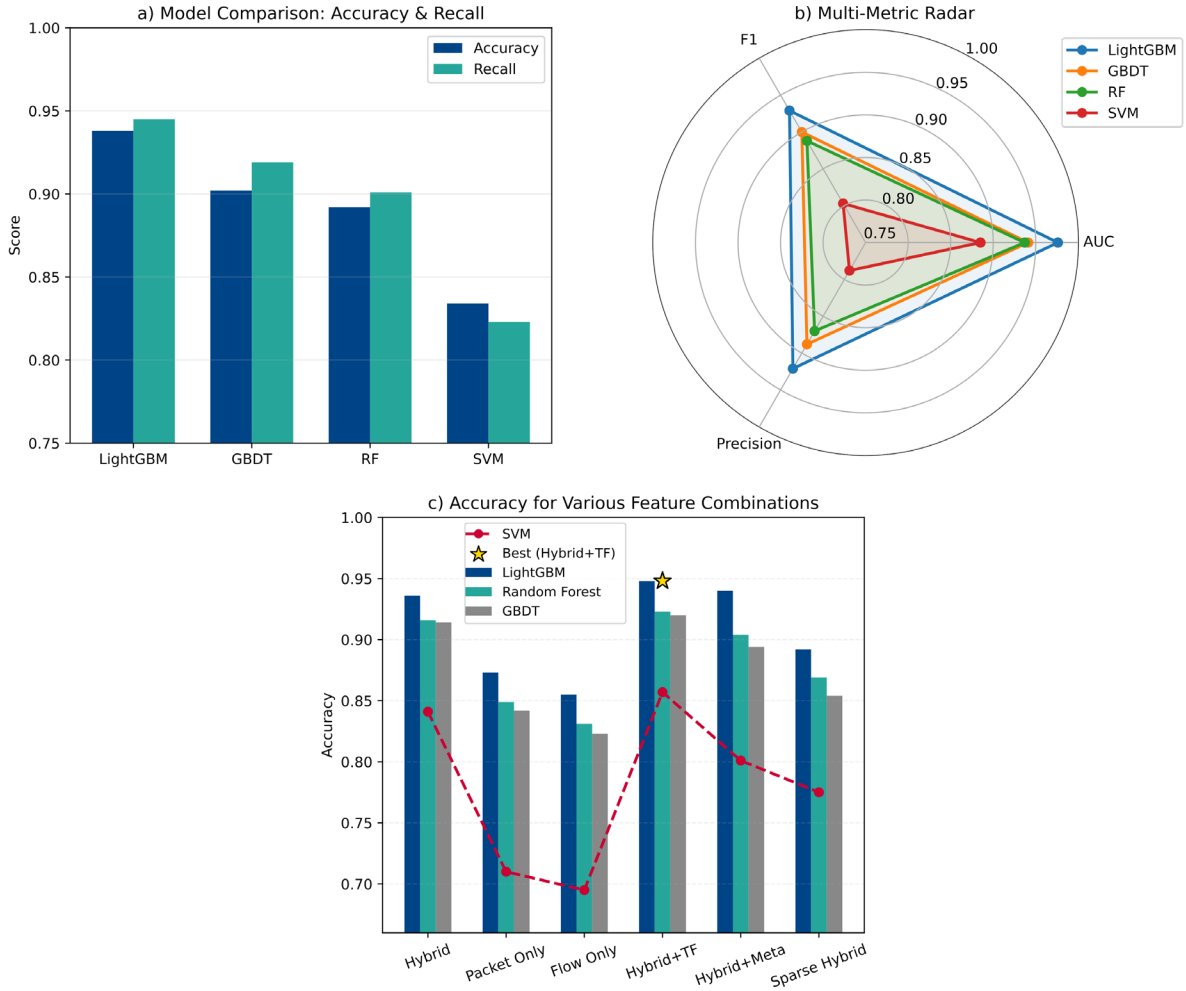

Figure 4. Model comparison results: (a) Classifier accuracy and recall; (b) multi-metric radar analysis; (c) Feature combination accuracy.

