Computer Systems
Science and Engineering

Tech Science Press

Doi:10.32604/csse.2025.061285

ARTICLE

Classifying Network Flows through a Multi-Modal 1D CNN Approach Using Unified Traffic Representations

Ravi Veerabhadrappa* **and Poornima Athikatte Sampigerayappa**

Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumkur, 572103, India

*Corresponding Author: Ravi Veerabhadrappa. Email: ravi@sit.ac.in

Received: 21 November 2024; Accepted: 06 February 2025; Published: 19 March 2025

ABSTRACT: In recent years, the analysis of encrypted network traffic has gained momentum due to the widespread use of Transport Layer Security and Quick UDP Internet Connections protocols, which complicate and prolong the analysis process. Classification models face challenges in understanding and classifying unknown traffic because of issues related to interpret ability and the representation of traffic data. To tackle these complexities, multi-modal representation learning can be employed to extract meaningful features and represent them in a lower-dimensional latent space. Recently, auto-encoder-based multi-modal representation techniques have shown superior performance in representing network traffic. By combining the advantages of multi-modal representation with efficient classifiers, we can develop robust network traffic classifiers. In this paper, we propose a novel multi-modal encoder-decoder model to create unified representations of network traffic, paired with a robust 1D-CNN (one-dimensional convolution neural network) classifier for effective traffic classification. The proposed model utilizes the ISCX Virtual Private Network-non Virtual Private Network 2016 datasets to extract general multi-modal representations and to train both shallow and deep learning models, such as Random Forest and the 1D-CNN model, for traffic classification. We compare these learning approaches based on the multi-modal representations generated from the autoencoder and the early feature fusion technique. For the classification task, both the Random Forest and 1D-CNN models, when trained on multimodal representations, achieve over 90% accuracy on a highly imbalanced dataset.

KEYWORDS: Encrypted network traffic; multi-modal; random forest; 1D-CNN

1 Introduction

Encrypted Network Traffic Analysis (ENTA) is a critical tool used by system administrators to monitor, analyze, and interpret traffic patterns that emerge from network captures. By leveraging this analysis, organizations can improve their resource allocation strategies and optimize performance across various types of networks, particularly those that are heterogeneous in nature, meaning they consist of different types of devices and technologies. The ENTA process can be enhanced through automation, allowing for real-time data assessment and decision making. Advanced methodologies, including both shallow learning and deep learning models [1,2], can be employed to uncover insights from the data. Shallow learning models can quickly identify simple patterns, while deep learning models are capable of analyzing complex features and relationships within the traffic data.

Despite the technological advancements in this field, one major challenge remains: the limited availability of data and the absence of labeled feature sets for training these machine-learning models [3]. Labeled data is essential for supervised learning approaches, which require examples of both input data and the corresponding correct outputs. Due to the sensitive and often encrypted nature of network traffic, gathering such labeled datasets is difficult [4].

To overcome these obstacles, organizations often need to rely on professionals with deep domain expertise. These skilled individuals possess the knowledge required to discern meaningful patterns within encrypted traffic, even in the context of limited data. Their insights are invaluable for refining model training, ultimately leading to more effective analysis and improved security measures for managing network traffic. Network captures generally involve two main components: header information and payload data [5–7]. The payload specifically comprises layer-7 data, which is generated by various application programs and is facilitated by protocols such as Transport Layer Security (TLS) and Quick UDP Internet Connections (QUIC). To effectively analyze this layer-7 data, it is essential to utilize specialized Deep Packet Inspection (DPI) engines. These engines are designed to decrypt both the payload and the headers, allowing for the extraction of meaningful patterns that can provide insights into network behavior.

In recent developments, data augmentation techniques have gained traction in the field of network traffic analysis. These techniques enhance the volume and variety of data available for analysis, ultimately leading to improved model performance [8–10]. Additionally, new strategies have emerged to tackle issues such as class imbalance and feature bias, which can affect the accuracy of analysis in network traffic. DPI-based methods are particularly valuable because they can identify significant patterns from various headers and payloads within the captured data. However, the presence of encryption protocols presents a unique challenge: they hinder the ability to extract features from encrypted data effectively. This challenge underscores the necessity for specialized DPI techniques that can handle encrypted traffic and extract useful information, thereby enabling a more comprehensive analysis of network captures.

Machine learning [11,12] and deep learning models [13–15] are increasingly utilized for analyzing encrypted traffic, taking into account both the spatial and temporal characteristics of data patterns. Despite their potential, these advanced techniques often produce model-agnostic feature representations, which hinder the reproducibility and consistency of results obtained from various analyses. Deep learning models [6], in particular, have shown considerable success in automatically extracting features and performing classification tasks on complex datasets. However, the challenge of achieving model-agnostic data representations complicates the development of robust classifiers that can function effectively across different network representations.

To address these challenges related to feature representation and extraction, innovative deep learning structures such as auto-encoders [16–18] and transformer models [19] have emerged as effective solutions. These models are particularly adept at simplifying the complexity involved in downstream classification tasks. However, designing and implementing these models can be intricate and require significant processing power, especially when dealing with large-scale datasets like network traffic captures. Auto-encoders offer a compelling approach for creating scalable architectures aimed at reducing data dimensionality and extracting features. Compared to transformer models, auto-encoders have demonstrated a remarkable ability to identify and capture relevant features from raw data, effectively representing them within a latent space [20,21].

Moreover, the architectural framework of auto-encoders [22,23] and their corresponding decoders can be utilized to amalgamate various modalities of information into a cohesive latent space. To effectively integrate multiple modalities, several strategies for feature-level fusion can be employed, including early fusion, late fusion, and decision-based fusion techniques [24,25]. Considering these aspects, we propose a novel approach aimed at enhancing feature representation, with the goal of creating a unified and comprehensive view of the available data [26,27].

In this paper, we present key contributions to the development of our proposed architecture for analyzing captured network traffic:

1. To highlight the existing research on different model architectures, we summarize related works in Table 1, which presents an overview of models used in prior studies, including Auto-encoders, Convolutional neural networks, and Transformer based models.

2. We conducted experiments using the ISCX VPN and non-VPN 2016 dataset to evaluate our proposed method. We extracted flow information using the NFStream application to generate features from the captures.

3. We have developed a novel autoencoder architecture that creates a compact representation of flow-level metadata from network captures. Our model employs early fusion to integrate different types of information, such as entities and quantities, transformed into a unified representation, as shown in Table 2. Additionally, we developed a novel 1D-CNN-based model to train and classify these generic representations into various traffic classes.

4. Our comprehensive analysis of model performance indicates that the 1D-CNN can achieve a classification accuracy exceeding 90% when trained on these generic representations, as shown in Table 3.

The remainder of this paper is organized as follows: In Section 2, we discuss related work on generic representations. Section 3 provides a detailed explanation of our proposed method. In Section 4, we present experiments utilizing the ISCX VPN and non-VPN datasets to represent traffic classifications as generic representations.

2 Related Work

In this section, we review the literature related to representing network captures using various deep-learning techniques. The studies on network traffic classification are model-agnostic, meaning that the input data is transformed into model-agnostic representations and trained for multiple tasks. Our research focuses on exploring the application of deep learning models in creating generic representations of network traffic. While this is not an exhaustive overview, Table 1 summarizes our findings regarding deep learning models such as CNN's, auto-encoders, and NLP based techniques that have been used to represent network traffic. Our survey demonstrates the extensive use of these models and highlights their limitations. Finally, we emphasize the need to combine feature fusion and multi-modal representation learning to develop improved input representation models.

Höchst et al. [1] address the challenges in processing network captures and propose a clustering labeling method for classifying traffic, achieving 80% accuracy on a public datasets. They also emphasize the need for an auto-encoder-based architecture for network traffic analysis. Bengio et al. [2] review various deep learning models, including DBN, CNN, and RNN, used for data representation. Their study highlights challenges such as multi-modality data handling, temporal misalignment in fusion operations, and over fitting in model training. Wang et al. [4] convert encrypted traffic into images for application type classification using CNN. His end-to-end model, based on the ISCX VPN-nonVPN datasets, successfully captures both local and global features.

