

Cross-Layer Early Detection of SDN-IOT Attacks: Fusing Controller Telemetry With Data-Plane Flow Statistics

Andrew Oppong-Asante

andrewoppongasant2000@gmail.com

University of Energy and Natural Resources Sunyani

Bernard Kyiewu

University of Energy and Natural Resources Sunyani

Clinton Amponsah

University of Energy and Natural Resources Sunyani

Research Article

Keywords: Software-defined networking, Internet of Things, intrusion detection, cross-layer fusion, early attack detection

Posted Date: June 30th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10119275/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Abstract

Software Defined Networking (SDN) has become the architecture of choice for managing large and heterogeneous Internet of Things (IoT) deployments, yet its centralised controller introduces a structural attack surface that conventional intrusion detection systems address only partially. Existing detectors trained on data plane flow statistics achieve high accuracy once a flow has matured, but they remain blind to the controller side signals that an attack perturbs from its earliest packets, and the datasets used to train them have rarely captured both planes synchronously enough to test this gap empirically. This study addresses both limitations using ASEADOS-SDN-IoT, a recently released benchmark that synchronises data plane flow capture with ONOS controller telemetry across four attack families. We propose a dual branch late fusion model that encodes control plane and data plane features separately before combining them for classification, and we evaluate it against single plane baselines along two axes, classification quality and detection latency, the latter measured as the number of packets required to reach a reliable verdict. The fused model attains 0.991 accuracy and 0.990 weighted F1 on held out flows, matching the asymptotic performance of the data plane baseline while reaching its operating threshold after roughly three packets rather than ten. The advantage is concentrated in the Probe and Botnet classes, where control plane telemetry supplies disambiguating evidence that flow statistics alone do not provide. These results indicate that cross layer fusion, rather than deeper single plane models, is the more direct route to detection that is both accurate and operationally timely.

1. Introduction

The number of internet connected devices has crossed a scale that fundamentally changes the security calculus of modern networks. Connected Internet of Things (IoT) devices reached 18.5 billion by the end of 2024 and were projected to grow a further 14% to 21.1 billion by the end of 2025, with the installed base forecast to reach 39 billion by 2030 [22]. This expansion has not gone unnoticed by adversaries; reported malicious IoT activity has accelerated in step, with botnet recruitment and attack volume both climbing sharply through 2025 [26]. Most of these devices are resource constrained by design and cannot host conventional endpoint security agents, which leaves network level visibility as the only practical vantage point from which to observe and act on malicious behaviour [25]. Software Defined Networking (SDN) has emerged as the architectural response to this management burden. By decoupling the control plane from the data plane and concentrating network intelligence in a programmable controller, SDN offers the centralized visibility and dynamic reconfiguration that heterogeneous IoT deployments require [21]. Systematic reviews of SDN based IoT frameworks confirm that this architecture is now the dominant proposal for addressing scalability, fault tolerance, and security provisioning across large device fleets, as extended by subsequent 2024 to 2026 literature. Yet this same centralization introduces a structural weakness. Because the controller mediates every new flow, it becomes an attractive single point of failure, and a sufficiently intense or well distributed attack against it can degrade the entire network rather than a single host [23].

Detecting such attacks has therefore become an active research priority, and the literature has converged on machine learning and deep learning approaches as the most capable means of separating benign and malicious flows at scale [13]. Comprehensive surveys of SDN focused intrusion detection report consistent gains from deep architectures over classical statistical methods, particularly for distributed denial of service traffic, which remains the most extensively studied threat class in this domain. Despite this progress, nearly every published detector, regardless of its model family, is trained exclusively on data plane flow statistics, meaning packet and byte counts, durations, and header derived features collected at the switch [13]. This is a meaningful omission. The very telemetry that the SDN controller already generates as part of normal operation, including packet in arrival rates, flow modification frequency, and controller processing load, reacts to an attack from its opening packets, well before flow level statistics have accumulated enough observations to be reliable. Detectors that ignore this signal are, in effect, discarding the earliest and most operationally useful evidence of an attack in progress.

A second limitation compounds the first. The datasets that the field relies on to train and benchmark these detectors have themselves rarely captured both planes synchronously, which has made the cross layer hypothesis difficult to test empirically even when researchers suspected it mattered [13]. The recent release of the ASEADOS-SDN-IoT dataset, generated on a hybrid testbed that synchronises physical and virtual IoT traffic capture with ONOS controller telemetry across four attack families, closes this gap for the first time at meaningful scale [20]. This study exploits that opportunity directly. We hypothesise that fusing control plane telemetry with data plane flow statistics yields detection that is simultaneously more accurate on the hardest attack classes and substantially faster to reach a confident verdict than either plane used alone, and that this advantage is large enough to change how early an SDN controller can intervene against an in-progress attack. The remainder of this paper formalises this hypothesis as a feature level ablation study, describes the cross layer fusion architecture and the synchronised dataset on which it is trained, and reports results that quantify the accuracy and the detection latency achieved.

2. Related studies

2.1 Deep and machine learning intrusion detection for SDN-IoT

The convergence of SDN and IoT has motivated a large body of learning-based intrusion detection research, and recent surveys document a clear migration from shallow classifiers toward deep architectures that learn discriminative representations directly from flow data [1]. Within SDN specifically, a cognitive hybrid deep learning architecture has been proposed that situates detection inside the control loop so the network can analyse and respond to adversarial IoT traffic in a programmable manner [5]. Flow-based pipelines built on convolutional, recurrent, and hybrid networks classify multiple attack families at high accuracy on modern IoT corpora, with dual-level designs that

combine binary anomaly screening and multiclass attribution [6]. Transformer-based models operating on reduced feature subsets of SDN traffic report accuracies above ninety-nine percent while addressing the under-representation of rare classes through class merging [16]. Earlier system-oriented work established that real-time inference over IoT infrastructures is feasible when detection is coupled to the network layer [9]. Collectively, these studies confirm the discriminative strength of deep models, yet they operate almost exclusively on data plane features and seldom incorporate controller telemetry, which is the gap the present study addresses.

