Few-Packet Traffic Classification under Dynamic Network Conditions

Xujing Li[*,1], Zhuochen Fan[*,2], Yuchao Zhang[†,1], Yi Wang[1]
[1]Beijing University of Posts and Telecommunications, Beijing, China [2]Pengcheng Laboratory, Shenzhen, China

***Abstract*—Few-packet traffic classification is essential for network management and security monitoring but faces severe instability under dynamic network conditions. Packet loss and reordering cause inconsistent partial observations of the same flow, degrading both accuracy and reliability. We identify this observation inconsistency as a fundamental challenge and model it through the Partial Observation Consistency (POC). Further, we propose POC-FPC, a framework that learns consistent representations across multiple partial views. POC-FPC comprises three novel components: a multi-view generator simulating packet loss and reordering, an order-relaxed set-based flow encoder that reduces sensitivity to packet order, and a two-level consistency mechanism that aligns representations and predictions only on overlapping packet subsets. This avoids noise supervision from unobservable components. Finally, our experiments on multiple datasets show that POC-FPC improves accuracy up to 41.76% and prediction stability up to 25.6% over state-of-the-art solutions, demonstrating robust few-packet classification under realistic network perturbations.**

I. Introduction

Traffic classification [1]–[3] is a fundamental task for network management, security monitoring, and quality-of-service (QoS) provisioning. Many systems are required to perform classification under **few-packet** constraints, where only one to a handful of packets of a flow are observable. There are two main reasons for this requirement. First, since most Internet traffic is encrypted (e.g., TLS and QUIC), deep payload inspection is no longer feasible, and classification must rely on early packet-level features. Second, with classification functions increasingly deployed at resource-constrained edges and in-network devices, decisions are often needed at the early stage of a connection. In this few-packet regime, flow-level statistical characteristics are not yet fully formed, and classification decisions become highly sensitive to the content and composition of a limited set of packets, making accurate and reliable inference particularly challenging.

This challenge is further exacerbated under dynamic network conditions. In real-world networks, packet loss and reordering frequently arise due to wireless access variability, congestion, multi-path forwarding, and middle-box scheduling [4]–[6]. These perturbations alter not only packet arrival timing, but also which packets are observable and how they are structurally organized [1]. Especially in the case of few-packet settings, even minor packet loss or local reordering can significantly change the observed packet composition [7]. As a result, the same underlying flow can naturally give rise to multiple incomplete and structurally inconsistent partial observations under different network conditions, despite unchanged traffic semantics.

Existing studies on few-packet traffic classification under dynamic conditions mainly pursue three directions: earlier decision mechanisms that adaptively determine when to predict based on limited observations [8]–[11], finer-grained packet- or byte-level modeling to compensate for immature flow statistics [12]–[14], and robustness enhancement via relaxed sequence dependence or self-supervised learning with data augmentation [1], [10], [13], [15]–[19]. Despite their differences, most methods implicitly rely on a stable packet prefix or assume that different observations preserve comparable semantics. See § II for more details.

When few-packet constraints and network perturbations coexist, this assumption no longer holds. Under packet loss and reordering, the same flow may be observed through different incomplete packet subsets, each reflecting only a fragment of the underlying semantics. This gives rise to three fundamental challenges: **(i)** high prediction instability due to extreme sensitivity to packet visibility and order; **(ii)** inadequate consistency modeling that enforces alignment on misaligned or unobservable components; and **(iii)** a lack of systematic evaluation of prediction stability across partial observations. Existing methods, which either remain sensitive to specific packet positions or enforce global alignment without modeling these discrepancies, fail to address these issues.

To address these challenges, our key observation is that these partial observations should be treated as multiple views of the same flow rather than independent samples. If a model can learn flow-level representations that remain stable across such incomplete views, its predictions need not depend on any fixed packet prefix. This insight motivates us to reformulate few-packet traffic classification as a problem of **P**artial **O**bservation **C**onsistency (POC) modeling.

To this end, we propose **POC-FPC**, which formulates a few-packet traffic classification under dynamic network conditions as a POC learning problem. We first construct a multi-view observation mechanism to simulate independent and burst packet loss as well as local reordering, explicitly capturing cross-view packet correspondences. Building upon

[*]Xujing Li and Zhuochen Fan are co-first authors and contributed equally to this paper. [†]Corresponding author: Yuchao Zhang (yczhang@bupt.edu.cn).

979-8-3195-1703-6/26/$31.00 © 2026 IEEE

this, we adopt a hierarchical packet-to-flow representation that integrates fine-grained packet encoding with an order-relaxed set-based aggregation, reducing sensitivity to strict packet ordering. On top of this representation, we design a two-level consistency learning strategy. Consistency is enforced only on overlapping and alignable packet subsets: representation-level alignment is restricted to shared regions, while prediction-level alignment is adaptively gated by shared evidence. This selective alignment avoids introducing noisy supervision from mismatched or unobservable components, enabling the model to learn perturbation-insensitive and cross-view-stable representations. See § III for more details.

We conduct extensive experiments across multiple datasets and diverse packet loss and reordering settings in § IV. The results show that POC-FPC achieves up to 41.76% improvement in classification accuracy over state-of-the-art solutions, along with up to 25.6% gain in prediction stability. These advantages demonstrate that POC-FPC not only enhances accuracy but also substantially strengthens robustness and consistency under dynamic network conditions.