Lotfollahi et al. [5] combine a stacked auto-encoder and CNN to extract features from the same datasets. His pre-processing efforts led to 98% accuracy in application identification and 94% in traffic characterization, underscoring the need to handle features from encrypted payloads and headers effectively. Ring et al. [7] highlight the significance of IP address embedding using the word2vec model to enhance classification models. They applied the IP2Vec model for embedding IP addresses and used clustering to classify them as normal or malicious, achieving an accuracy of 86% with the CTU-13 datasets. Their work also addresses challenges in feature extraction from flow metadata and the alignment between packet header context and vector representations.

Zhao et al. [13] introduced Yet Another Traffic Classifier (YaTC), a novel traffic classification model that employs a masked auto encoder (MAE)-based transformer with multi-level flow representations. YaTC integrates packet-level and flow-level attention mechanisms for efficient feature extraction, utilizing MAE for pre-training on unlabeled data and fine-tuning on labeled data. The model, evaluated on five real-world encrypted traffic datasets (e.g., ISCXTor2016, USTC, CICIoT2022, and Cross Platform), achieves high accuracy rates of 98.07% on one datasets, 98.04% on ISCXTor 2016 datasets, 99.72% on USTC, and 96.58% on CICIoT2022, demonstrating superior performance compared to state-of-the-art methods.

Li et al. [15] introduce the L2-BiTCN-CNN model, which combines bidirectional temporal convolution networks (TCN) and convolution neural networks (CNN) for multi-class network traffic classification. By fusing spatial-temporal features, the model effectively identifies various internet applications. It has been evaluated on the USTC-TFC2016 datasets (containing malware and benign traffic) and the ISCX VPN-nonVPN2016 datasets (containing encrypted and non-encrypted traffic), demonstrating high accuracy. Future work will focus on improving classification accuracy for specific traffic types and exploring knowledge distillation and edge computing for deployment efficiency. The model's ability to handle complex patterns and differentiate between similar traffic patterns is particularly noteworthy.

Lin et al. [20] introduced ET-BERT, a novel approach for classifying encrypted network traffic that uses a feature engineering pipeline and BERT based transformer models. This method addresses challenges in handling multi-modal network traffic data and selects optimal features from packet statistics. Trained on the CSTNET datasets, the model achieves 99% accuracy on the ISCX VPN and USTC-FTC datasets, and 97% accuracy on the CSNET-TLS-1.3 datasets. ET-BERT enhances classification performance by creating contextualized representations of datagram, thereby improving accuracy despite encryption.

Authors Barua et al. [23] and Ramachandram et al. [28] reviewed trends in representation learning and fusion techniques, emphasizing the difficulty of selecting appropriate methods for various applications and integrating modality-specific information. Gonzalez et al. [29] introduced the Net2Vec model, which generates embedding for input data using machine learning and word2vec. This process allows for compact feature representation, suitable for tasks like traffic classification and user profiling. Vu et al. [30] tackled class imbalance in traffic datasets, proposing an auxiliary hybrid model combining machine learning methods. They reported baseline models such as SVM+ACGAN with an impressive 99% classification accuracy, highlighting the need for effective feature extraction and data augmentation.

Aceto et al. [31] discussed challenges in applying GRU and CNN models to multi-modal datasets and detailed pre-processing methods for various input types using LSTM and CNN. They achieved over 90% accuracy in Android-based traffic classification and underscored the importance of informative inputs and the potential biases in machine learning models that require frequent retraining. They also called for fine-grained classifiers with advanced hybrid architectures for better information pre-processing. Shahbaz Rezaei et al. [32] proposed a CNN-based model for extracting features from network traffic's time-series data, utilizing both header and payload information for a multi-label classification problem. Their work examined combining CNN's with sequence models like LSTM and RNN, highlighting challenges in applying deep learning to encrypted traffic.

Cohen et al. [33] extracted flows from traffic using a windowing technique and transformed sequences into embedding with the word2vec model. These embedding trained a clustering model to differentiate between dark-net traffic and normal traffic, emphasizing the importance of feature engineering for metadata, including IP addresses and ports. Barut et al. [34] discussed challenges in feature extraction from large datasets, advocating for generic feature representations and pre-trained models to improve network flow classification. Holland et al. [35] introduced nPrint, a tool that generates standardized packet representations to improve automated machine learning (AutoML) in traffic analysis. nPrintML automates feature extraction and model tuning, proving effective in device fingerprinting and OS detection while addressing challenges in pre-processing multi-modal pcap files.

Shahraki et al. [36] discussed deep learning models for network traffic analysis and monitoring operations. Their proposed method combines a set of Convolution Neural Network (CNN) models into an ensemble of classifiers. The outputs of these models are then merged to generate the final prediction. Performance evaluation results indicate that their methodology achieves an average accuracy rate of 98% for classifying traffic (e.g., FTP-DATA, MAIL, etc.) using the Cambridge Internet traffic datasets.

Kallitsis et al.'s work [37] extends network traffic analysis to the DarkNet by employing autoencoders and decoders to represent packet statistics, focusing on bot and attack detection. Using one-hot encoding for port addresses and a semi-supervised approach for clustering embedding, the model achieves 90% accuracy in detecting temporal changes in DarkNet traffic. It introduces methods for identifying anomalies and variations in scanning behaviors to enhance network security monitoring. Feng et al. [38] proposed a CNN-based approach for classifying VPN traffic, addressing the limitations of traditional methods when dealing with encrypted network traffic. By combining payload-based techniques with a modified AlexNet structure, their model achieves an accuracy of 89.97% using the VPN-nonVPN datasets. It outperforms other network structures like LeNet, VGG, and ResNet in certain categories, demonstrating the effectiveness of deep learning in accurately classifying encrypted traffic.

Houidi et al.'s work [39] explored multi-modal representation learning for network data, proposing a bimodal approach that combines language models with traditional features. By integrating entity-based and quantity based representations, this approach enhances classification tasks, as shown in use cases like clickstream identification and terminal movement prediction. The study emphasizes the need for systematic representation learning and suggests incorporating additional modalities, such as time-evolving graphs, and using graph neural networks for more effective modeling, highlighting the complexity of network data and the promise of advanced techniques in improving machine learning outcomes. Gioacchini et al. [40] present i-DarkVec, a novel method for analyzing DarkNet traffic that utilizes NLP-based embedding, specifically employing word2vec, to represent network traffic efficiently. Trained on a neural network model, i-DarkVec reaches 97% accuracy on DarkNet traffic datasets. This approach improves analysis by allowing for dynamic and continuous updates of the embedding as new traffic data arrives, thereby enhancing adaptability to evolving traffic patterns.

Yang et al. [41] introduced the Dual Mode Hybrid Neural Network (DM-HNN) for network traffic classification. This method combines packet length and byte representations using Gated Recurrent Units (GRU) and stacked auto-encoder techniques. When compared to baseline models and the DISTILLER multi modal deep learning model, DM-HNN achieves an impressive 99% accuracy on ISCX datasets. By integrating both time-domain and frequency-domain features through advanced hybrid neural network architectures, this approach significantly enhances the effectiveness of traffic classification.