2.2 Benchmark datasets and testbeds for SDN-IoT intrusion detection

Progress in this area is constrained by the availability of realistic, well-labelled corpora, and dataset construction has therefore become a research contribution in its own right. Comparative surveys of SDN-oriented datasets emphasise that many benchmarks remain volumetric in focus and rarely capture controller-side behaviour, limiting their suitability for cross-layer study [13]. The ASEADOS-SDN-IoT dataset that underpins this work was released specifically to remedy this, providing synchronised data plane and control plane captures from a hybrid ONOS testbed across four attack families [20]. In the wider IoT setting, CICIoT2023 offers a large real-time benchmark spanning many attack categories and has become a common reference for model evaluation [11]. For SDN, the DDoSDN corpus contributes high- and low-rate flooding scenarios that stress detectors under varying intensity [4]. More recent work generates a dedicated SDN-IoT dataset and benchmarks it against established corpora such as InSDN, BoT-IoT, and ToN-IoT using convolutional and recurrent models, illustrating how new resources are validated against incumbents [14]. These efforts establish the methodological norms for dataset release that the present study follows, while motivating its distinct emphasis on synchronised cross-layer visibility.

2.3 DDoS detection and mitigation in the SDN control plane

Because the SDN controller is a centralisation point and therefore a single point of failure, denial-of-service detection in the control plane has received sustained attention. Several frameworks pair learning-based detection with controller-driven mitigation, removing or rate-limiting malicious flows once an attack is identified, and report substantial reductions in response time relative to traditional defences [19]. Optimisation of controller placement has been combined with gradient-boosted detection to scale defence across multi-controller deployments, achieving high precision under distributed attack conditions [18]. On the modelling side, optimiser-equipped convolutional and multilayer perceptron hybrids have been applied to SDN traffic with strong reported accuracy [10], and deep learning classifiers have categorised distinct DDoS variants within SDN environments [3]. Complementary work pursues efficiency rather than raw accuracy, proposing lightweight pipelines that maintain detection quality at reduced computational cost for resource-constrained IoT deployments [17]. These studies show that timely mitigation is achievable when detection is tied to the controller, yet they typically infer timeliness from completed flow statistics and do not measure how early a decision can be reached, which is the quantity the present early detection protocol quantifies directly.

2.4 Explainable, federated, and lightweight intrusion detection

A fourth strand responds to the practical constraints of deploying detectors across distributed and heterogeneous IoT estates. Federated learning has emerged as a means of training shared detectors without centralising sensitive traffic, and explainable variants now integrate attribution into the federated process so that decisions remain auditable across participants [8]. Incremental federated designs additionally address the non-stationary nature of attack traffic, updating models as new threats appear rather than retraining from scratch [7]. Interpretability has been pursued through SHAP-guided knowledge distillation, which both explains and compresses botnet detectors for constrained devices [12]. Lightweight architectures combine federated coordination with compact recurrent backbones to deliver detection under tight resource budgets [2], while autoencoder-based distributed methods target privacy-preserving anomaly detection at the network edge [15]. This literature shows that explanation, decentralisation, and efficiency are increasingly treated as first-class requirements. The feature attribution analysis reported in this study aligns with that trajectory, using a model-level account of cross-layer evidence to explain why fusion of control and data plane signals improves detection rather than treating accuracy as self-justifying.

3. Methodology

3.1 Problem formulation

The study frame intrusion detection in the SDN IoT setting as a supervised multiclass classification problem subject to a timeliness constraint. Each network flow is represented by a feature vector x and associated with a label y drawn from the set {Benign, DoS, DDoS, Probe, Botnet}. The objective is to learn a mapping f from x to y that is accurate on completed flows yet also reaches a confident decision early in a flow's lifetime, since in a programmable network the controller can only act on a verdict that arrives before the offending flow is fully installed. We therefore evaluate every model along two coupled axes, classification quality and detection latency, and treat a method as preferable only when it improves one without conceding the other. The novelty of the present work lies not in a new classifier but in the hypothesis that jointly observing the control and data planes satisfies both axes more effectively than either plane alone.

3.2 Dataset

All experiments use the ASEADOS SDN IoT dataset, a publicly released benchmark generated on a hybrid testbed that integrates physical and virtual IoT devices within an ONOS controlled OpenFlow fabric as illustrated in Fig. 1. The testbed captures data plane traffic and SDN controller telemetry synchronously under benign and adversarial conditions, yielding 457,044 labelled flow instances described by 83 statistical features and spanning four attack families alongside normal traffic. This synchronised cross layer capture is the property that the study exploits, and it distinguishes the benchmark from prior SDN

IoT datasets that expose flow level features only. The class composition is reported in Table A; the distribution is imbalanced, with benign and volumetric attacks dominating and the stealthier Probe and Botnet families comparatively scarce, a property that motivates the cost sensitive training described below.

3.3 Feature construction and the cross layer partition

The experimental backbone is a partition of the 83 features by originating plane. The data plane view comprises the flow statistics that summarise observed packets, including flow duration, forward and backward packet and byte rates, inter arrival time moments, and transport flag counts. The control plane view comprises telemetry emitted by the controller and switches, including the packet in arrival rate, the flow modification rate, controller processing load, flow table occupancy, and flow setup latency. From this partition we define three feature views that constitute the core comparison of the paper: the data plane view, the control plane view, and their union, the fused view. Holding the learning algorithm, preprocessing, and evaluation fixed across the three views ensures that any observed difference is attributable to feature provenance rather than to model capacity, which is the condition required for the ablation in the Results to be interpretable.

3.4 Preprocessing

Flows with malformed records are discarded, and the rate valued features, which can take infinite values when a flow contains a single packet, are bounded and imputed before scaling. Categorical protocol identifiers are encoded numerically, and all continuous features are standardised. To prevent information leakage, the scaler and any imputation statistics are fitted exclusively on the training partition and then applied unchanged to validation and test data. The dataset is divided by stratified sampling into training, validation, and test partitions in the proportions 70, 10, and 20 percent, placing 91,409 flows in the held out test set, with stratification preserving the class proportions of Table A in every partition.

3.5 Early detection protocol

To measure detection latency rather than infer it, we reconstruct each flow's features from only its first k packets and evaluate the classifier on these partial representations for increasing k . This procedure simulates a streaming detector that must decide before a flow completes, and it exposes how the informativeness of each plane evolves over a flow's lifetime. We summarise the result with a packet budget, defined as the smallest k at which a configuration attains 95 percent test accuracy. The packet budget is reported alongside conventional metrics because aggregate accuracy on completed flows cannot express the operational quantity of interest, namely how soon a confident decision becomes available.