In summary, the main contributions of this paper include:

- We identify observation inconsistency induced by packet loss and reordering under dynamic network conditions as a fundamental cause of performance degradation and prediction instability in few-packet traffic classification.
- We propose POC-FPC, a POC framework that explicitly models partially overlapping observations and combines order-relaxed flow representation with selective two-level consistency learning to obtain perturbation-invariant and stable representations without introducing noisy supervision.
- We develop a multi-view partial-observation evaluation protocol with flow-level stability metrics and demonstrate consistent improvements in both accuracy and robustness across multiple datasets and perturbation settings.

II. Related Work

A. Few-Packet Traffic Classification

Early Decision Mechanism-based Few-Packet Traffic Classification. Early work [20], [21] showed that accurate identification is possible using only the first 4-7 packet lengths, establishing the feasibility of few-packet classification. Salman et al. [11] proposed an information-theoretic stopping rule that tracks mutual information and training accuracy to determine the minimal packets needed for prediction. TaTic [8] separates easy and hard flows to make early decisions with reduced delay. FastFlow [10] formulates early classification as a sequential decision problem with a Q-learning-based controller, requiring on average 8.37 packets in deployment. RoNeTC [22] uses uncertainty estimation to trigger prediction when the model becomes sufficiently confident. These methods improve early decision efficiency, but mainly focus on when to predict, rather than how to ensure stability when early observations are incomplete or structurally inconsistent.

Fine-grained Packet-level and Byte-level Semantic Modeling. Another line of work improves few-packet classification by extracting richer features from limited observations. FS-Net [23] learns flow representations from packet length sequences with a bidirectional GRU encoder-decoder. ET-BERT [12] learns contextualized datagram representations through pretraining, while Pcap-Encoder [3] further exploit protocol header and payload structures. RBLJAN [24] models header-payload interactions with byte-label joint attention. TFE-GNN [14] and MH-Net [13] introduce graph-based or multi-granularity modeling to capture fine-grained traffic semantics. These methods enhance few-packet representations, but largely assume stable observations and rarely account for partial-observation inconsistency under packet loss and reordering.

B. Classification under Dynamic Network Perturbations

Structural Modeling and Relaxation of Sequence Dependency. Some studies reduce sensitivity to packet order by replacing raw packet sequences with more robust structural representations. FlowPic [9] represents traffic as size–time histograms to decouple linear order through spatial patterns. EC-GCN [25] learns adaptive graph structures to capture dependencies under packet loss or reordering. DigTraffic [26] and MH-Net [13] build message-level or heterogeneous graphs with contrastive learning to enhance structural invariance. MIETT [27] adopts multi-instance learning with two-level attention to preserve intra-flow structure, while FG-SAT [15] introduces a Flow Graph abstraction with stability-based selection for environmental adaptability. These methods improve robustness to local perturbations but still assume learnable structural correspondences across observations.

Semantic Invariance Modeling and Cross-Environment Distribution Generalization. Another line of work learns invariant representations across noisy or heterogeneous environments. TrafficFormer [16] introduces pretraining objectives for packet loss and reordering awareness. Rosetta [1] incorporates TCP-aware augmentation to extract protocol-invariant features across heterogeneous network conditions. ReCDA [17] and Holmes [18] address distribution drift through drift-aware perturbation and contrastive adaptation, respectively. RBLJAN [24], AN-Net [19] and MOTA [28] improve robustness under noisy or adversarial settings through adversarial training, attention reweighting, or collaborative inference. MetaTraffic [29] employs meta-learning to learn stable feature representations across heterogeneous network conditions, but it focuses on cross-condition distribution shifts at the flow level, rather than the partial-observation inconsistency caused by packet loss and reordering in few-packet scenarios.

III. Methodology

A. Framework Overview

We present a unified learning framework for few-packet traffic classification under dynamic network perturbations, with the goal of achieving accurate and stable predictions across different partial observations of the same original flow. As illustrated in Fig. 1, the framework consists of three modules: a Multi-View Observation Generator (MVOG), a Few-Packet Flow Representation module, and a Partial Observation Consistency (POC) Learning module. These modules work

together to enhance prediction stability and accuracy under dynamic network conditions.

- **The MVOG module** simulates dynamic network perturbations (packet loss and reordering) to generate multiple partial observation views from the same clean flow, identifying overlapping and alignable packets.
- **The Few-Packet Flow Representation module** encodes packet sequences (e.g., header and payload) and aggregates them into a flow representation using an order-relaxed set-based mechanism, reducing sensitivity to packet order.
- **The POC module** enforces consistency in prediction distributions and representation space only on truly overlapping packet subsets, using shared packets as semantic anchors and avoiding noisy supervision.

B. Problem Definition

We study the few-packet traffic classification under dynamic network perturbations, where packet loss and reordering lead to partial and inconsistent observations. In this work, an original flow (also referred to as an underlying flow) refers to a communication entity with a unique semantic label (e.g., a connection or application session). Under ideal conditions without network perturbations, an original flow can be represented by its first K packets in canonical order $X = (x_1, x_2, \ldots, x_K)$, where x_i denotes the i-th packet composed of header fields and payload bytes, and K is a fixed packet budget. Each original flow is associated with a class label $y \in \{1, 2, \ldots, C\}$.