Gioacchini et al. [42] investigate the use of Temporal Graph Neural Networks (TGNNs) for analyzing DarkNet traffic. They propose a method for creating embeddings from both statistical and sequence fields of packets. These embedding facilitate classification tasks using Graph Neural Network (GNN)-based models, with kNearest Neighbors (kNN) clustering used for data labeling. The approach achieves an F1 Score of 80%, demonstrating improved performance in understanding and classifying DarkNet traffic patterns through enhanced temporal information integration. Li et al. [43] present a decision-level multi-modal fusion technique for managing network traffic data. In this method, embedding from various modalities are combined using a stacked auto-encoder model. The pre-processing phase includes a feature engineering pipeline to extract both spatial and temporal features. This technique reaches 93% accuracy on a real-time mobile device datasets of application traffic, showcasing its effectiveness in classifying encrypted traffic. By integrating multiple decision sources through late fusion, this approach enhances classification accuracy.

Pang et al. [44] present the Multi-Modal Classification Method (MTCM) for context-aware network traffic classification. This approach integrates graph neural networks and BERT for feature extraction and fusion. MTCM enhances traditional deep learning classifiers by incorporating contextual information from communication sessions and text semantics. It achieves high accuracy rates of 92.2% for application traffic, 98.7% for malicious traffic, and 98.7% for encrypted traffic. The method demonstrates robustness across various datasets, outperforming existing techniques and improving classification performance through adaptive context-aware feature extraction.

Gioacchini et al. [45] introduce the Multi-Modal Auto-Encoder (MAE) architecture for network traffic analysis, offering a deep learning approach that minimizes the need for extensive feature engineering. The MAE integrates different types of input data—such as quantities and entities—into a compact representation, trained in a self-supervised manner. This method produces embedding that surpass traditional concatenation methods and are highly effective in several supervised traffic classification tasks. Evaluated on three datasets: MIRAGE for mobile app traffic, DARKNET for dark net traffic, and ISCX for traffic flows, the MAE shows improved performance and requires fewer trainable parameters compared to conventional learning methods. The research also identifies challenges, such as parameter tuning, the scalability of One-Hot Encoding, and performance issues with small datasets, suggesting areas for further research and optimization.

Cui et al. [46] introduced a novel multimodal hybrid parallel network intrusion detection model (MHPN) to enhance the accuracy and robustness of network intrusion detection systems. The MHPN model utilizes statistical network traffic information and raw traffic payload data. It employs convolutional neural networks (CNNs), long short-term memory (LSTM) networks for feature extraction, and a CosMargin classifier that improves classification in imbalanced datasets. Experiments conducted on the ISCX-IDS2012 and CIC-IDS-2017 datasets show that the MHPN model outperforms single-modal models, achieving an impressive average accuracy of 99.98%. The study also examines the model's components and compares its efficiency to other existing methods.

Wang et al. [47] present MeDF, a novel multi modal encrypted traffic classification model that integrates intraflow and inter-flow features to enhance classification accuracy. Intra-flow features are derived from raw byte spectrograms and statistical characteristics of individual flows, while inter-flow features are extracted from flow relation graphs to capture complex relationships between multiple flows. Evaluated on two real-world datasets, MeDF achieves high accuracy rates of 98.57% and 94.73%, surpassing both traditional single-modality methods and existing multi-modal approaches. By combining these features, MeDF addresses the limitations of current models and improves the effectiveness of encrypted traffic classification.

Horowicz et al. [48] present a novel approach to internet traffic classification using "Mini-FlowPics," which are smaller and more manageable than traditional FlowPics. By leveraging a limited number of labeled samples and applying augmentations that mimic network behavior—such as changes in round-trip time (RTT) and randomization of packet lengths—this approach enhances model performance even with minimal data. This method improves accuracy and simplifies engineering compared to larger FlowPics, offering an effective solution for traffic classification with limited labeled data.

Gioacchini et al. [49] emphasize the importance of word embedding techniques like DarkVec and iDarkVec for generating embeddings from network traffic, showcasing their effectiveness through Darknet and honeypot use cases. They argue that host embeddings from network data are more complex than those from natural language. Li et al. [50] address the challenges of single-modality classification with their FusionTC framework, which uses a stacking approach to extract features from packet distributions, sequences, and statistics, resulting in a 3.2% accuracy improvement for classifiers applied to custom mobile application traffic.

Park et al. [51] introduce a multi-task learning method for classifying encrypted network traffic using the DistilBERT model. This approach overcomes limitations of single-task methods by classifying traffic based on encapsulation, category, and application, achieving high accuracy (96.89%–99.29%). The authors implement weight adjustments for data imbalance and varying task difficulties, enhancing performance and efficiency while noting some trade-offs between speed and accuracy. Validation is performed using the ISCX 2016 VPN/Non-VPN dataset, with comparisons made to seventeen other methods. Future work aims to expand to additional datasets and enhance model efficiency. Mo et al. [52] introduced a hybrid model for network traffic classification that combines One-Dimensional Convolutional Neural Networks (1D-CNN), Temporal Convolutional Networks (TCN), and Gated Recurrent Units (GRU). This model efficiently extracts features from dynamic and encrypted network traffic, leveraging 1D-CNN for feature extraction, TCN for capturing temporal relationships, and GRU for sequential analysis. Evaluations show the model outperforms traditional methods in classification accuracy, demonstrating its potential for real-time applications in Software-Defined Networking (SDN). The work highlights the model's ability to enhance quality of service and address complex challenges in network traffic management.

Niu et al. [53] proposed a deep learning framework called DarkGuardNet, designed to identify dark web traffic and classify its applications. The framework employs Spatio-temporal Feature Fusion (STFF) and Multi-Head Self-Attention (MHSA) modules to extract features from network data, effectively addressing issues related to imbalanced datasets. When evaluated on a new dataset from ISCXVPN and ISCXTor, DarkGuardNet demonstrated superior performance compared to existing methods, achieving an accuracy of 0.99 in identifying darknet traffic and 0.98 in classifying applications. Baek et al. [54] proposed a combined machine learning and deep learning model that employs a filter and refine approach to manage large datasets. This work was tested on the ISCX-VPN 2016 and ISCX-Tor 2016 datasets. The results, when compared to transformer-based models like ET-BERT, demonstrated that the proposed model achieved an accuracy that was 3.9% higher and also provided faster classification speeds.

Table 1: Overview of models and associated research works

Sl. No.	Related work(s)	Model used
1	Hochst et al. [1]	Auto-encoders
2	Wang et al. [4]	Convolutional neural networks
3	Lotfollahi et al. [5]	
4	Ring et al. [7]	NLP models (word2vec)
5	Zhao et al. [13]	Bidirectional Encoder Representations from Transformers (BERT)
6	Lin et al. [20]	
7	Shahraki et al. [22]	Convolutional neural networks
8	Gonzalez et al. [29]	NLP models (word2vec)
9	Aceto et al. [31]	Convolutional neural networks
10	Rezaei et al. [32]	
11	Kallitsis et al. [37]	Auto-encoders
12	Feng et al. [38]	Convolutional neural networks/nlp models (word2vec)

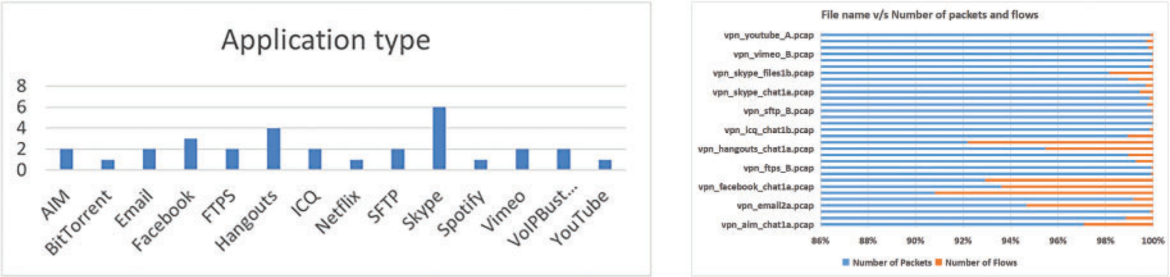
(Continued)

Table 1 (continued)