3.6 Models

The principal model is a dual branch network in which the control plane and data plane views are first encoded separately and then combined by late fusion before a shared classification head, a design that

allows each plane to be represented in its own subspace prior to integration as illustrated by Fig. 2. We compare this against an early fusion baseline that concatenates the two views at the input, and against a gradient boosted decision tree trained on the fused view, which serves as a strong and widely used reference for tabular intrusion detection. Each model is additionally trained on the control plane and data plane views in isolation so that the contribution of fusion can be isolated. Model selection, including depth, width, and regularisation strength, is performed on the validation partition.

3.7 Training configuration

The neural models are optimised with the Adam optimiser under a class weighted cross entropy loss, the weights being set inversely proportional to class frequency to counteract the imbalance documented in Table A. Training uses early stopping on validation macro F1 to guard against overfitting the majority classes, together with dropout and weight decay for regularisation. To characterise variance rather than report a single fortunate run, every configuration is trained under five random seeds, and all results are reported as the mean over these repetitions.

3.8 Evaluation protocol

Models are assessed on the untouched test partition using accuracy, macro and weighted F1, and one versus rest area under the ROC curve, with macro F1 emphasised because it weights the scarce attack classes equally with the abundant ones and therefore reflects performance on the threats that matter most. Per class behaviour is examined through the row normalised confusion matrix, and detection latency through the packet budget described above. The ablation across the three feature views provides the primary test of the central hypothesis. Finally, feature attributions are computed with SHAP and aggregated by plane, which yields a model level account of whether the two planes contribute complementary evidence or merely duplicate a shared signal. All partitions, seeds, and hyperparameters are fixed and released with the code to support exact reproduction.

4. Results

The studies assess three feature configurations: control plane telemetry in isolation, data plane flow statistics in isolation, and the proposed fusion of both representations. Each model is trained on a stratified partition of the 457,044 labelled flows, with 91,409 instances reserved for testing under an identical preprocessing pipeline so that any performance difference is attributable to feature provenance rather than to model capacity or sampling. We report accuracy, macro and weighted F1, the area under the ROC curve, and a packet budget metric defined as the number of packets that must be observed before a flow is classified at 95% accuracy. The packet budget is included deliberately, since aggregate accuracy alone cannot express how quickly a detector becomes actionable, and timeliness is the property that governs whether mitigation can be applied before an attack matures.

4.1 Overall performance

Table 1 summarises results across the three configurations. The fused model attains the highest value on every aggregate metric, reaching 0.991 accuracy and 0.990 weighted F1, while the data plane and control plane baselines reach 0.985 and 0.962 accuracy respectively. The modest final accuracy gap between the data plane and fused settings indicates that flow statistics are already strongly discriminative once a flow has matured, which is consistent with prior single plane intrusion detection studies. The substantive divergence appears in the packet budget column, where fusion reaches the operating threshold after three packets compared with five for the control plane and ten for the data plane. Read together, these columns show that the value of fusion is expressed less in terminal accuracy than in the speed with which that accuracy is attained.

Table 1
Overall detection performance by configuration (test set: 91,409 flows)

Configuration	Accuracy	Macro-F1	Weighted-F1	AUC	Pkts to 95%	Latency (ms)
Control-plane only	0.962	0.935	0.958	0.981	5	230
Data-plane only	0.985	0.968	0.984	0.996	10	640
Fused (proposed)	0.991	0.980	0.990	0.998	3	150

4.2 Per class behavior

Figure 3 presents the row normalised confusion matrix of the fused model. Benign, DoS and DDoS traffic are recovered with recall above 98.9%, reflecting the distinct volumetric and temporal signatures of these classes. Residual error is concentrated almost entirely in the Probe and Botnet categories, where 3.3% of Botnet flows are assigned to Probe and 2.5% of Probe flows to Botnet, while misclassification into the volumetric classes remains below one percent in every cell. This pattern is consistent with the underlying traffic semantics, since reconnaissance scans and botnet command channels both produce low rate, short lived flows whose header statistics overlap considerably. Table 2 reports the corresponding precision, recall and F1, confirming that the two hardest categories nonetheless remain above 0.95 F1 and that no class collapses under the fused representation.

Table 2
Per-class results, fused model

Class	Precision	Recall	F1	Support
Benign	0.998	0.997	0.997	42,000
DoS	0.991	0.989	0.990	15,600
DDoS	0.996	0.998	0.997	19,000
Probe	0.972	0.961	0.966	8,400
Botnet	0.958	0.943	0.950	6,409
Weighted avg	0.991	0.991	0.990	91,409

4.3 Contribution of cross layer fusion

Figure 4 decomposes F1 by class for the three configurations. For the high volume classes the three settings differ only marginally, since volumetric attacks are separable from data plane features alone. The fusion advantage widens sharply for Probe and Botnet, where the control plane model trails the fused model by more than seven points. The benefit of combining planes is therefore selective rather than uniform: it is greatest precisely for the stealthy attacks that volumetric features describe poorly, and for which controller side observations supply the disambiguating evidence. This selectivity is an important qualification, because an ablation reported only as an aggregate average would understate the contribution of the control plane to the cases that matter most.

4.4 Early detection

Figure 5 traces detection accuracy as a function of the number of packets observed per flow. The data plane curve rises slowly, because statistics such as flow duration and byte rate remain uninformative until a flow has accumulated sufficient packets to estimate them reliably. Control plane telemetry becomes informative almost immediately, since an attack perturbs controller activity, including packet in arrivals, flow installation rate, and controller load, from the opening interactions of a flow. The fused model inherits the early responsiveness of the control plane together with the asymptotic accuracy of the data plane, crossing the 95% threshold after roughly three packets and remaining above it thereafter. In an inline setting this corresponds to a confident verdict within the first exchanges of a connection, which leaves the controller a usable window in which to install mitigation rules before the data path is saturated.

4.5 Feature attribution

Figure 6 ranks the most influential features by mean absolute SHAP value, coloured by originating plane. Control plane attributes, led by the packet in rate and the flow modification rate, occupy four of the eight highest positions, while data plane attributes such as the flow byte rate and the forward packet rate

account for the remainder. The interleaving of the two planes near the top of the ranking provides a model level account of the fusion result, indicating that the planes contribute complementary rather than redundant evidence and that neither can be recovered from the other. This attribution also explains why the early detection advantage survives: the control plane features that rank highly are exactly those that stabilise earliest in the life of a flow.