In real networks, packet loss and reordering affect observable packets and their order, leading to multiple partial observation views of the same flow, each reflecting specific network conditions. We denote the v-th observation view as:

$$\tilde{X}^{(v)} = \mathcal{O}_v(X), \quad v = 0, 1, \ldots, V-1 \tag{1}$$

where $\mathcal{O}_v(\cdot)$ represents a network-induced observation process. Due to packet loss and reordering, $\tilde{X}^{(v)}$ is typically may contain fewer than K packets, with views varying in packet content and sequence structure.

To explicitly model partial observability and enable cross-view alignment, we record an alignment mapping π_v from original packet index i to its position j in view v:

$$\pi_v : i \mapsto j \tag{2}$$

If packet x_i is not observable in view v, we set $\pi_v(i) = -1$. Based on π_v, we define the overlap set between the reference view $v = 0$ and a perturbed view v as:

$$\Omega^{(0,v)} = \{\, i \mid \pi_0(i) \geq 0,\ \pi_v(i) \geq 0 \,\} \tag{3}$$

and the corresponding overlap ratio:

$$r^{(0,v)} = \frac{|\Omega^{(0,v)}|}{K} \tag{4}$$

Only overlapping packets are simultaneously observable and alignable across views, thus serving as reliable anchors for consistency.

Given multiple partial observations $\{\tilde{X}^{(v)}\}_{v=0}^{V-1}$ originating from the same original flow, our goal is to learn a classifier $f_\theta(\cdot)$ that achieves **(i)** strong discriminative performance under a fixed packet budget K, and **(ii)** stable predictions across different partial views induced by dynamic perturbations.

C. Multi-View Observation Generator (MVOG) Module

The goal of the module is to explicitly model packet loss and reordering in dynamic networks under few-packet constraints as a controllable and reproducible multi-view observation generation process. This allows the same original flow to naturally correspond to multiple structurally inconsistent partial observation views under different network conditions, while providing aligned structural information to support subsequent consistency modeling.

*1) **Modeling Dynamic Network Perturbations**:*

For each original flow segment X, MVOG generates V observation views $\{\tilde{X}^{(v)}\}_{v=0}^{V-1}$. The reference view $v = 0$ keeps the original K-packet prefix without perturbation, while each perturbed view $v \geq 1$ is produced by an observation operator:

$$\mathcal{O}_v \triangleq \mathcal{R}_v \circ \mathcal{S}_v \tag{5}$$

where S_v models packet loss by selecting observable packets and R_v models packet reordering by permuting the retained packets.

Packet Selection Operator S_v: We characterize packet observability using a binary mask $\{m_i^{(v)}\}_{i=1}^{K}$, where $m_i^{(v)} = 1$ indicates that the i-th packet of the original flow is observable in view v, and $m_i^{(v)} = 0$ indicates a drop. The observable packet index set of view v is defined as

$$I^{(v)} = \{\, i \mid m_i^{(v)} = 1 \,\} \subseteq \{1, \ldots, K\}, \quad |I^{(v)}| \geq 1 \tag{6}$$

MVOG supports two representative packet loss patterns:

(i) Independent loss: which models sporadic packet drops by assuming that each packet is retained independently:

$$m_i^{(v)} \sim \text{Bernoulli}(1 - p_{\text{drop}}). \tag{7}$$

(ii) Burst loss: which models temporally correlated packet drops (e.g., under wireless fluctuations or congestion) using a bursty process such as a two-state Gilbert–Elliott model, producing contiguous dropped segments and reducing cross-view overlap.

Reordering Operator R_v: Given the observable packet set $I^{(v)}$, the reordering operator R_v permutes their order to simulate arrival disorder. Since packet reordering in practical networks is usually local rather than arbitrary, MVOG mainly adopts local reordering within a window of size w:

$$|\sigma^{(v)}(t) - t| \leq w, \quad t = 1, \ldots, |I^{(v)}| \tag{8}$$

where $\sigma^{(v)}$ denotes the permutation applied to the retained packet sequence. A global permutation can also be used as a stronger stress-test setting.

*2) **Alignment Mapping and Cross-view Overlap Structure**:*

Beyond producing perturbed views, MVOG explicitly records the packet-level correspondence between each observation view and the original flow. Specifically, when a packet x_i is retained and placed at position j in view v, MVOG records $\pi_v(i) = j$; otherwise, $\pi_v(i) = -1$. This mapping(as defined in Eq.2)) directly determines the overlap set and overlap ratio between the reference view and each perturbed view, as defined in Eq.3-4.

The overlap set $\Omega^{(0,v)}$ identifies packets that are simultaneously observable and alignable across views. These packets serve as reliable anchors for representation-level consistency,