Sl. No.	Related work(s)	Model used
13	Houidi et al. [39]	Auto-encoders/NLP models (word2vec)
14	Gioacchini et al. [40]	NLP models (word2vec)
15	Yang et al. [41]	Auto-encoders
16	Gioacchini et al. [42]	NLP models (word2vec)
17	Pang et al. [44]	Bidirectional Encoder Representations from Transformers (BERT)
18	Gioacchini et al. [45]	NLP models (word2vec)
19	Cui et al. [46]	Convolutional neural networks
20	Wang et al. [47]	Bidirectional Encoder Representations from Transformers (BERT)
21	Horowicz et al. [48]	
22	Li Mo et al. [52]	Convolutional neural networks
23	Niu et al. [53]	
24	Baek et al. [54]	Combination of machine learning and deep learning models

3 Proposed System

The ISCX VPN-nonVPN 2016 datasets stands out as the most valuable resource for the analysis of encrypted network traffic. This comprehensive datasets includes pcap (packet capture) network recordings, which can be effectively processed using NFStream software. Notably, it allows for the analysis to be conducted even at the L7 (application layer) level, providing deep insights into the traffic patterns. To illustrate the effectiveness of the proposed analytical system, this study focuses exclusively on the pcap files generated from the VPN setup. Fig. 1 presents a detailed classification of 14 distinct applications represented within the VPN setup PCAP files. It also provides a comprehensive visualization of the flow composition for each PCAP file, highlighting the intricacies of data traffic and the relationships between various applications.



(a) Application classes

(b) Class count, Packet count, and flow count in each file

Figure 1: Application class distribution with class, packet, and flow counts per file

The architecture of the proposed method consists of three major subsystems: the Data Augmentation and pre-processing Module, the Multi-modal Feature Representation Module, and Application Classification using the 1D-CNN model, as illustrated in Fig. 2. First, flow information is captured from the NFStream application, and suitable data augmentations are then applied to the flow metadata to ensure the datasets is class-balanced. The next subsystem generates a generic feature representation by compressing the datasets with a multi-modal auto-encoder. Finally, application classification is performed by training a 1D-CNN model on the datasets.

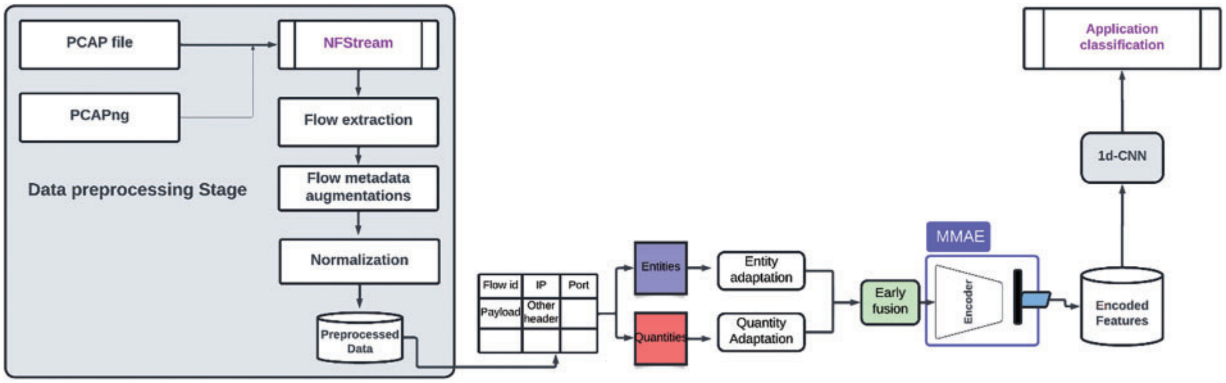


Figure 2: 1D-CNN with Multi-modal representation learning architecture

In this methodology, we outline a series of steps designed to ensure comprehensive data collection and analysis. These steps include defining the algorithmic perspective of flow processing, where flow features are categorized into entities and quantities. We will also establish criteria for evaluating the effectiveness of each algorithm to ensure that the results are reliable and reproducible. Algorithm 1 illustrates the extraction of quantities from network traffic and the pre-processing of these network quantities for further analysis. This process will involve filtering out noise and irrelevant data, allowing us to focus on the key metrics that influence performance. Meanwhile, Algorithm 1 builds upon the initial findings by applying statistical methods to identify patterns and correlations within the network flow to pinpoint the entities. This dual approach not only enhances our understanding of the underlying network dynamics but also facilitates the development of predictive models capable of scaling to large datasets for various downstream tasks.

The initial research is limited to analyzing non-VPN pcap files sourced from the ISCX datasets and does not include a mention of an integration module. To advance this work further, we propose the potential inclusion of fusion-based integration modules, which could facilitate the combination of various data sources. Additionally, deep or shallow learning models can be developed and trained on these compact representations to enhance their effectiveness. In our proposed architecture, we introduce a novel adaptation module that significantly improves the multi-modal auto-encoder's capabilities. This module is designed to classify network traffic in both VPN and non-VPN environments, addressing a gap in the original study. Furthermore, the adaptation module possesses the flexibility to dynamically adjust to various flow extractors, such as NFStream. It also classifies different attributes identified as Quantities and Entities, detailed comprehensively in Table 2. This refined approach aims to provide a more robust and adaptable framework for analyzing network traffic patterns.

Table 2: Commonly used **e**mbedding techniques for Entities and Quantities columns from pcap

Quantities columns	**Embedding type**
id, src_port, dst_port, protocol, ip_version, vlan_id, tunnel_id, bidirectional_first_seen_ms, bidirectional_last_seen_ms, bidirectional_duration_ms, bidirectional_packets, bidirectional_bytes, src2dst_first_seen_ms, src2dst_last_seen_ms, src2dst_duration_ms, src2dst_packets, src2dst_bytes, dst2src_first_seen_ms, dst2src_last_seen_ms, dst2src_duration_ms, dst2src_packets, dst2src_bytes, bidirectional_min_ps, bidirectional_max_ps, src2dst_min_ps, src2dst_max_ps, dst2src_min_ps, dst2src_max_ps, bidirectional_min_piat_ms, bidirectional_max_piat_ms, src2dst_min_piat_ms, src2dst_max_piat_ms, dst2src_min_piat_ms, dst2src_max_piat_ms	Linear embedding, Binning + One-hot
user_agent	Drop/Impute
expiration_id	(No embedding specified)
(no embedding specified for content_type)	Content_type

Entities columns	Embedding type
src_oui, dst_ip, dst_mac, dst_oui, application_name, application_category_name, requested_server_name, client_fingerprint, server_fingerprint	Word embedding (word2vec, GloVe, FastText)

Table 3: Performance of proposed classifier on different datasets

Data sample	**RF Classifier**	**1D-CNN**	**(1D-CNN with MMAE)**
vpn_aim_chat1a	0.99	0.25	0.75
vpn_aim_chat1b	0.98	0.25	0.89
vpn_bittorrent	0.98	0.69	0.88
vpn_email2a	0.96	0.48	0.48
vpn_email2b	1.00	0.46	0.75
vpn_facebook_audio2	0.99	0.44	0.75
vpn_facebook_chat1a	1.00	0.86	0.92
vpn_ftps_A	0.93	0.62	0.64
vpn_hangouts_audio2	1.00	0.40	0.96
vpn_hangouts_chat1b	1.00	0.95	0.96
vpn_icq_chat1a	0.88	0.54	0.55
vpn_netflix_A	0.97	0.53	0.95
vpn_skype_audio1	0.98	0.80	0.77
vpn_skype_files1b	0.98	0.56	0.71

(Continued)

Table 3 (continued)

Data sample	RF Classifier	1D-CNN	(1D-CNN with MMAE)
vpn_voipbuster1b	0.99	0.79	0.81
vpn_youtube_A	0.95	0.42	0.41