Figure 4(a) shows the overall accuracy and recall of all the classifiers considered. LightGBM outperforms traditional GBDT, random forest, and SVM on the test set, achieving an accuracy of 93.8% and a recall of 94.5%. The majority class can use the random forest baseline, but its accuracy is only 89.2% and its recall rate is only 90.1%; the shortcomings of SVM for the minority class are even more severe, with a recall rate of only 82.3%. Separate evidence indicates that the LightGBM model is more suitable for real-time backbone service applications, capable of handling fine-grained flow patterns and class imbalance.

https://doi.org/10.64972/dea.2025.v4i2.1994d:43-56

Figure 4(b) shows the AUC, F1-score, and accuracy of each model in the radar chart. LightGBM is convex, with a large coverage area, an average AUC of 0.976, and a macro F1 of 0.929. LightGBM is more stable under different traffic conditions and rare categories, with less bias. Although Random Forest and GBDT still perform well in high-traffic categories, the radar chart shows a significant decline in accuracy and F1, which is crucial for encryption and volatility applications. Kernel-based discriminators perform poorly when faced with high-entropy and obfuscated modern encrypted streams. This indicates that SVM performs poorly in many areas.

Figure 4(c) shows the impact of rich feature integration. The figure shows the accuracy of four common classifiers on six sets of high-level features: hybrid fusion, packet only, stream only, hybrid + time-frequency descriptors (Hybrid+TF), hybrid + meta-features (Hybrid+Meta), and sparse hybrid configuration. The results indicate that feature enhancement benefits all classifiers. For example, LightGBM achieved the highest accuracy of 94.8% under the Hybrid+TF combination; Random Forest and GBDT also showed similar trends. Due to its lower feature representation capability, SVM performed poorly. In order to create a reliable and accurate encrypted traffic identification system, context-aware multimodal fusion should be adopted.