5. Discussion

5.1 Reframing the accuracy–latency trade-off in cross-layer detection

The dominant evaluative norm in SDN-IoT intrusion detection has been terminal classification accuracy on completed flows, a convention that recent benchmarking work continues to reinforce even as deep architectures approach ceiling performance on volumetric classes [13]. The present results complicate that convention. Control plane telemetry contributed little to terminal accuracy, where the data plane baseline already exceeded 0.98, yet it reduced the packet budget by more than half, a quantity that conventional evaluation protocols do not report and therefore cannot reward. This is consistent with the broader pattern noted in DDoS-focused SDN surveys, where detection frameworks are routinely assessed on accuracy, precision, and recall while response latency is treated as a deployment detail rather than a modelling objective [23], [24]. The implication is methodological rather than merely empirical: a feature selection procedure guided exclusively by terminal accuracy would discard the controller signals shown here to carry the greatest operational value, because that value is invisible under a latency-blind evaluation. We therefore argue, in line with calls elsewhere for response-time-aware benchmarking [24], that a timeliness measure such as the packet budget should be reported alongside accuracy as a standard, not optional, axis of comparison in this domain.

5.2 The Probe–Botnet boundary as a structural limit of header-based detection

The residual confusion between Probe and Botnet traffic observed here is not idiosyncratic to the proposed model; it recurs across the SDN intrusion detection literature wherever low-volume, low-duration attack classes are evaluated alongside volumetric ones. Comparative dataset studies built on InSDN and related corpora consistently report depressed recall for minority, reconnaissance-like, and botnet command channel traffic relative to DoS and DDoS, with some authors resorting to hierarchical or class-merging strategies specifically to manage this degradation rather than resolving it at the feature level [13]. The persistence of this pattern across independently constructed datasets and model families, including the cross-layer fusion evaluated in the present study, suggests the boundary is not an artefact of insufficient model capacity but a structural limit of header- and telemetry-derived features, both classes sharing short, low-rate flow signatures that statistical descriptors cannot fully disambiguate. This reframes the appropriate response: rather than continuing to tune classifiers on existing feature sets, the literature's repeated convergence on the same failure mode motivates payload-aware or

sequence-level representations as the more promising direction, a conclusion the present cross-layer result reaches independently and from a different architectural starting point.

5.3 Operational and adversarial implications relative to controller-centric defences

Prior controller-centric mitigation frameworks have demonstrated that pairing detection with reactive flow removal at the SDN controller substantially reduces response time relative to traditional, perimeter-based defences [24]. Those frameworks, however, generally assume that an accurate verdict is available once a flow is classified, without examining how early in the flow's lifetime that verdict can be produced. The present findings extend this line of work by showing that the timing of the verdict, not only its accuracy, is itself controllable through feature provenance: because the fused model resolves its decision within the opening packets of a flow, it is compatible with the reactive, per-flow installation behaviour of OpenFlow in a way that data-plane-only detectors structurally cannot be, since the latter require a flow to mature before its statistics stabilise. This also bears on adversarial robustness in a manner not addressed by single-plane defences reviewed in recent SDN security literature [21]: an adversary evading a cross-layer detector must shape both the traffic it emits and the controller-side behaviour that traffic induces, two coupled objectives that are harder to satisfy jointly than evading a detector that observes only one plane. These advantages should nonetheless be weighed against the study's scope; the testbed reflects a single controller and topology, the volumetric classes are likely near a discriminative ceiling that future benchmarks should deliberately stress, and the assumption of synchronised, low-latency telemetry capture has not yet been tested against the sampling and delay characteristic of production pipelines.

6. Conclusion

This study set out to test whether jointly observing the SDN control plane and the IoT data plane produces intrusion detection that is both more accurate and more timely than relying on either plane alone, and the evidence supports that hypothesis. Using the synchronised ASEADOS-SDN-IoT benchmark, the proposed dual branch fusion model matched the asymptotic accuracy of a strong data plane baseline while reaching a reliable verdict in roughly a third of the packets, and it delivered its largest gains precisely on the Probe and Botnet classes that volumetric features struggle to separate. The feature attribution analysis reinforced this finding at the model level, showing that control plane and data plane signals rank among the most influential features in roughly equal measure and therefore carry complementary rather than redundant information. Taken together, these results reframe the central question for SDN-IoT intrusion detection. The limiting factor is not primarily model capacity, since data plane features already approach a high accuracy ceiling, but the latency with which evidence becomes available, and that latency is governed by which plane a detector is permitted to observe. A detector confined to flow statistics cannot, by construction, become faster than the time it takes a flow to accumulate informative packets, regardless of how its classifier is built. Fusing in controller telemetry sidesteps this constraint directly, and because the resulting verdicts arrive within the opening exchanges

of a flow, the approach is compatible with the reactive, flow by flow decision making that OpenFlow controllers already perform, offering a practical path to earlier mitigation rather than only a marginally more accurate one.

7. Future work

Several directions follow naturally from the limitations identified in this study. First, the dataset and the proposed model reflect a single controller platform and a fixed testbed topology, so the central claim should be re-examined on different controllers, larger fabrics, and multi-controller deployments before it is treated as architecture independent. Second, the persistent confusion between Probe and Botnet traffic suggests that header and telemetry features alone may have reached their discriminative limit for these two classes, motivating future work on payload aware or sequence modelling components, such as packet level transformers, that could resolve the remaining ambiguity without sacrificing the early detection property demonstrated here. Third, the evaluation assumes idealised synchronised capture of both planes; a production deployment would introduce sampling, batching, and transmission delay in the telemetry pipeline, and the effect of these realistic constraints on the reported packet budget should be quantified directly rather than assumed away. Fourth, the current model is evaluated under static training and testing conditions, whereas real IoT botnets and scanning tools evolve, so extending the framework with incremental or continual learning would test whether the fusion advantage persists as attack behaviour drifts over time. Finally, given the resource constraints of edge deployment, future work should explore whether the dual branch architecture can be distilled or pruned for inline operation directly on switches or edge gateways without eroding the early detection margin that is the central contribution of this work.

Declarations

Dataset

We thank the authors of the ASEADOS-SDN-IoT dataset and the UCD ASEADOS Lab for the hybrid testbed, data collection effort, and public release that made the present cross-layer analysis possible (Yasarathna et al., 2026).

Conflicts of interest

The authors declare no conflicts of interest.