Algorithm 1: Network flow extraction and pre-processing of quantities and entities using NFStream

```
Input: Network traffic data
Output: Pre-processed Quantity and Entity embedding
1. Extract network flows using NFStream
2. Analyze Flow Metadata for Missing Values
3. Check flow metadata for any missing values
4. Handle missing values as per the chosen strategy (e.g., imputation or removal)
5. Normalize and Standardize the Quantities in the Flow
6. Normalize quantitative columns to a specific range (e.g., [0, 1])
7. Standardize quantitative columns to have zero mean and unit variance
8. Apply Scaling to Quantities and Remove Outliers if any
9. Detect and handle outliers using statistical methods (e.g., Z-score or IQR)
10. Apply scaling to adjust quantity values as needed
11. Use Quantity adaptation for generating embedding for Quantities
12. Adapt quantities and generate embedding for quantitative data
13. Store the Embedding to Dateset with class labels
14. Store the generated embedding into a datasets for further analysis or usage
```

3.1 Dataset Pre-Processing

The most useful datasets for encrypted traffic analysis is the ISCX VPN-nonVPN 2016 datasets, which provides pcap network captures that can be processed using NFStream and can process the pcap files even at the L7 layer. To show the effectiveness of the proposed system, only pcap files from the VPN setup are used.

The proposed work only considers the use of non-VPN pcap files from the ISCX dataset and does not explicitly mention the integration module. Fusion-based integration modules can be added as an extension to the work, and also training deep or shallow learning models can be achieved on compact representations. In the proposed architecture, we extend and propose to add an adaptation module for multi-modal auto-encoder architecture which can classify the Traffic under VPN and non-VPN setup. The proposed system has an adaptation module that can dynamically adapt to flow extractors such as NFStream and classify the attributes as Quantities and Entities listed in Table 2.

3.2 Quantities and Entities Pre-Processing from Datasets

Quantities serve as essential attributes that are derived from capture files, as detailed in Table 2. These files encompass a wealth of information concerning flow metadata, crucial for understanding underlying patterns. Prior to utilizing these attributes, it is vital to preprocess and encode them using suitable encoding techniques. This step is critical in ensuring that the datasets remains well-balanced and devoid of biases, thereby creating an optimal learning environment for the model. Such rigorous preparation enhances the model's ability to learn effectively and generalize successfully to unseen data.

In the feature selection process, particular attention must be directed toward identifying those features that significantly bolster the model's predictive power. This thoughtful selection process may involve various techniques, including feature selection, dimensionality reduction, and normalization. Each of these techniques plays a pivotal role in refining the datasets, thus improving the overall performance of the model. Once the attributes have been extracted, it is essential to analyze them for any missing values and normalize them according to their numerical and categorical types. Additionally, to maintain the integrity of the datasets, outliers are addressed and removed using the Inter quartile Range (IQR) technique applied across the feature list.

Upon completing these preparatory steps, the encoded quantities are ready to be employed in the fusion technique outlined in Section 3.3. Within the context of the proposed system, entities are methodically identified and extracted from captures during the pre-processing phase, subsequently integrating them into the fusion architecture. Table 2 provides a comprehensive list of the extracted attributes, while Algorithm 1 details the step-by-step process of entity embedding.

The embedding process is a transformation phase that converts these attributes into a multidimensional vector space. This transformation is instrumental for the efficient integration and analysis of data within the fusion framework. Through this process, the system significantly enhances its ability to identify complex patterns and relationships among the entities, thereby yielding more precise and insightful outcomes. Furthermore, these vector representations empower machine learning algorithms to harness these relationships, ultimately facilitating advanced predictive modeling and sharpening decision-making capabilities. Integrating diverse embedding from various quantities and entities leads to the creation of a sophisticated and multifaceted set of encoded features. These encoded representations are crucial, as they need to be systematically stored and effectively utilized to train the model. The distinctive feature representations extracted from these embedding play a vital role in enabling classifiers to accurately categorize traffic patterns.

To achieve optimal performance in classification, different fusion techniques can be employed. These techniques include early fusion, which merges features at the input level; late fusion, which combines results at the decision level; and decision-based fusion, which integrates predictions from multiple classifiers. Prior to employing any of these fusion methods, it is imperative to identify and select the relevant attributes that significantly contribute to traffic classification. This involves not only selecting the most pertinent features but also applying appropriate techniques to integrate them seamlessly, enhancing the overall effectiveness of the classification system.

In this context, we introduce an innovative Multi-modal Auto Encoder-Decoder Architecture (MMAE), designed to effectively integrate various modality features while also facilitating dimensionality reduction. This model intricately combines embedding derived from quantities and entities, as previously detailed in Section 3.3, utilizing the strengths of the multi-modal auto-encoder framework. The architecture of the MMAE consists of three critical components: an encoder layer, a bottleneck layer, and a decoder layer. The encoder layer plays a pivotal role in generating a latent representation of the input features, enabling the training of classification models within a more manageable lower-dimensional space. Within our proposed system, the embedding from both quantities and entities are processed through the MMAE, resulting in the creation of joint representations of these features. This synthesis is essential, as it allows for a richer, more cohesive understanding of the data that can enhance the performance of subsequent analytical tasks.

3.3 Proposed 1D-CNN with Multi-Modal Representation Architecture

The proposed model utilizes advanced auto-encoder and decoder architectures, which have been widely recognized for their effectiveness in feature dimension reduction and encoding tasks. By integrating a multimodal approach, this model aims to create unified and comprehensive representations of various types of input data. The joint representation derived from this encoded data can be leveraged for a range of critical operations, including the generation of synthetic packet headers and metadata, as well as effective feature extraction. These capabilities allow for more efficient data processing and analysis. Moreover, the Multi-Modal Auto-Encoder (MMAE) significantly enhances the generalization abilities of classifiers. This improvement empowers the model to accurately classify packet captures that it has not encountered before, thereby increasing its robustness and versatility in real-world applications.

The Multi-Modal Auto-Encoder (MMAE) model plays a crucial role in encoding quantities and entities into feature representations within a latent space. This process allows for a more nuanced understanding of the relationships and characteristics present in the data. To evaluate the effectiveness of these feature representations, we employ a loss function that incorporates various attributes from different modalities, ensuring a comprehensive assessment. One of the significant advantages of the MMAE model is its ability to reduce the size of the training set while still maintaining the integrity of the data. This reduction is particularly beneficial for training a one-dimensional convolution neural network (1D-CNN), as it allows for a more efficient learning process and improves model performance. In our proposed framework, we will thoroughly investigate the impact of the MMAE model on network traffic classification. This evaluation will be based on key performance indicators such as accuracy and other relevant metrics, providing a clear understanding of how MMAE contributes to the overall effectiveness of the classification task.

4 Evaluation Results and Analysis

4.1 Evaluation Setup

The proposed system was meticulously implemented using Python version 3.11 and the PyTorch library version 2.0.1 on a standalone machine that is equipped with an Intel(R) Xeon E5 1620 processor, along with 16 GB of RAM. This powerful configuration ensured that we could efficiently handle the computational demands of our experiments.

For our experimental approach, we specifically selected the ISCX VPN-nonVPN datasets, focusing exclusively on the PCAP (Packet Capture) files associated with VPN traffic. This decision allowed us to concentrate on the nuances of VPN usage while disregarding non-VPN data. We methodically extracted random samples from each application category represented within the datasets, ensuring that our analysis included 14 distinct applications in each PCAP file. Notably, these applications comprised popular services like Google mail for email communication, BitTorrent for peer-to-peer file sharing, and a variety of chat applications to represent real-time messaging.

To process the data effectively, we trained an auto-encoder designed to adapt entities and quantities based on the Multi-Modal Auto-Encoder (MMAE) methodology. The auto-encoder learns to compress and encode the input data into a lower-dimensional space, capturing the essential features while discarding less relevant noise. The encoded features generated by the auto-encoder were systematically stored and then utilized to train a one dimensional Convolution Neural Network (CNN) classifier, which serves to classify the VPN traffic based on the learned representations.