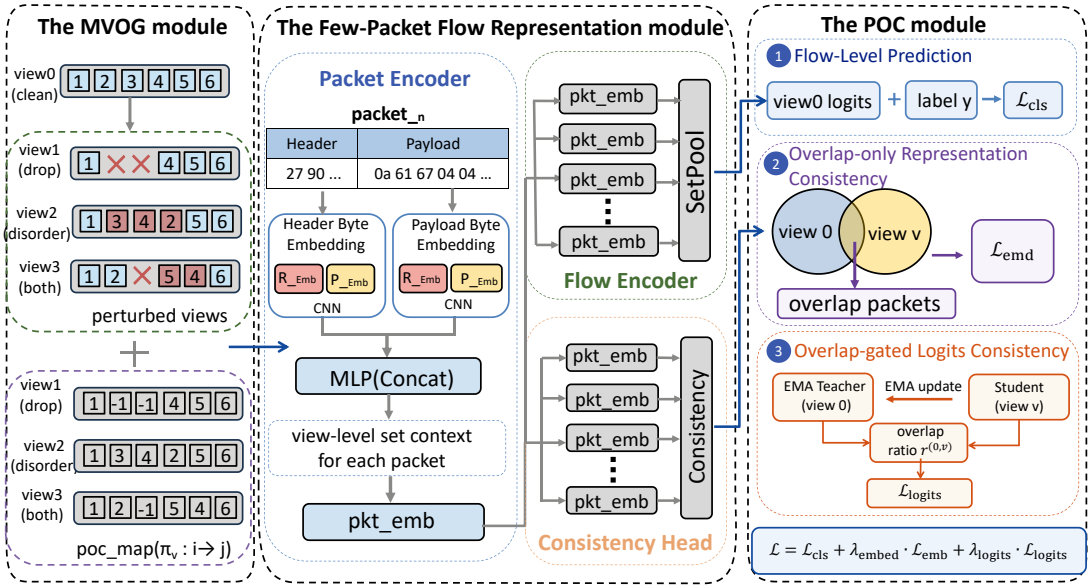


Fig. 1. The framework of POC-FPC.

while the overlap ratio provides an evidence signal for gating prediction-level consistency. In this way, MVOG provides not only perturbed inputs, but also the structural information required by POC to avoid aligning unobservable or mismatched packets.

D. Few-Packet Flow Representation Module

In few-packet scenarios, flow-level statistical characteristics have not yet fully formed, and the information available to the model mainly comes from the local structure and byte content of a very small number of early packets. Under dynamic network conditions, packet loss and reordering further alter the set of observable packets and their ordering, making modeling approaches that rely on strict sequential structures prone to instability. To address these challenges, we adopt a hierarchical packet-to-flow representation strategy that **(i)** preserves stable intra-packet structural information and **(ii)** relaxes inter-packet ordering dependency at the flow level. This design preserves fine-grained packet semantics while reducing dependence on strict inter-packet order.

*1) **Packet Encoder**:*

For an observation view $\tilde{X}^{(v)} = (\tilde{x}_1^{(v)}, \ldots, \tilde{x}_M^{(v)})$, where $M \leq K$, each packet consists of a header byte sequence and a payload byte sequence, denoted as $header_i \in \{0, \ldots, 255\}^{L_h}$ and $payload_i \in \{0, \ldots, 255\}^{L_p}$. Since encrypted payloads exhibit high randomness but still contain discriminative local structures (e.g., protocol headers, handshake patterns, length and direction information), we encode packets at the byte level to capture stable fine-grained patterns.

To this end, we construct a byte-level encoder separately for packet headers and payloads to preserve their different semantic roles. Specifically, each byte value is first mapped to a learnable embedding vector, and positional embeddings are introduced to preserve the relative ordering information within a packet:

$$\mathbf{u}_{i,j} = \mathrm{Emb}(b_{i,j}) + \mathrm{PosEmb}(j) \tag{9}$$

where $b_{i,j} \in \{0, \ldots, 255\}$ denotes the $j-th$ byte in the $i-th$ packet, $\mathrm{Emb}(b_{i,j}) \in \mathbb{R}^d$, and d is the input embedding dimension. It is important to emphasize that the order modeling here is restricted to the byte sequence within a single packet; this structure remains stable under network perturbations and is independent of cross-packet arrival order.

Based on this representation, we employ a one-dimensional convolutional neural network (1D-CNN) to extract local patterns along the byte dimension, followed by masked mean pooling over the sequence dimension to obtain a fixed-dimensional segment representation:

$$\mathbf{s}_i = \mathrm{MaskedMeanPool}(\mathrm{CNN}(\mathbf{u}_{i,1:L})) \tag{10}$$

We obtain header and payload representations $\mathbf{s}_i^h$ and $\mathbf{s}_i^p$, and fuse them through a feed-forward projection:

$$\mathbf{h}_i = \psi\left([\mathbf{s}_i^h \,\|\, \mathbf{s}_i^p]\right) \tag{11}$$

where $h_i \in \mathbb{R}^D$ denotes the packet embedding, and$\psi(\cdot)$ is a two-layer multilayer perceptron (MLP) with dropout, used to perform dimensional mapping and feature interaction. Accordingly, we obtain the packet-level embedding sequence $\mathbf{H} = (\mathbf{h}_1, \ldots, \mathbf{h}_M) \in \mathbb{R}^{M \times D}$, which serves as the basic unit for subsequent flow-level representation learning.

*2) **Order-Relaxed Set-Based Flow Aggregation**:*

Given the packet embeddings $H^{(v)} = \{h_i^{(v)}\}_{i=1}^M$, we first derive a view-level set context from the visible packet set and inject it into each packet embedding. This yields contextualized packet representations $\hat{H}^{(v)} = \{\hat{h}_i^{(v)}\}_{i=1}^M$, which preserve packet-level semantics while incorporating information from the current observation view. Since the context is computed from the visible packet set rather than the packet arrival order, this step remains order-relaxed and is less sensitive to local reordering.