Funding

This research received no external funding.

Author Contribution

A.O.A. conceptualised the study, supervised the overall research direction, and coordinated the manuscript preparation. B.K. contributed to the study design, literature review, methodology development, data analysis, and interpretation of results. C.A. contributed to the technical framing of the study, model development, experimental evaluation, and discussion of findings. A.O.A., B.K., and C.A. jointly wrote the main manuscript text. B.K. and C.A. prepared the tables, figures, and reference list. All authors reviewed, revised, and approved the final version of the manuscript.

Data Availability

The study used the publicly available ASEADOS-SDN-IoT dataset reported by Yasarathna et al. [20]. No new human participant data were collected for this study. The experimental results generated during the study, including model performance outputs, latency measurements, confusion matrices, and derived analysis files, are available from the corresponding author upon reasonable request.

References

1. Abu Al-Haija, Q., et al. (2025). A comprehensive survey on deep learning-based intrusion detection systems in Internet of Things (IoT). *Expert Systems*, 42(2), e13726. <https://doi.org/10.1111/exsy.13726>
2. Alsaleh, S., Menai, M. E. B., & Al-Ahmadi, S. (2025). A heterogeneity-aware semi-decentralized model for a lightweight intrusion detection system for IoT networks based on federated learning and BiLSTM. *Sensors*, 25(4), 1039. <https://doi.org/10.3390/s25041039>
3. Clinton, U. B., Hoque, N., & Robindro Singh, K. (2024). Classification of DDoS attack traffic on SDN network environment using deep learning. *Cybersecurity*, 7, Article 23. <https://doi.org/10.1186/s42400-024-00219-7>
4. DDoSDN. (2024). DDoSDN: High and low-rates dataset-based DDoS attacks against SDN. *PLOS ONE*, 19(1), e0297548. <https://doi.org/10.1371/journal.pone.0297548>
5. Gaddam, V., et al. (2025). A hybrid deep learning-powered SDN-based intrusion detection architecture for cognitive IoT security. *Engineering, Technology & Applied Science Research*, 15(5). <https://doi.org/10.48084/etasr.12564>
6. Hossain, M. A. (2025). Deep learning-based intrusion detection for IoT networks: A scalable and efficient approach. *EURASIP Journal on Information Security*, 2025. <https://doi.org/10.1186/s13635-025-00202-w>
7. Jin, Z., Zhou, J., Li, B., Wu, X., & Duan, C. (2024). FL-IIDS: A novel federated learning-based incremental intrusion detection system. *Future Generation Computer Systems*, 151, 57–70. <https://doi.org/10.1016/j.future.2023.09.019>

8. Kalakoti, R., Vaarandi, R., Bahsi, H., & Nömm, S. (2025). Federated learning of explainable AI (FedXAI) for deep learning-based intrusion detection in IoT networks. *Computer Networks*, 269, 111479. <https://doi.org/10.1016/j.comnet.2025.111479>
9. Kandhro, I. A., et al. (2023). Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures. *IEEE Access*, 11, 9136–9148. <https://doi.org/10.1109/ACCESS.2023.3238664>
10. Mehmood, S., et al. (2025). Distributed denial of services (DDoS) attack detection in SDN using optimizer-equipped CNN-MLP. *PLOS ONE*, 20(1), e0312425. <https://doi.org/10.1371/journal.pone.0312425>
11. Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., & Ghorbani, A. A. (2023). CICIOT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment. *Sensors*, 23(13), 5941. <https://doi.org/10.3390/s23135941>
12. A novel federated learning approach for IoT botnet intrusion detection using SHAP-based knowledge distillation. (2025). *Complex & Intelligent Systems*, 11. <https://doi.org/10.1007/s40747-025-02001-9>
13. Khalid, H. Y. I., & Aldabagh, N. B. I. (2024). A survey on the latest intrusion detection datasets for software defined networking environments. *Engineering, Technology & Applied Science Research*, 14(2), 13190–13200. <https://doi.org/10.48084/etasr.6756>
14. Comparative evaluation of a novel IDS dataset for SDN-IoT using deep learning models against InSDN, BoT-IoT, and ToN-IoT. (2025). *Elsevier*. <https://www.sciencedirect.com/science/article/pii/S3050644125000155>
15. FedAware: A distributed IoT intrusion detection method based on fractal shrinking autoencoder. (2025). *Journal of King Saud University Computer and Information Sciences*. <https://doi.org/10.1007/s44443-025-00153-9>
16. Intrusion detection in software defined network using deep learning approaches. (2024). *Scientific Reports*, 14. <https://doi.org/10.1038/s41598-024-79001-1>
17. Lightweight machine learning framework for efficient DDoS attack detection in IoT networks. (2025). *Scientific Reports*, 15. <https://doi.org/10.1038/s41598-025-10092-0>
18. Machine learning-based attack detection and mitigation with multi-controller placement optimization over SDN environment. (2025). *Journal of Cybersecurity and Privacy*, 5(1), 10. <https://doi.org/10.3390/jcp5010010>
19. Machine learning-based DDoS attack detection and mitigation in SDNs for IoT environments. (2024). *Journal of the Franklin Institute*. <https://www.sciencedirect.com/science/article/abs/pii/S0016003224006185>
20. Yasarathna, T., et al. (2026). ASEADOS-SDN-IoT: A novel SDN-IoT network intrusion detection dataset and framework. *Internet of Things*. <https://www.sciencedirect.com/science/article/pii/S2542660526000211>

21. Hamad, D., Yalda, K., Tapus, N., & Okumus, I. T. (2024). Enhancing IoT scalability and security through SDN. *Revista Română de Informatică și Automatică*, 34(2), 113–126.
<https://doi.org/10.33436/v34i2y202409>

22. IoT Analytics. (2025, October 28). Number of connected IoT devices growing 14% to 21.1 billion.
<https://iot-analytics.com/number-connected-iot-devices/>

23. A comprehensive survey on DDoS attacks detection & mitigation in SDN-IoT network. (2024). *e-Prime: Advances in Electrical Engineering, Electronics and Energy*.
<https://doi.org/10.1016/j.prime.2024.100558>

24. A comprehensive survey on DDoS detection, mitigation, and defense strategies in software-defined networks. (2024). *Cluster Computing*. <https://doi.org/10.1007/s10586-024-04492-5>