The architecture of the 1D-CNN is specifically designed for the nature of our input data, which consists of sequential VPN setup information. The Conv1D layer is configured with 32 filters, each utilizing a kernel size of 3, and employs the ReLU (Rectified Linear Unit) activation function to introduce non-linearity into the model, enhancing its ability to learn complex patterns. Following this, the MaxPooling1D layer is integrated with a pool size of 2, effectively down-sampling the representation and retaining only the most salient features.

The intermediate output is then passed through a flattened layer, which reshapes the multidimensional data into a one-dimensional array, preparing it for dense layer processing without introducing additional neurons.

The dense layer is composed of 100 neurons, which also employs the ReLU activation function to maintain activation levels after processing. Finally, the output layer is designed to contain a number of neurons that corresponds precisely to the number of unique classes identified in the datasets, ensuring that the model can accurately classify each instance of VPN traffic into its respective category. This structured training approach allows us to create a robust classifier capable of distinguishing between different types of VPN traffic effectively.

4.2 Evaluation Metrics

The ISCX 2016 VPN/Non-VPN datasets poses significant challenges for accurate model training and evaluation due to its highly imbalanced nature. This imbalance is further exacerbated by biases present in the attribute values, which can lead to over fitting in machine learning models. Specifically, the datasets comprise packet captures that originate from a variety of applications, each generating differing quantities of network flows. This variability can produce misleading results in experiments, as the distribution of traffic can heavily influence the outcomes. To effectively address the bias associated with VPN packet captures, it is crucial to implement robust evaluation strategies. While average accuracy measures can provide insights into the overall performance of the model, this metric alone may not fully reflect the complexities of the classification challenge. Fig. 3. depicts the accuracy of the proposed model trained on latent representations compared to that of a random forest model.

Therefore, incorporating Macro average metrics is recommended, as these metrics average the performance across different classes and help to mitigate the impact of class imbalance. By employing these evaluation metrics, practitioners can achieve a more comprehensive understanding of model performance, particularly in scenarios characterized by mixed network traffic. This approach will facilitate more reliable classification outcomes and enhance the model's ability to generalize effectively across diverse network conditions.

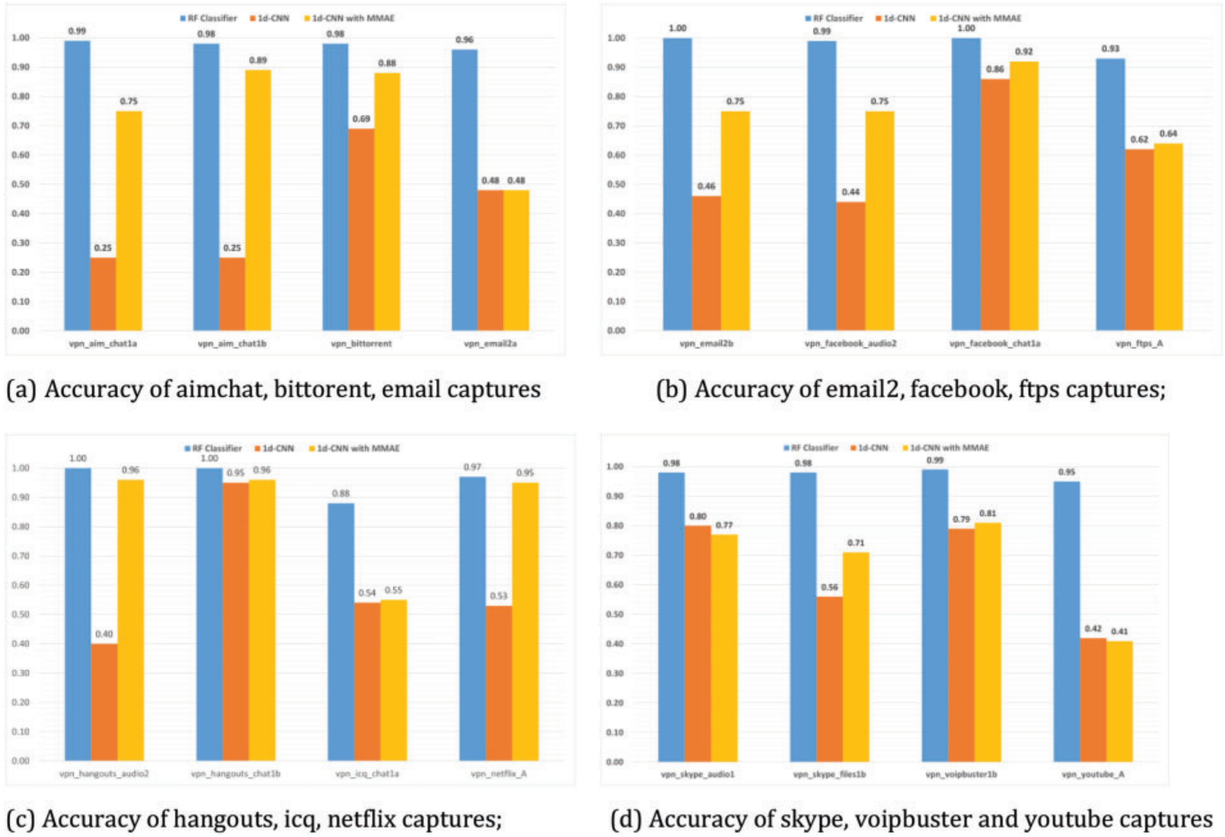
4.3 Evaluation Results

In this section, we provide a comprehensive overview of the experiments conducted and the resulting data to validate the effectiveness of the proposed method. We start by outlining the necessary hyperparameters for the classification model in Section 4.3.1, where we also present the performance metrics required for training the models. Section 4.3.2 provides a detailed performance comparison between our model and Tree-based classifiers, such as Random Forest. It discusses the strengths and weaknesses of each approach to contextualize how our method performs within the current landscape of methodologies.

Finally, Section 5 will delve into comprehensive discussions regarding our findings, addressing implications, potential limitations, and future directions for research to enhance the understanding of our method's impact and applicability.

4.3.1 Hyperparameters of Random Forest Classifier

To validate the effectiveness of the generic representations, we compare and evaluate the performance of the Random forest classifier with the proposed 1D-CNN model with MMAE support. Random forest is a supervised learning model, a labeled datasets can result in high accuracy. An accuracy of 99% is achieved with the hyper-parameters mentioned in Table 4.



(a) Accuracy of aimchat, bittorent, email captures

(b) Accuracy of email2, facebook, ftps captures;

(c) Accuracy of hangouts, icq, netflix captures;

(d) Accuracy of skype, voipbuster and youtube captures

Figure 3: Random forest and 1D-CNN model accuracies for PCAP flow classification

Table 4: Hyper-parameters of random forest model

Hyperparameter	Values
n_estimators	50, 100, 200
max_depth	None, 10, 20
min_samples_split	2, 5, 10
min_samples_leaf	1, 2, 4
bootstrap	True, False

4.3.2 Performance of Random Forest Classifier and 1D-CNN Model on Multi-Modal Representations

When comparing the accuracy of the Random Forest model and the 1D-CNN with Multi-Modal Auto-Encoder (MMAE), each exhibits unique strengths. The Random Forest model is robust and excels with structured datasets, effectively capturing feature relationships through ensemble learning. In contrast, the 1D-CNN with MMAE is adept at processing sequential or multimodal data, learning rich feature representations and often achieving higher accuracy in tasks involving time-series or varied inputs. While the Random Forest performs well on simpler tasks, the choice between models depends on the specific dataset and application context.