We then perform attention-based set pooling over the contextualized packet embeddings:

$$\mathbf{f}^{(v)} = \sum_{i=1}^{M} \alpha_i \hat{h}_i^{(v)}, \quad \alpha_i = \mathrm{softmax}_i \left(w^\top \tanh(W \hat{h}_i^{(v)}) \right) \tag{12}$$

where α_i measures the contribution of each visible packet to the flow representation. This aggregation allows the model to emphasize informative packets while suppressing weak or noisy ones. Because the flow representation is computed over the visible packet set, rather than a strict packet sequence, it reduces sensitivity to packet reordering and partial packet loss.

*3) **Flow-Level Prediction**:*

Based on flow-level representation $\mathbf{f}^{(v)} \in \mathbb{R}^D$, we employ a two-layer MLP as the classifier, which produces the prediction distribution:

$$\mathbf{p}^{(v)} = \mathrm{softmax}\big(\mathbf{W}_2\, \sigma(\mathbf{W}_1 \mathbf{f}^{(v)} + \mathbf{b}_1) + \mathbf{b}_2\big) \tag{13}$$

where $\mathbf{W}_1, \mathbf{W}_2$ and $\mathbf{b}_1, \mathbf{b}_2$ are learnable parameters, $\sigma(\cdot)$ denotes a nonlinear activation function.

Importantly, while the order-relaxed aggregation reduces sensitivity to packet reordering within a single view, it does not by itself enforce consistency across different partial views of the same original flow. This motivates the subsequent Partial Observation Consistency (POC) learning module, which explicitly aligns shared information across views.

E. Partial Observation Consistency Learning (POC) Module

In § III-D, we obtain flow-level representations that are more robust to packet reordering via order-relaxed set aggregation. However, under dynamic network conditions, the same underlying flow often gives rise to incomplete and inconsistent partial observations across different views. In few-packet settings, global consistency enforcement across views may introduce noisy supervision due to unobservable or misaligned components.

To address this issue, we propose Partial Observation Consistency (POC), which restricts consistency learning to jointly observable evidence and adaptively controls prediction-level alignment based on shared information.

*1) **Overlap-only Representation Consistency**:*

Reliable representation alignment exists only for packets that are simultaneously observable and alignable across views. For the reference view $\tilde{X}^{(0)}$ and the v-th perturbed view $\tilde{X}^{(v)}$, we use the overlap set $\Omega^{(0,v)}$ defined in (Eq. 3) to extract the corresponding packet representations. The overlap-level representation of view v is computed as:

$$\bar{\mathbf{h}}_\Omega^{(v)} = \frac{1}{|\Omega^{(0,v)}|} \sum_{i \in \Omega^{(0,v)}} \hat{\mathbf{h}}_{\pi_v(i)}^{(v)} \tag{14}$$

where $\hat{\mathbf{h}}_{\pi_v(i)}^{(v)}$ denotes the contextualized representation of the shared packet x_i in view v. We then impose representation consistency between the reference and perturbed views using cosine distance:

$$\mathcal{L}_{\mathrm{emb}}^{(v)} = 1 - \cos\big(\bar{\mathbf{h}}_\Omega^{(0)},\ \bar{\mathbf{h}}_\Omega^{(v)}\big) \tag{15}$$

This design treats shared packets as consistency anchors across views, avoiding unreliable supervision from unobservable or misaligned packets, and thus learning representations that are more robust to packet loss and reordering.

*2) **Overlap-gated Logits Consistency**:*

Compared to representation alignment, prediction-level consistency is more sensitive to noisy supervision. When shared evidence across views is insufficient, forcibly aligning prediction distributions may amplify the uncertainty introduced by partial observations and degrade model stability. To address this issue, we introduce an evidence-aware gating mechanism at the prediction level, which adaptively adjusts the strength of prediction consistency based on the shared packet ratio.

Let $\tilde{p}^{(0)}$ and $\mathbf{p}^{(v)}$ denote the prediction distributions of the EMA teacher on the reference view and the student on the v-th view, respectively. The prediction consistency loss is defined as:

$$\mathcal{L}_{\mathrm{logits}}^{(v)} = w\Big(r^{(0,v)}\Big) \cdot D\Big(\tilde{p}^{(0)},\, p^{(v)}\Big) \tag{16}$$

The teacher parameters are updated via exponential moving averaging of the student parameters. Using the EMA prediction as the reference anchor stabilizes the consistency target under partial observations while preserving the overlap-gated learning mechanism. $D(\cdot,\cdot)$ measures the discrepancy between prediction distributions, and the gating weight is defined as:

$$w(r) = \mathbb{I}(r \geq \tau) \cdot r^\gamma \tag{17}$$

Here, τ denotes the minimum overlap ratio threshold, and γ controls the decay rate of consistency strength as the amount of shared evidence decreases. This mechanism ensures that prediction consistency is strengthened only when sufficient shared evidence exists, while being automatically weakened or disabled under severe partial observability, thereby effectively suppressing noisy supervision.