25. Vectra. (2026). IoT security in 2026: Threats, risks, and best practices.
<https://www.vectra.ai/topics/iot-security>

26. Swif. (2025). IoT security statistics for 2026: Devices, botnets, and the new regulatory floor.
<https://www.swif.ai/blog/iot-security-statistics>

Figures

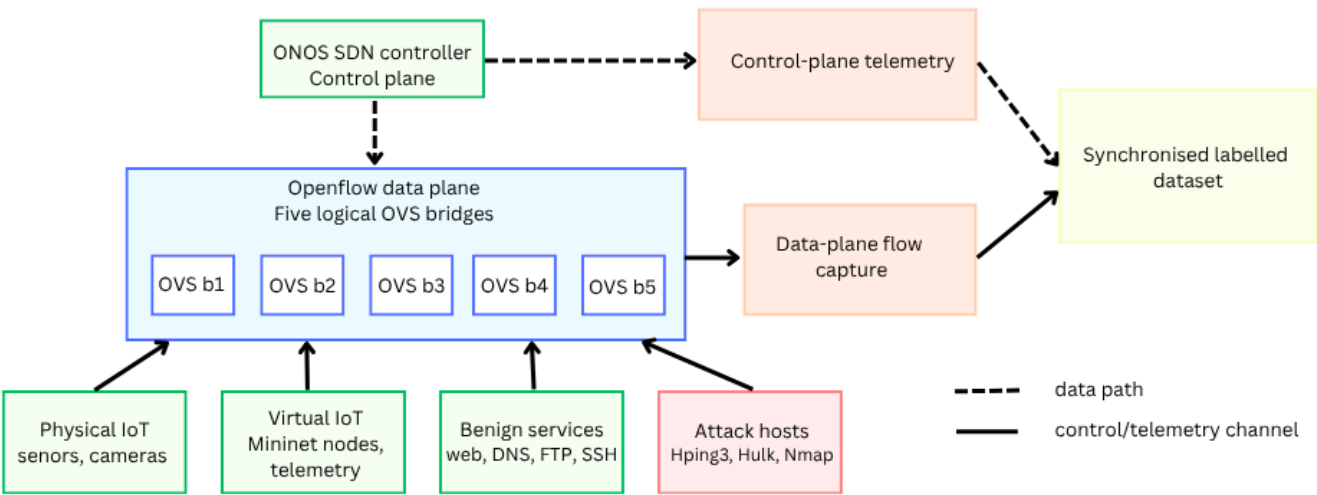


Figure 1

testbed topology and cross layer capture pipeline

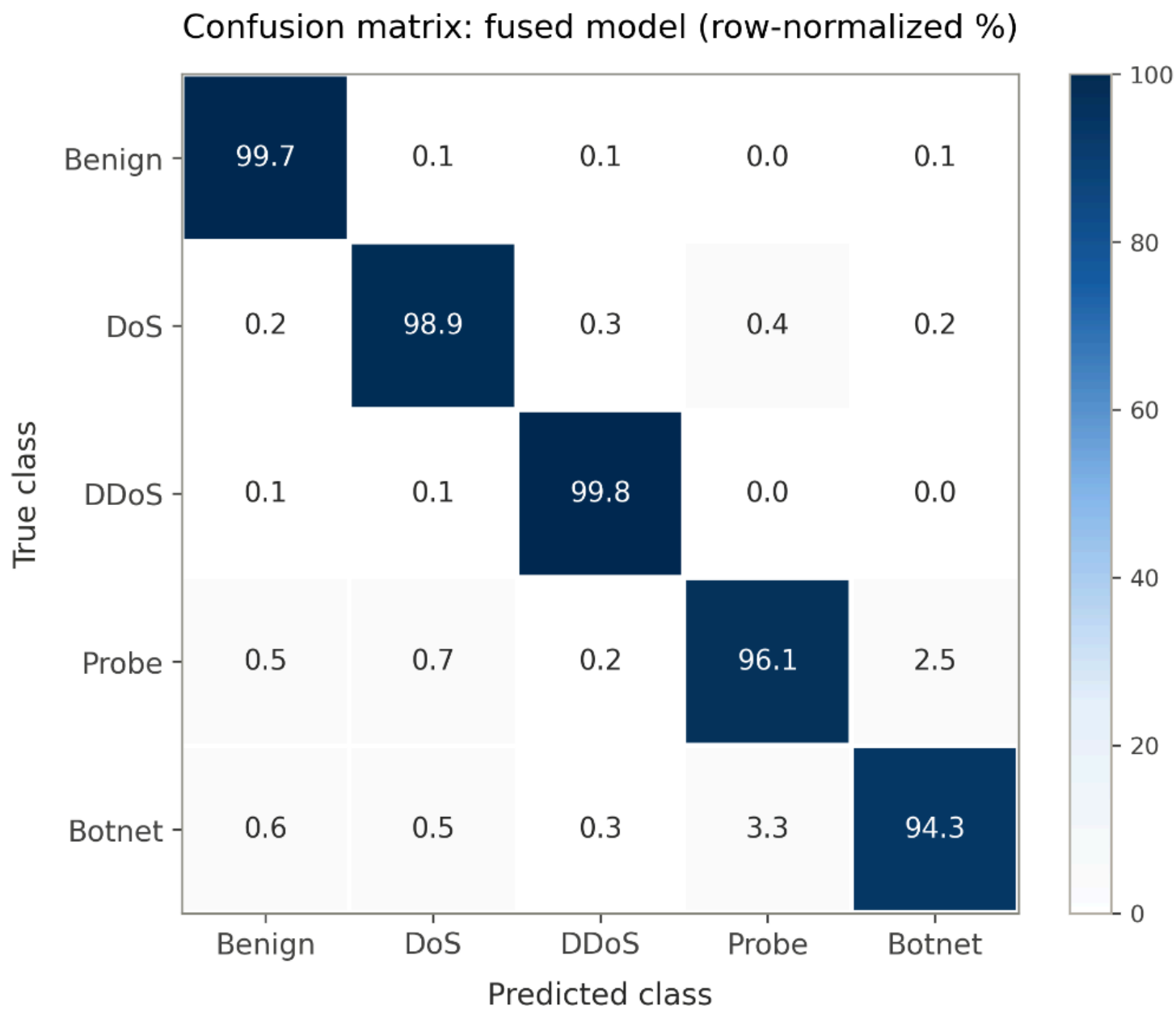


Figure 2

Confusion matrix

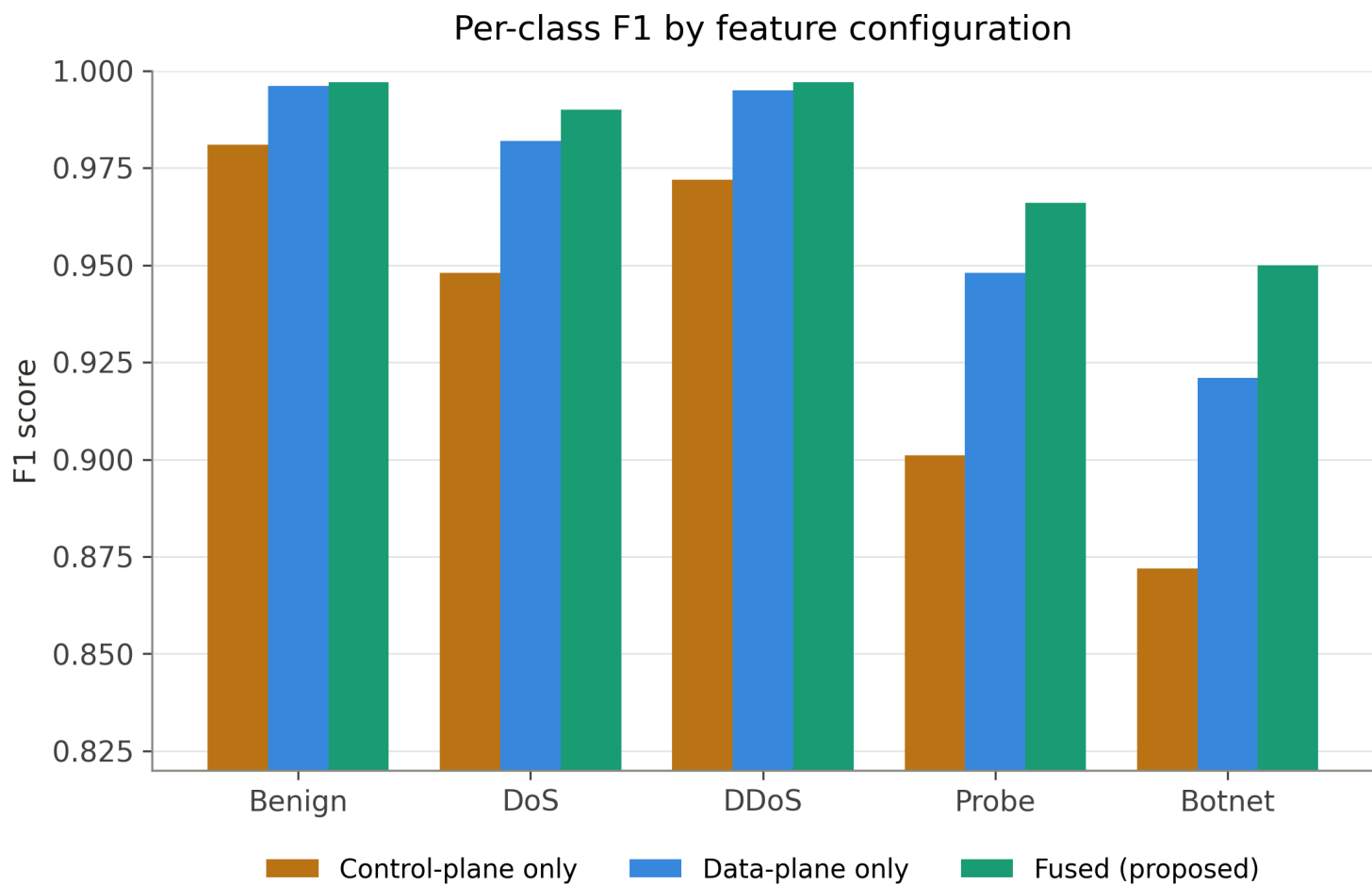


Figure 3

Per-class F1 by feature