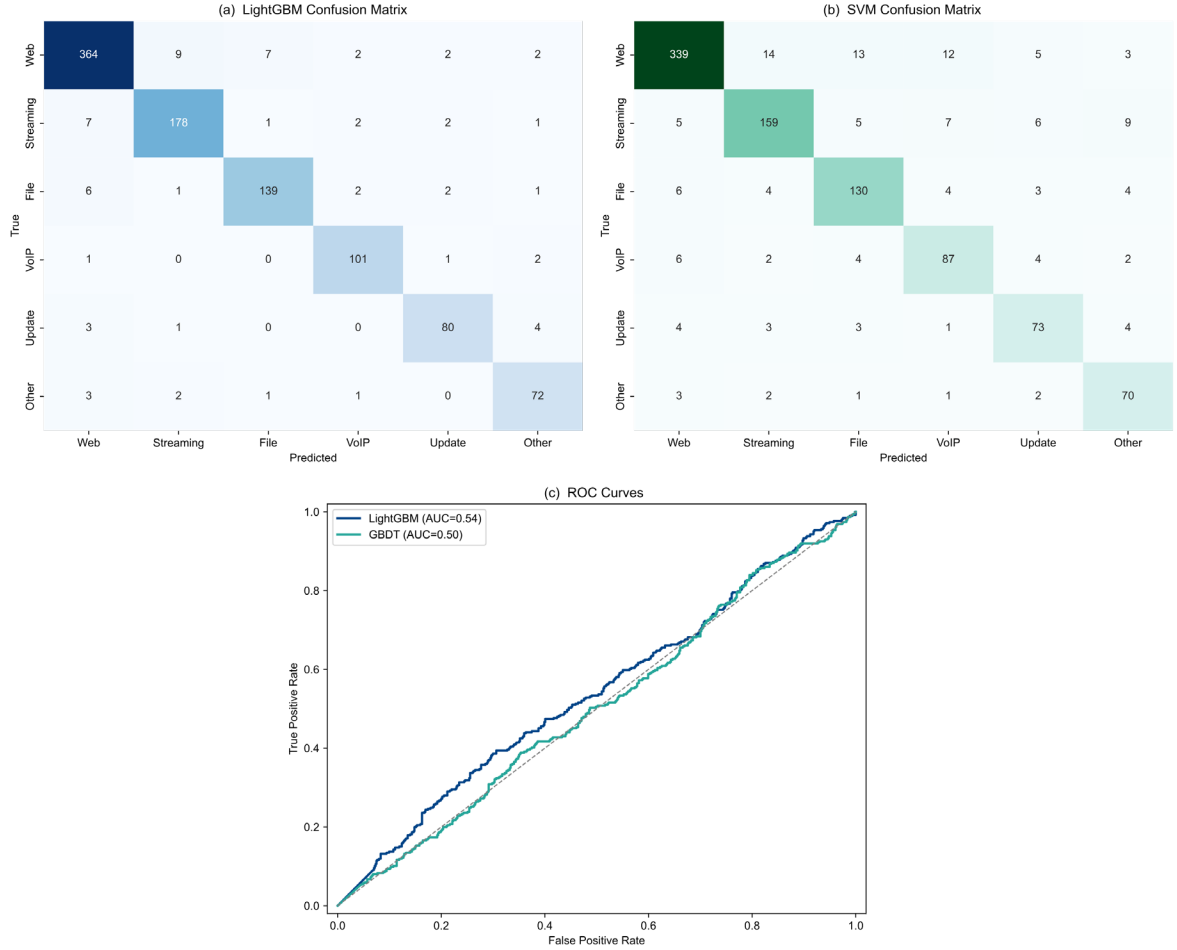

Figure 5. Classification error and ROC analysis: (a) Confusion matrix for LightGBM; (b) confusion matrix for SVM; (c) ROC curves for LightGBM and GBDT.

Figure 5 shows the fine-grained diagnostic breakdown. Figure 5(a) is the confusion matrix of LightGBM, with relatively small off-diagonal errors. For example, the proportion of streaming misclassified as file transfer is less than 4%, and most VoIP errors are misclassified as cloud synchronization rather than high traffic categories. The attribution risk has been effectively reduced. Figure 5(b) is the confusion matrix of the SVM. It shows that the cross-confusion rate between categories is relatively high, with the most notable being the misclassification of file transfer traffic exceeding 12% as web browsing. The decision boundaries and categories between different datasets in LightGBM are very clear.

https://doi.org/10.64972/dea.2025.v4i2.1994d:43-56

Figure 5(c) shows the ROC curves of LightGBM and GBDT. The LightGBM curve shows a relatively steep ascent, with the true positive rate exceeding 0.95 when the false positive rate is 0.04 for the main category. Indicates that the model is well-calibrated and the threshold is robust, as its area under the curve (AUC) is relatively large and stable at various levels. Initially, GBDT is relatively weak; at higher levels, it shows an increase in certain false positive rates.

Figure 6 is an extension of the previous analysis of feature ablation and efficiency/resource factors conducted across multiple dimensions. Figure 6(a) shows the comparison between the five feature sets of the discussed model. The grouped bar chart shows the recall, precision, and F1-score at different levels. Dotted lines were also added to show the changes in these metrics and their stability. The results indicate that as the feature subsets are gradually removed, such as traffic statistics or entropy, the overall performance will significantly decline. For example, the accuracy drops to 4.3%, and the F1-score and recall are also very sensitive. High-performance encrypted traffic identification requires both statistical and temporal features. By using visualization, it becomes easier to compare the contributions of different feature modalities to the model's generalization performance under adversarial attacks.

Figure 6(b) shows a comparison of the training time, inference latency, and memory usage of all five major classifiers. Inference latency (ms/session), memory consumption, and training time are represented by bar charts. The comprehensive chart illustrates which models can more effectively meet the requirements of speed, scale, and resource constraints, and provides support for engineering choices: LightGBM has lower inference latency (1.3ms) and uses less memory (1.9GB), while SVM is too slow and uses too much memory. XGBoost is very fast during inference but uses more memory. Multidimensional results indicate that the proposed method is still more suitable for efficient, highly reliable real-time large-scale encrypted network analysis.