5 Conclusion

In conclusion, this research presents a compelling exploration of the multi-modal approach combined with an auto-encoder-decoder model for generating generic representations of encrypted network traffic across a diverse range of application classes. By employing a novel integration module that facilitates early feature fusion, the study effectively addresses the persistent challenges of class imbalance and biases introduced during the pre-processing stage. The experiments conducted on the ISCX datasets yielded insightful results, revealing that the random forest model excels in accurately classifying network traffic when trained on the generated embedding. Its strong performance underscores the model's effectiveness in navigating the complexities inherent in encrypted traffic scenarios. In contrast, the 1D-CNN model, while demonstrating high accuracy for larger sets of network flows, exhibits a decline in classification performance with fewer network captures. This indicates a potential area for further investigation into enhancing the model's robustness.

Despite the promising findings, several limitations warrant attention. The reliance on a specific datasets (ISCX 2016 VPN/Non-VPN) restricts the generalization of the conclusions drawn, highlighting the need for validation across a broader spectrum of datasets, such as ISCX Tor. Moreover, the methodology's dependence on an analysis of just eight packets can contribute to extended processing times, suggesting the necessity for optimization in this area. Looking ahead, future research endeavors should aim to broaden the datasets diversity, refine the model architectures, and enhance pre-processing techniques. These initiatives will not only boost performance and efficiency but also contribute to the development of more effective classification systems for encrypted network traffic. This work ultimately lays a strong foundation for advancing the field of network security and traffic management, promising significant implications for the protection and analysis of data in increasingly complex network environments.

Acknowledgement: I express my deepest gratitude to my guide, Dr. Poornima Athikatte Sampigerayappa, for her unwavering support, expert guidance, and valuable insights throughout this study. Her meticulous reviews, constructive feedback, and encouragement have been instrumental in shaping the direction and quality of this work. I am deeply appreciative of her time, effort, and commitment, which have significantly contributed to the successful completion of this study.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm their contributions to the manuscript as follows: study conception and design: Ravi Veerabhadrappa; data collection: Ravi Veerabhadrappa; analysis and interpretation of results: Poornima Athikatte Sampigerayappa and Ravi Veerabhadrappa; draft manuscript preparation: Ravi Veerabhadrappa. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available in the ISCXVPN2016 dataset at https://www.unb.ca/cic/datasets/vpn.html (accessed on 05 February 2025).

Ethics Approval: This study did not involve human or animal subjects, and therefore, ethical approval was not required.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Höchst J, Baumgärtner L, Hollick M, Freisleben B. Unsupervised traffic flow classification using a neural autoencoder. In: 2017 IEEE 42nd Conference on Local Computer Networks (LCN); 2017 Oct 9–12; Singapore: IEEE; 2017. p. 523–6. doi:10.1109/LCN.2017.57.
2. Bengio Y, Courville A, Vincent P. Representation learning: a review and new perspectives. arXiv:1206.5538. 2012.

3. Finsterbusch M, Richter C, Rocha E, Muller JA, Hanssgen K. A survey of payload-based traffic classification approaches. IEEE Commun Surv Tutorials. 2014;16(2):1135–56. doi:10.1109/SURV.2013.100613.00161.
4. Wang W, Zhu M, Wang J, Zeng X, Yang Z. End-to-end encrypted traffic classification with one-dimensional convolution neural networks. In: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI); 2017 Jul 22–24; Beijing, China: IEEE; 2017. p. 43–8. doi:10.1109/isi.2017.8004872.
5. Lotfollahi M, Zade RSH, Siavoshani MJ, Saberian M. Deep packet: a novel approach for encrypted traffic classification using deep learning. arXiv:1709.02656. 2017.
6. Wang W, Sheng Y, Wang J, Zeng X, Ye X, Huang Y, et al. HAST-IDS: learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection. IEEE Access. 2017;6:1792–806. doi:10.1109/ACCESS.2017.2780250.
7. Ring M, Dallmann A, Landes D, Hotho A. IP2Vec: learning similarities between IP addresses. In: 2017 IEEE International Conference on Data Mining Workshops (ICDMW); 2017 Nov 18–21; New Orleans, LA, USA: IEEE; 2017. p. 657–66. doi:10.1109/ICDMW.2017.93.
8. Zion Y, Aharon P, Dubin R, Dvir A, Hajaj C. Enhancing encrypted internet traffic classification through advanced data augmentation techniques. arXiv:2407.16539. 2024.
9. Wang C, Finamore A, Michiardi P, Gallo M, Rossi D. Data augmentation for traffic classification. arXiv:2401.10754. 2024.
10. Guthula S, Battula N, Beltiukov R, Guo W, Gupta A. netFound: foundation model for network security. arXiv:2310.17025. 2023.
11. Alwhbi IA, Zou CC, Alharbi RN. Encrypted network traffic analysis and classification utilizing machine learning. Sensors. 2024;24(11):3509. doi:10.3390/s24113509.
12. Wang X, Wei W, Yu X, Zheng D, Kuma N, Liu L. Ensemble learning-based traffic classification with small-scale datasets for wireless networks. In: IEEE INFOCOM 2024-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS); 2024 May 20; Vancouver, BC, Canada: IEEE; 2024. p. 1–6. doi:10.1109/INFOCOMWKSHPS61880.2024.10620836.
13. Zhao R, Zhan M, Deng X, Wang Y, Wang Y, Gui G, et al. Yet another traffic classifier: a masked autoencoder based traffic transformer with multi-level flow representation. Proc AAAI Conf Artif Intell. 2023;37(4):5420–7. doi:10.1609/aaai.v37i4.25674.
14. Jorgensen S, Holodnak J, Dempsey J, de Souza K, Raghunath A, Rivet V, et al. Extensible machine learning for encrypted network traffic application labeling via uncertainty quantification. IEEE Trans Artif Intell. 2024;5(1):420–33. doi:10.1109/TAI.2023.3244168.
15. Li Z, Xu X. L2-BiTCN-CNN: spatio-temporal features fusion-based multi-classification model for various Internet applications identification. Comput Netw. 2024;243(1):110298. doi:10.1016/j.comnet.2024.110298.
16. Qu J, Ma X, Li J. TrafficGPT: breaking the token barrier for efficient long traffic analysis and generation. arXiv:2403.05822. 2024.
17. Cui J, Bai L, Zhang X, Lin Z, Liu Q. The attention-based autoencoder for network traffic classification with interpretable feature representation. Symmetry. 2024;16(5):589. doi:10.3390/sym16050589.
18. Wadekar SN, Chaurasia A, Chadha A, Culurciello E. The evolution of multimodal model architectures. arXiv:2405.17927. 2024.
19. Kiflay A, Tsokanos A, Fazlali M, Kirner R. Network intrusion detection leveraging multimodal features. Array. 2024;22(10):100349. doi:10.1016/j.array.2024.100349.
20. Lin X, Xiong G, Gou G, Li Z, Shi J, Yu J. ET-BERT: a contextualized datagram representation with pre-training transformers for encrypted traffic classification. In: Proceedings of the ACM Web Conference 2022; 2022; Lyon, France: ACM. p. 633–42. doi:10.1145/3485447.3512217.
21. Alsaedi M, Ghaleb FA, Saeed F, Ahmad J, Alasli M. Multi-modal features representation-based convolutional neural network model for malicious website detection. IEEE Access. 2023;12(6):7271–84. doi:10.1109/ACCESS.2023.3348071.