*3) **Overall Objective**:*

By combining the above loss terms, the overall training objective is defined as:

$$\mathcal{L} = \mathcal{L}_{\mathrm{cls}} + \lambda_{\mathrm{emb}} \sum_{v=1}^{V-1} \mathcal{L}_{\mathrm{emb}}^{(v)} + \lambda_{\mathrm{logits}} \sum_{v=1}^{V-1} \mathcal{L}_{\mathrm{logits}}^{(v)} \tag{18}$$

$$\mathcal{L}_{\mathrm{cls}} = \mathrm{CE}\big(f_\theta(\tilde{X}^{(0)}), y\big) \tag{19}$$

where $f_\theta(\cdot)$ denotes the classifier built upon the flow representations described in § III-D, and y is the ground-truth label.$\mathcal{L}_{\mathrm{cls}}$ is applied only to the reference view $\tilde{X}^{(0)}$. λ_{emb} and $\lambda_{\mathrm{logits}}$ balance the two consistency terms. During training, the reference view provides the supervised classification target, while perturbed views provide partial observations for consistency regularization.

By explicitly restricting consistency learning to the alignable partial observation subsets, and adaptively controlling the strength of prediction-level consistency in an evidence-aware manner, POC learns flow representations that are robust to dynamic network perturbations without introducing unobservable noisy supervision. Combined with the order-relaxed flow representations in § III-D, this module forms a unified framework for few-packet traffic classification under dynamic network environments, and constitutes one of the core methodological contributions of this work.

IV. Evaluation

A. Evaluation Setup

Datasets and Pre-processing: We evaluate POC-FPC on four encrypted traffic benchmarks: ISCX-VPN, ISCX-NonVPN [30], ISCX-Tor [31], and CIC-IoT [32]. Bidirectional flows are reconstructed from raw pcap files using the five-tuple, and each original flow is associated with a single class label. To avoid segment-level leakage, we perform stratified splitting at the original-flow level with a 9:1 train and test ratio, ensuring that all K-packet samples derived from the same flow remain in the same split. For few-packet construction, each flow is converted into fixed-budget K-packet samples. We report $K = 6, 10, 15, 20$, covering strict to relatively relaxed few-packet settings; although accurate classification may also be possible with $3 < K < 6$, we focus on these representative budgets due to space constraints and leave more fine-grained packet-budget analysis for future work. For each packet, fixed-length header and payload byte sequences are extracted with truncation or padding and mapped to $[0, 255]$.

Implementation Details: All experiments are implemented in PyTorch on a single RTX 3080. The packet encoder employs 128-dimensional byte embeddings and two 1D convolutional layers, yielding 256-dimensional packet features, which are aggregated by an attention-based set module to form flow representations. Training uses AdamW ($lr = 1e-3$, weight decay$= 1e-4$), batch size 64, for 40 epochs with linear warmup (10%) and cosine decay. Moderate perturbations are applied during training, including Bernoulli packet loss and local reordering. The consistency weights are set to 0.5, overlap gating uses $\tau = 0.2$ and $\gamma = 1.2$, and EMA with decay 0.999 is adopted.

Baselines and Metrics: We compare POC-FPC with representative few-packet baselines, including FS-Net [23], TaTic [8], ET-BERT [12], and TFE-GNN [14]. Under dynamic network perturbations, we further incorporate FlowPic [9], MH-Net [13], Rosetta [1], and AN-Net [19]. All baselines are retrained under the same K-packet budget, flow-level split, and perturbation pipeline. We report Accuracy and Macro-F1 for classification performance. To evaluate cross-view stability, Agreement measures label consistency across partial observations of the same flow, while Mean-JS measures the Jensen–Shannon divergence between predictive distributions.

B. Comparison Experiments (RQ1)

1) Robustness to Packet Loss and Reordering in Few-Packet Classification:

Dynamic network perturbations alter observable packet sets and ordering, causing the same flow to appear as inconsistent partial observations. We therefore evaluate performance under both clean conditions and dynamic perturbations (packet loss and reordering) across four datasets, as summarized in Tables I-IV.

Performance under Clean Observations: Under clean observations, POC-FPC achieves strong few-packet classification performance across all datasets and packet budgets. Compared with sequence-based or prefix-dependent baselines such as FS-Net and TaTic, POC-FPC shows a clear advantage, especially at the strict budget $K = 6$. This indicates that the proposed packet-to-flow representation can extract more stable semantics from limited packets, rather than relying on a longer packet prefix.

Performance under Dynamic Perturbations: Under dynamic perturbations, traditional few-packet baselines exhibit substantial degradation, particularly at $K = 6$. On ISCX-VPN and ISCX-Tor, FS-Net and TaTic incur accuracy drops of approximately 10%-14%, indicating strong sensitivity to packet visibility and order changes.

Robustness-oriented models reduce degradation but their performance is still less stable across datasets and packet budgets. In contrast, POC-FPC limits degradation to a narrow range and consistently achieves the highest perturbed accuracy across datasets. Even at $K = 6$ where several baselines decline toward 85% accuracy, POC-FPC maintains performance above 92% on most datasets. These results demonstrate that overlap-aware consistency learning effectively stabilizes predictions under packet loss and reordering.

2) Cross-View Consistency under Partial Observations:

We evaluate decision stability under inconsistent partial observations using cross-view Agreement and Mean-JS (Fig. 2). In comparison with few-packet baselines, Fig. 2 shows that although stability improves slightly as K increases from 6 to 20, substantial disparities persist. POC-FPC consistently maintains Agreement at approximately 90%, with Mean-JS confined to 0.07, exhibiting minimal variation across packet budgets. In contrast, FS-Net achieves Agreement as low as 70% at small K, accompanied by Mean-JS values of 0.34, indicating high sensitivity to packet visibility and ordering changes. TaTic, ET-BERT, and TFE-GNN demonstrate improved stability relative to FS-Net; however, their Agreement typically ranges between 75%-86%, with Mean-JS remaining within 0.14-0.27, suggesting that reliance on limited prefix information constrains cross-view decision stability. Overall, POC-FPC achieves consistently higher label consistency and lower distribution divergence across packet budgets.