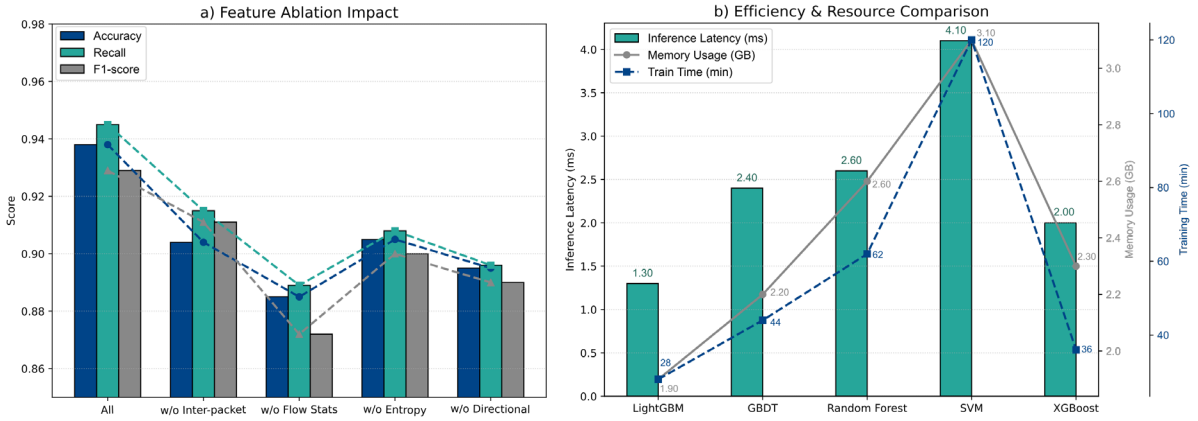

Figure 6. Feature and efficiency analysis: (a) Performance under feature ablation; (b) Efficiency and resource comparison for mainstream classifiers.

According to data and analysis, the LightGBM structure can maintain high detection rate performance in both majority and minority traffic. More scalable than traditional ensemble and kernel methods, and not affected by class confusion in adversarial traffic mixtures. Due to its sensitivity to feature space ablation and class imbalance, it demonstrates good generalization ability and provides reliable solutions for various forms and variations in encrypted network security.

Discussion on Model Interpretability

In order to ensure the credibility of machine learning applications in encrypted traffic analysis, the model must be interpretable [26]. In order to ensure the reliable operation of critical infrastructure, system auditors, security operators, and engineers must carefully examine feature attribution and instance-level decision logic beyond the headline statistics [27].

As shown in Figure 7, the following is a complete, multi-dimensional interpretability analysis of the proposed classification framework [28]. Figure 7(a) shows the overall importance of SHAP features. The entropy of the first payload, the count of upstream packets, and the variance of arrival times are the main factors in network traffic classification [29]. These eight top-ranked features together constitute a significant portion of the total

https://doi.org/10.64972/dea.2025.v4i2.1994d:43-56

model gain. High values are closely related to research on protocol determinism and flow structure in streaming and VoIP applications [30].

Figure 7(b) depicts several typical characteristics of the SHAP value distribution. In this case, the distribution and direction of the SHAP values for each feature indicate that changes in individual input features consistently affect the model's output [31]. For example, sessions with abnormally fixed frequency or low-entropy patterns produced positive SHAP values. This situation directly increases the likelihood of predicting structured flows, such as file transfers or automated backup service usage. Due to the bursty characteristics exhibiting a wide SHAP distribution, there is a high degree of distinction between highly dynamic and relatively stable traffic categories [32].

Figure 7(c) shows the LIME local explanation for a representative session sample. It includes eight different features, including structural indicators and time-related ones [33]. Visualize the contributions of positive and negative features, using different colors to distinguish between statistical data, time series data, and harmful factors. The annotation values indicate that the average arrival time and entropy have a strong positive impact, while burstiness and traffic duration have a smaller negative or neutral impact. These specific reasons provide useful guidance for practitioners and reveal how the model adjusts its decision boundaries based on session changes [34].