22. Wang X, Lu Z, Wang X, He M, Wang X. GETRF: a general framework for encrypted traffic identification with robust representation based on datagram structure. IEEE Trans Cogn Commun Netw. 2024;10(6):2045–60. doi:10.1109/TCCN.2024.3400825.
23. Barua A, Ahmed MU, Begum S. A systematic literature review on multimodal machine learning: applications, challenges, gaps and future directions. IEEE Access. 2023;11(1):14804–31. doi:10.1109/ACCESS.2023.3243854.
24. Baek UJ, Lee MS, Park JT, Choi JW, Shin CY, Kim MS. Preprocessing and analysis of an open dataset in application traffic classification. In: 2023 24st Asia-Pacific Network Operations and Management Symposium (APNOMS); 2023; Sejong, Republic of Korea. p. 227–30.
25. Zhao F, Zhang C, Geng B. Deep multimodal data fusion. ACM Comput Surv. 2024;56(9):1–36. doi:10.1145/3674501.
26. Manzoor MA, Albarri S, Xian Z, Meng Z, Nakov P, Liang S. Multimodality representation learning: a survey on evolution, pretraining and its applications. arXiv:2302.00389. 2023.
27. Nascita A, Montieri A, Aceto G, Ciuonzo D, Persico V, Pescapé A. XAI meets mobile traffic classification: understanding and improving multimodal deep learning architectures. IEEE Trans Netw Serv Manag. 2021;18(4):4225–46. doi:10.1109/TNSM.2021.3098157.
28. Ramachandram D, Taylor GW. Deep multimodal learning: a survey on recent advances and trends. IEEE Signal Process Mag. 2017;34(6):96–108. doi:10.1109/MSP.2017.2738401.
29. Gonzalez R, Manco F, Garcia-Duran A, Mendes J, Huici F, Niccolini S, et al. Net2Vec: deep learning for the network. In: Proceedings of the Workshop on Big Data Analytics and Machine Learning for Data Communication Networks; 2017; Los Angeles, CA, USA: ACM. p. 13–8. doi:10.1145/3098593.3098596.
30. Vu L, Bui CT, Nguyen QU. A deep learning based method for handling imbalanced problem in network traffic classification. In: Proceedings of the Eighth International Symposium on Information and Communication Technology; 2017; Nha Trang City, Vietnam: ACM. p. 333–9. doi:10.1145/3155133.3155175.
31. Aceto G, Ciuonzo D, Montieri A, Pescapé A. Mobile encrypted traffic classification using deep learning: experimental evaluation, lessons learned, and challenges. IEEE Trans Netw Serv Manag. 2019;16(2):445–58. doi:10.1109/TNSM.2019.2899085.
32. Rezaei S, Liu X. Deep learning for encrypted traffic classification: an overview. IEEE Commun Mag. 2019;57(5):76–81. doi:10.1109/MCOM.2019.1800819.
33. Cohen D, Mirsky Y, Kamp M, Martin T, Elovici Y, Puzis R, et al. DANTE: a framework for mining and monitoring darknet traffic. In: Lecture notes in computer science. Cham: Springer; 2020. p. 88–109. doi:10.1007/978-3-030-58951-6_5.
34. Barut O, Luo Y, Zhang T, Li W, Li P. NetML: a challenge for network traffic analytics. arXiv:2004.13006. 2020.
35. Holland J, Schmitt P, Feamster N, Mittal P. New directions in automated traffic analysis. In: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security; 2021; Republic of Korea: ACM. p. 3366–83. doi:10.1145/3460120.
36. Abbasi M, Shahraki A, Taherkordi A. Deep learning for network traffic monitoring and analysis (NTMA): a survey. Comput Commun. 2021;170(3):19–41. doi:10.1016/j.comcom.2021.01.021.
37. Kallitsis M, Prajapati R, Honavar V, Wu D, Yen J. Detecting and interpreting changes in scanning behavior in large network telescopes. IEEE Trans Inf Forensics Secur. 2022;17(2):3611–25. doi:10.1109/TIFS.2022.3211644.
38. Feng R, Hu T, Jia X. VPN traffic classification based on CNN. In: 2022 14th International Conference on Computer Research and Development (ICCRD); 2022 Jan 7–9; Shenzhen, China: IEEE; 2022. p. 94–9. doi:10.1109/ICCRD54409.2022.9730292.
39. Ben Houidi Z, Azorin R, Gallo M, Finamore A, Rossi D. Towards a systematic multi-modal representation learning for network data. In: Proceedings of the 21st ACM Workshop on Hot Topics in Networks; 2022; Austin, TX, USA: ACM. p. 181–7. doi:10.1145/3563766.3564108.
40. Gioacchini L, Vassio L, Mellia M, Drago I, Ben Houidi Z, Rossi D. I-DarkVec: incremental embeddings for darknet traffic analysis. ACM Trans Internet Technol. 2023;23(3):1–28. doi:10.1145/3595378.
41. Yang Y, Yan Y, Gao Z, Rui L, Lyu R, Gao B, et al. A network traffic classification method based on dual-mode feature extraction and hybrid neural networks. IEEE Trans Netw Serv Manag. 2023;20(4):4073–84. doi:10.1109/TNSM.2023.3262246.

42. Gioacchini L, Cavallo A, Mellia M, Vassio L. Exploring temporal GNN embeddings for darknet traffic analysis. In: Proceedings of the 2nd on Graph Neural Networking Workshop 2023; 2023; Paris, France: ACM. p. 31–6. doi:10.1145/3630049.3630175.
43. Li P, Pei Y, Li J. A comprehensive survey on design and application of autoencoder in deep learning. Appl Soft Comput. 2023;138(7553):110176. doi:10.1016/j.asoc.2023.110176.
44. Pang B, Fu Y, Ren S, Shen S, Wang Y, Liao Q, et al. A multi-modal approach for context-aware network traffic classification. In: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2023 Jun 4–10; Greece: Rhodes Island, Greece; 2023. p. 1–5. doi:10.1109/ICASSP49357.2023.10095124.
45. Gioacchini L, Mellia M, Drago I, Houidi B, Rossi D. Learning generic multi-modal representations from network traffic for machine learning tasks [Internet]. [cited 2025 Feb 5]. Available from: https://ssrn.com/abstract=4524861.
46. Shi S, Han D, Cui M. A multimodal hybrid parallel network intrusion detection model. Connect Sci. 2023;35(1):2227780. doi:10.1080/09540091.2023.2227780.
47. Wang X, Yuan Q, Wang Y, Gou G, Gu C, Yu G, et al. Combine intra- and inter-flow: a multimodal encrypted traffic classification model driven by diverse features. Comput Netw. 2024;245(4):110403. doi:10.1016/j.comnet.2024.110403.
48. Horowicz E, Shapira T, Shavitt Y. Self-supervised traffic classification: flow embedding and few-shot solutions. IEEE Trans Netw Serv Manag. 2024;21(3):3054–67. doi:10.1109/TNSM.2024.3366848.
49. Gioacchini L, Mellia M, Vassio L, Drago I, Milan G, Ben Houidi Z, et al. Cross-network embeddings transfer for traffic analysis. IEEE Trans Netw Serv Manag. 2024;21(3):2686–99. doi:10.1109/TNSM.2023.3329442.
50. Li S, Huang Y, Gao T, Yang L, Chen Y, Pan Q, et al. FusionTC: encrypted app traffic classification using decision-level multimodal fusion learning of flow sequence. Wirel Commun Mob Comput. 2023;2023(9):9118153. doi:10.1155/2023/9118153.
51. Park JT, Shin CY, Baek UJ, Kim MS. Fast and accurate multi-task learning for encrypted network traffic classification. Appl Sci. 2024;14(7):3073. doi:10.3390/app14073073.
52. Mo L, Qi X, Liu L. Network traffic grant classification based on 1DCNN-TCN-GRU hybrid model. Appl Intell. 2024;54(6):4834–47. doi:10.1007/s10489-024-05375-4.
53. Niu T, Li W, Liu Y. DarkGuardNet: a deep learning framework for imbalanced dark web traffic identification and application classification. Research Square [Internet]. 2024 Feb 28. doi:10.21203/rs.3.rs-3974633/v1.
54. Baek UJ, Park JT, Jang YS, Kim JS, Choi YS, Kim MS. A filter-and-refine approach to lightweight application traffic classification. ICT Express. 2024;6:320. doi:10.1016/j.icte.2024.06.003.