Compared with robustness-oriented models, POC-FPC maintains a clear stability advantage. FlowPic attains Agreement close to 94% at $K = 15$, yet its Mean-JS remains around 0.1. AN-Net reduces Mean-JS to approximately 0.10-0.15, with Agreement approaching 90%, while MH-Net exhibits higher divergence and lower Agreement. By contrast, POC-FPC simultaneously achieves the highest Agreement and the lowest Mean-JS across all settings, demonstrating different perturbed views of the same flow are more likely to produce both the same label and similar prediction distributions.

C. Ablation Study (RQ2)

In this section, we conduct an ablation study of POC-FPC on the ISCX-VPN and ISCX-NonVPN datasets, and show the experimental results in Table V.

1) Contribution Analysis of the Partial Observation Consistency (POC) Mechanism:

TABLE I
CLASSIFICATION PERFORMANCE UNDER STATIC AND DYNAMIC (PERTURBED-FUSED) CONDITIONS ON ISCX-VPN DATASET ($K = 6/10/15/20$).

Method	ISCX-VPN															
	$K=6$				$K=10$				$K=15$				$K=20$			
	clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
FS-Net	0.6797	0.6662	0.5381	0.5028	0.7234	0.7145	0.6034	0.6293	0.7398	0.7277	0.6327	0.6018	0.7182	0.7024	0.5999	0.6023
TaTic	0.9050	0.9156	0.7974	0.7613	0.9302	0.9246	0.8126	0.8032	0.9433	0.9545	0.8041	0.7935	0.9587	0.9375	0.7926	0.8139
ET-BERT	0.8840	0.8769	0.8065	0.8246	0.9332	0.9163	0.8324	0.7900	0.9442	0.9435	0.8072	0.8103	0.9526	0.9484	0.8414	0.8070
TFE-GNN	0.9464	0.9370	0.8994	0.9079	0.9430	0.9209	0.9062	0.8950	0.9593	0.9448	0.8990	0.9043	0.9507	0.9385	0.9096	0.8982
FlowPic	0.8490	0.8173	0.8025	0.7945	0.8574	0.8026	0.8183	0.8251	0.9067	0.9364	0.8747	0.8974	0.9272	0.9271	0.9005	0.9164
MH-Net	0.7588	0.7563	0.6138	0.6029	0.7409	0.7687	0.7016	0.6973	0.7902	0.7841	0.7478	0.7051	0.7830	0.8176	0.7563	0.7393
Rosetta	0.9229	0.9062	0.8978	0.8875	0.9388	0.9108	0.9094	0.8921	0.9596	0.9529	0.9306	0.9115	0.9620	0.9584	0.9399	0.9273
AN-Net	0.9240	0.9164	0.9066	0.9123	0.9346	0.9389	0.9187	0.9205	0.9328	0.9295	0.9267	0.9129	0.9409	0.9242	0.9274	0.9087
POC-FPC	0.9866	0.9704	0.9557	0.9378	0.9724	0.9587	0.9560	0.9381	0.9860	0.9708	0.9494	0.9290	0.9781	0.9527	0.9520	0.9263

TABLE II
CLASSIFICATION PERFORMANCE UNDER STATIC AND DYNAMIC (PERTURBED-FUSED) CONDITIONS ON ISCX-NONVPN DATASET ($K = 6/10/15/20$).

Method	ISCX-nonVPN															
	$K=6$				$K=10$				$K=15$				$K=20$			
	clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
FS-Net	0.5624	0.5510	0.5091	0.4819	0.6544	0.6440	0.6133	0.6028	0.7326	0.7055	0.6711	0.6393	0.7591	0.7529	0.6829	0.6571
TaTic	0.8019	0.7996	0.6671	0.6259	0.8111	0.8041	0.6740	0.5999	0.8456	0.8240	0.7642	0.7127	0.8702	0.8790	0.7871	0.8051
ET-BERT	0.7909	0.8079	0.7223	0.6910	0.8360	0.8235	0.7551	0.6851	0.9167	0.9235	0.8636	0.7776	0.9214	0.9294	0.8530	0.8076
TFE-GNN	0.9061	0.8898	0.8053	0.7977	0.8309	0.8532	0.8099	0.8000	0.8496	0.8549	0.7823	0.7962	0.8376	0.8243	0.8011	0.8167
FlowPic	0.8164	0.7746	0.7572	0.7098	0.8192	0.8073	0.7972	0.7791	0.8356	0.8284	0.8082	0.7935	0.8462	0.8390	0.7925	0.7828
MH-Net	0.6707	0.6534	0.5732	0.5659	0.7084	0.7049	0.6287	0.5963	0.7302	0.7413	0.7091	0.6902	0.7927	0.7641	0.7320	0.7492
Rosetta	0.8527	0.8304	0.8098	0.8119	0.8599	0.8482	0.7983	0.8084	0.8608	0.8431	0.8397	0.8526	0.8632	0.8540	0.8276	0.8399
AN-Net	0.8356	0.8067	0.8183	0.8196	0.8462	0.8266	0.8009	0.7994	0.8499	0.8503	0.8340	0.8073	0.8689	0.8627	0.8373	0.8183
POC-FPC	0.8832	0.8593	0.8718	0.8565	0.8864	0.8614	0.8796	0.8642	0.8885	0.8767	0.8784	0.8591	0.8798	0.8530	0.8769	0.8621

TABLE III
CLASSIFICATION PERFORMANCE UNDER STATIC AND DYNAMIC (PERTURBED-FUSED) CONDITIONS ON ISCX-TOR DATASET ($K = 6/10/15/20$).