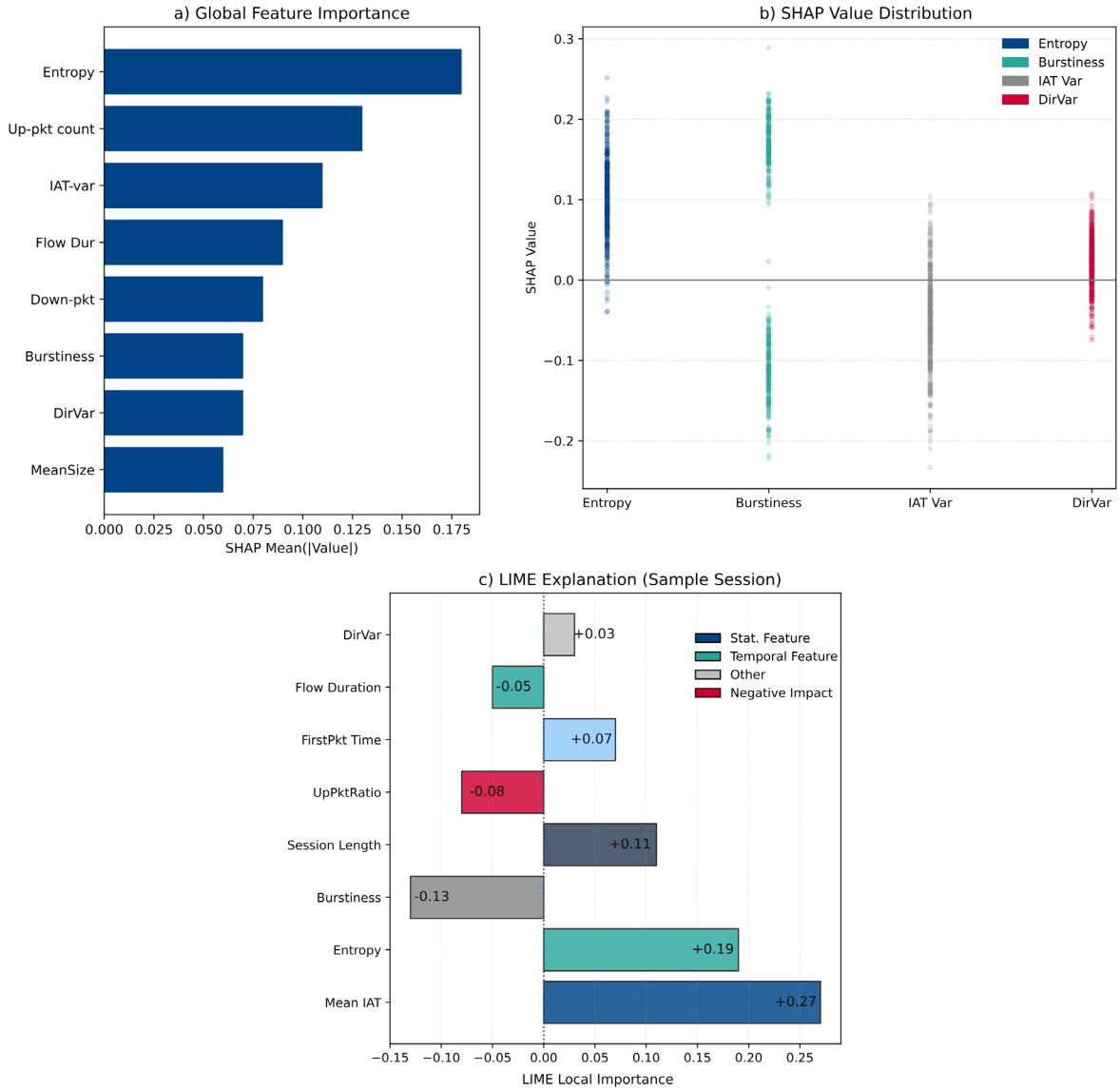

Figure 7. Model interpretability: (a) Global SHAP feature importance; (b) SHAP value distribution; (c) LIME local explanation.