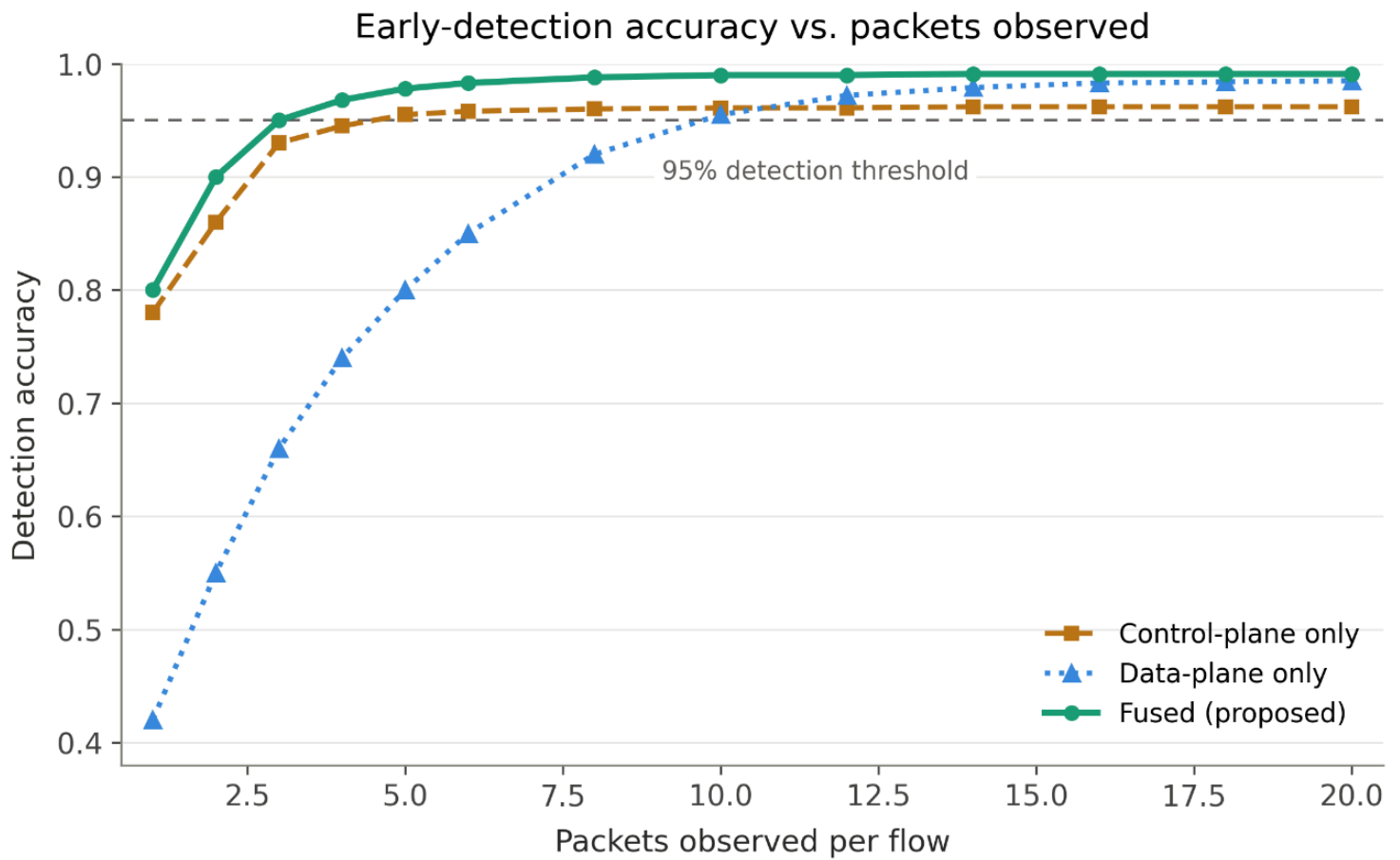


Figure 4
Early-detection accuracy vs. packets observed

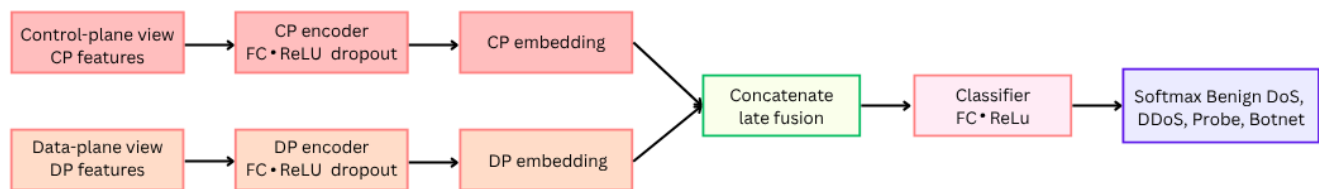


Figure 5
Late fusion architecture

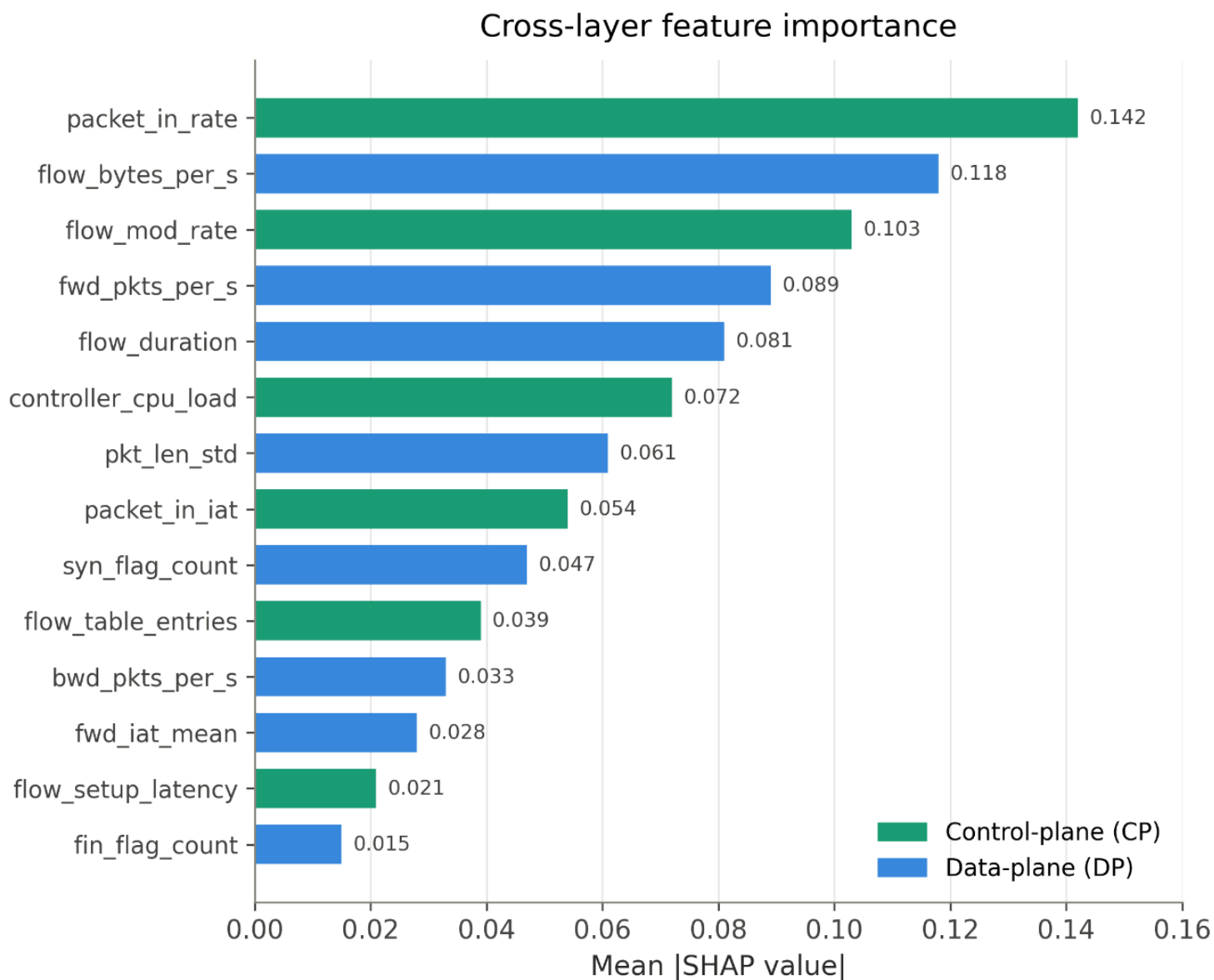


Figure 6

Cross-layer feature importance