Explainable visualization provides strong support for the engineering and operational safety of systems, helping network analysts understand the reasons behind classification results, quickly identify abnormal traffic behavior, discover new attack patterns, and ensure model stability in the face of adversarial attacks or protocol drift [35]. Global SHAP analysis and local LIME explanations are used to link model transparency with traceable sample-level reasoning, thereby building trust in the scientific community and the public. The diversity and quality of the selected feature set will limit the scope of interpretability. In terms of the practicality and transparency of next-generation encrypted traffic classifiers, further progress in adaptive and domain-oriented feature engineering will enhance this limitation.

Conclusion

This paper will build a comprehensive encrypted traffic classification framework based on the powerful features and flexibility of LightGBM. By carefully studying a large and diverse set of network datasets, a meticulously designed hybrid feature set achieves significant and measurable improvements over traditional kernel and ensemble methods. Empirical results show that it consistently achieves high accuracy and recall rates, and is very effective for various types of conversations and class imbalances. The results were supported by a rigorous experimental plan. The LightGBM model performs better in identifying long-tail and minority traffic categories while reducing computational costs and inference latency. The typical feature of the benefits is the integration of packet-level, flow-level, and entropy-driven attributes, which enables the classifier to extract a large amount of structural information from encrypted streams.

This study has discovered an interpretable classification pipeline. In order to translate abstract machine learning into engineering and operational knowledge, SHAP and LIME analyzes have been used to determine the specific reasons for classification results. More professionals will be responsible for the security of high-risk areas and will be able to quickly identify and handle threats in complex, hostile networks. Explainability mechanisms can reduce the gap between experimental accuracy and deployment stability, as the model's predictions are based on intuitive, reasonable, and protocol-related features.

According to the above results, the following three issues must be addressed. Developing true real-time detection is a common issue, which requires continuous optimization of adaptive learning and low-latency stream processing. The robustness of modified and continuously evolving encryption protocols will be enhanced by using deeper levels of temporal and volumetric information, as well as context-aware information. Due to the continuous improvement of adversarial modification strategies for network traffic, research in this field has recently shifted toward defense, robust training methods, and dynamic model adjustment. This research sets a new standard for the maximum encryption capability of current encrypted traffic classification and provides strong support for studying network security analysis with operational flexibility, interpretability, and future reliability.

Author Contributions

Sławomir Feliks Jarosz contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Shen, M., Ye, K., Liu, X., Zhu, L., Kang, J., Yu, S., ... & Xu, K. (2022). Machine learning-powered encrypted network traffic analysis: A comprehensive survey. IEEE Communications Surveys & Tutorials, 25(1), 791-824. https://doi.org/10.1109/COMST.2022.3208196

[2] Alwhbi, I. A., Zou, C. C., & Alharbi, R. N. (2024). Encrypted network traffic analysis and classification utilizing machine learning. Sensors, 24(11), 3509. https://doi.org/10.3390/s24113509

[3] Khani, P., Moeinaddini, E., Abnavi, N. D., & Shahraki, A. (2024). Explainable artificial intelligence for feature selection in network traffic classification: A comparative study. Transactions on Emerging Telecommunications Technologies, 35(4), e4970. https://doi.org/10.1002/ett.4970

[4] Kondaiah, C., Pais, A. R., & Rao, R. S. (2024). Enhanced malicious traffic detection in encrypted communication using tls features and a multi-class classifier ensemble. Journal of Network and Systems Management, 32(4), 76. https://doi.org/10.1007/s10922-024-09847-3

[5] Zhang, H., Gou, G., Xiong, G., Liu, C., Tan, Y., & Ye, K. (2021, August). Multi-granularity mobile encrypted traffic classification based on fusion features. In International Conference on Science of Cyber Security (pp. 154-170). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-89137-4_11

[6] Zou, X., Chung, E., Zhou, Y., Long, M., & Lam, W. H. (2024). A feature extraction and deep learning approach for network traffic volume prediction considering detector reliability. Computer-Aided Civil and Infrastructure Engineering, 39(1), 102-119. https://doi.org/10.1111/mice.13062

[7] Hassan, F., Yu, J., Syed, Z., Magsi, A., & Ahmed, N. (2023). Developing transparent IDS for VANETs using LIME and SHAP: An empirical study. Computers, Materials, & Continua, 77(3), 3185. https://doi.org/10.32604/cmc.2023.044650

[8] Aouedi, O., Piamrat, K., & Parrein, B. (2022). Ensemble-based deep learning model for network traffic classification. IEEE Transactions on Network and Service Management, 19(4), 4124-4135. https://doi.org/10.1109/TNSM.2022.3193748

[9] Liu, Y., Wang, X., Qu, B., & Zhao, F. (2024). ATVITSC: A novel encrypted traffic classification method based on deep learning. IEEE transactions on information forensics and security, 19, 9374-9389. https://doi.org/10.1109/TIFS.2024.3433446

[10] Kenyon, A., Deka, L., & Elizondo, D. (2024). Characterising payload entropy in packet flows—baseline entropy analysis for network anomaly detection. Future Internet, 16(12), 470. https://doi.org/10.3390/fi16120470

[11] Çelebi, M., Özbilen, A., & Yavanoğlu, U. (2023). A comprehensive survey on deep packet inspection for advanced network traffic analysis: issues and challenges. Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi, 12(1), 1-29. https://doi.org/10.28948/ngumuh.1184020

[12] Lasfar, R., & Tóth, G. (2024). The difference of model robustness assessment using cross-validation and bootstrap methods. Journal of Chemometrics, 38(6), e3530. https://doi.org/10.1002/cem.3530

[13] Ullah, W., Ullah, A., Hussain, T., Muhammad, K., Heidari, A. A., Del Ser, J., ... & De Albuquerque, V. H. C. (2022). Artificial Intelligence of Things-assisted two-stream neural network for anomaly detection in surveillance Big Video Data. Future Generation Computer Systems, 129, 286-297. https://doi.org/10.1016/j.future.2021.10.033

[14] Wang, X., Yuan, Q., Wang, Y., Gou, G., Gu, C., Yu, G., & Xiong, G. (2024). Combine intra-and inter-flow: A multimodal encrypted traffic classification model driven by diverse features. Computer Networks, 245, 110403. https://doi.org/10.1016/j.comnet.2024.110403

[15] Alserhani, F. (2024). Analysis of encrypted network traffic for enhancing cyber-security in dynamic environments. Applied Artificial Intelligence, 38(1), 2381882. https://doi.org/10.1080/08839514.2024.2381882

[16] Seydali, M., Khunjush, F., & Dogani, J. (2024). Streaming traffic classification: a hybrid deep learning and big data approach. Cluster Computing, 27(4), 5165-5193. https://doi.org/10.1007/s10586-023-04234-0

[17] Thakkar, A., & Lohiya, R. (2023). Fusion of statistical importance for feature selection in deep neural network-based intrusion detection system. Information Fusion, 90, 353-363. https://doi.org/10.1016/j.inffus.2022.09.026

[18] Islam, M. T., Syfullah, M. K., Rashed, M. G., & Das, D. (2024). Bridging the gap: advancing the transparency and trustworthiness of network intrusion detection with explainable AI. International Journal of Machine Learning and Cybernetics, 15(11), 5337-5360. https://doi.org/10.1007/s13042-024-02242-z

[19] Fu, Z., Liu, M., Qin, Y., Zhang, J., Zou, Y., Yin, Q., ... & Duan, H. (2022, October). Encrypted malware traffic detection via graph-based network analysis. In Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses (pp. 495-509). https://doi.org/10.1145/3545948.3545983

[20] Almahdi, A., Al Mamlook, R. E., Bandara, N., Almuflih, A. S., Nasayreh, A., Gharaibeh, H., ... & Jamal, A. (2023). Boosting ensemble learning for freeway crash classification under varying traffic conditions: A hyperparameter optimization approach. Sustainability, 15(22), 15896. https://doi.org/10.3390/su152215896

[21] Wang, C., Fu, Y., Xu, L., Ye, X., Shi, J., & Wang, Y. (2024, November). A Multi-faceted Analysis and Comprehensive Research of Trustworthy Path Routing Technologies. In International Conference on Information Processing and Network Provisioning (pp. 451-462). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-95-1331-4_38

[22] Gaspar, D., Silva, P., & Silva, C. (2024). Explainable AI for intrusion detection systems: LIME and SHAP applicability on multi-layer perceptron. IEEE Access, 12, 30164-30175. https://doi.org/10.1109/ACCESS.2024.3368377

[23] Chiang, C. (2024). Drift-aware adaptive classification for imbalanced data via dynamic class reweighting and structural regularization. Transactions on Computational and Scientific Methods, 4(12). https://doi.org/10.5281/zenodo.18647079

[24] Elmaghraby, R. T., Aziem, N. M. A., Sobh, M. A., & Bahaa-Eldin, A. M. (2024). Encrypted network traffic classification based on machine learning. Ain Shams Engineering Journal, 15(2), 102361. https://doi.org/10.1016/j.asej.2023.102361

[25] Lotfollahi, M., Jafari Siavoshani, M., Shirali Hossein Zade, R., & Saberian, M. (2020). Deep packet: A novel approach for encrypted traffic classification using deep learning. Soft Computing, 24(3), 1999-2012. https://doi.org/10.1007/s00500-019-04030-2

[26] Polemi, N., Praça, I., Kioskli, K., & Bécue, A. (2024). Challenges and efforts in managing AI trustworthiness risks: a state of knowledge. Frontiers in big Data, 7, 1381163. https://doi.org/10.3389/fdata.2024.1381163

[27] Zhang, Z., Al Hamadi, H., Damiani, E., Yeun, C. Y., & Taher, F. (2022). Explainable artificial intelligence applications in cyber security: State-of-the-art in research. IEEe Access, 10, 93104-93139. https://doi.org/10.1109/ACCESS.2022.3204051

[28] Nair, R. (2023). Unraveling the Decision-making Process Interpretable Deep Learning IDS for Transportation Network Security. Journal of Cybersecurity & Information Management, 12(2). https://doi.org/10.54216/JCIM.120205

[29] AsSadhan, B., Bashaiwth, A., & Binsalleeh, H. (2024). Enhancing explanation of lstm-based ddos attack classification using shap with pattern dependency. IEEE Access, 12, 90707-90725. https://doi.org/10.1109/ACCESS.2024.3421299

[30] Peng, J., & Tang, S. (2020). Covert communication over VoIP streaming media with dynamic key distribution and authentication. IEEE Transactions on Industrial Electronics, 68(4), 3619-3628. https://doi.org/10.1109/TIE.2020.2979567

[31] Wang, H., Liang, Q., Hancock, J. T., & Khoshgoftaar, T. M. (2024). Feature selection strategies: a comparative analysis of SHAP-value and importance-based methods. Journal of Big Data, 11(1), 44. https://doi.org/10.1186/s40537-024-00905-w

[32] Leite, R., Amado, C., & Azeitona, M. (2024). Online burst detection in water distribution networks based on dynamic shape similarity measure. Expert Systems with Applications, 248, 123379. https://doi.org/10.1016/j.eswa.2024.123379

[33] Maloney, K. O., Buchanan, C., Jepsen, R. D., Krause, K. P., Cashman, M. J., Gressler, B. P., ... & Schmid, M. (2022). Explainable machine learning improves interpretability in the predictive modeling of biological stream conditions in the Chesapeake Bay Watershed, USA. Journal of Environmental Management, 322, 116068. https://doi.org/10.1016/j.jenvman.2022.116068

[34] Guan, L., & Yuan, X. (2024). Dynamic weighting and boundary-aware active domain adaptation for semantic segmentation in autonomous driving environment. IEEE Transactions on Intelligent Transportation Systems, 25(11), 18461-18471. https://doi.org/10.1109/TITS.2024.3431537

[35] Rahman, M. M., Soumik, M. S., Farids, M. S., Abdullah, C. A., Sutrudhar, B., Ali, M., & HOSSAIN, M. S. (2024). Explainable anomaly detection in encrypted network traffic using data analytics. Journal of Computer Science and Technology Studies, 6(1), 272-281. https://doi.org/10.32996/jcsts.2024.6.1.31