Method	ISCX-Tor															
	$K=6$				$K=10$				$K=15$				$K=20$			
	clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
FS-Net	0.6932	0.6539	0.5382	0.5499	0.7592	0.6711	0.6524	0.6092	0.8286	0.7242	0.7465	0.7340	0.8373	0.7410	0.7320	0.7205
TaTic	0.9271	0.9072	0.7854	0.7900	0.9346	0.9182	0.8187	0.7975	0.9543	0.9403	0.7811	0.8093	0.9480	0.9465	0.7561	0.7990
ET-BERT	0.8926	0.8572	0.7833	0.7794	0.9035	0.8900	0.8022	0.7847	0.9421	0.9397	0.8300	0.7991	0.9696	0.9347	0.8528	0.8031
TFE-GNN	0.9167	0.9212	0.8049	0.8100	0.9287	0.9034	0.7638	0.7627	0.9586	0.9355	0.7412	0.7390	0.9483	0.9479	0.8132	0.7934
FlowPic	0.7706	0.7945	0.7368	0.7615	0.7954	0.7517	0.7507	0.7302	0.8010	0.7954	0.6827	0.6648	0.8263	0.8478	0.7940	0.7727
MH-Net	0.8627	0.8294	0.6787	0.6597	0.8681	0.8323	0.7406	0.7210	0.8582	0.8147	0.8283	0.7823	0.8790	0.8483	0.8313	0.8050
Rosetta	0.9472	0.9316	0.8916	0.9083	0.9429	0.9248	0.8807	0.8995	0.9374	0.9376	0.9076	0.8991	0.9565	0.9375	0.9078	0.9173
AN-Net	0.9580	0.9339	0.8826	0.9099	0.9563	0.9538	0.9284	0.9173	0.9752	0.9635	0.9233	0.9072	0.9863	0.9884	0.9413	0.9432
POC-FPC	0.9759	0.9509	0.9473	0.9298	0.9724	0.9423	0.9417	0.9268	0.9675	0.9453	0.9307	0.9154	0.9635	0.9358	0.9224	0.9279

Table V shows that the POC design is essential for maintaining both discriminative performance and cross-view stability. Training only on clean observations without MVOG leads to clear degradation. On ISCX-VPN, Agreement drops by about 8% and Mean-JS more than doubles, showing that clean training alone cannot handle inconsistent partial observations caused by packet loss and reordering. When MVOG is retained but consistency learning is removed, the model still suffers from low Agreement and high Mean-JS, indicating that multi-view generation by itself is insufficient without explicit cross-view alignment.

When consistency is enforced globally without restricting to overlapping packets (w/o Overlap-only), Agreement further declines (over 4% on VPN) and divergence increases substantially, confirming that aligning non-overlapping components introduces noisy supervision. Likewise, removing logits gating or embedding-level consistency leads to consistent stability deterioration. These results collectively verify that overlap-aware weighting combined with dual-level consistency is essential for maintaining cross-view-aligned and perturbation-invariant representations under dynamic partial observations.

2) Contribution Analysis of the Few-Packet Flow Representation Module:

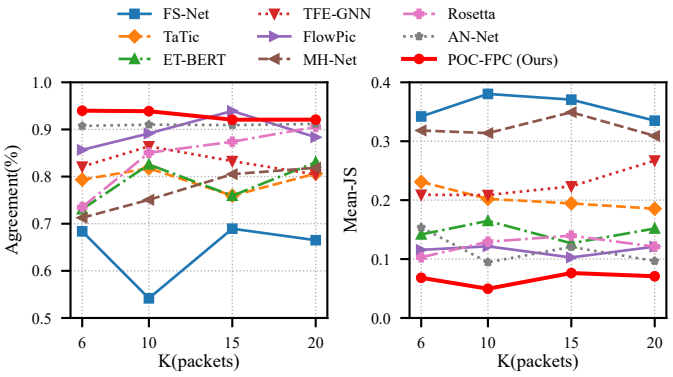
The representation ablations further validate the role of order-relaxed set aggregation. When set aggregation is removed and only the GRU branch is used, accuracy and

TABLE IV
CLASSIFICATION PERFORMANCE UNDER STATIC AND DYNAMIC (PERTURBED-FUSED) CONDITIONS ON CIC-IOT DATASET ($K = 6/10/15/20$).

Method	CIC-IoT															
	$K = 6$				$K = 10$				$K = 15$				$K = 20$			
	clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused		clean(view0)		perturbed-fused	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
FS-Net	0.6838	0.6903	0.5603	0.5399	0.6995	0.6767	0.6145	0.5811	0.7005	0.7059	0.6597	0.6743	0.7682	0.7386	0.6320	0.6019
TaTic	0.9335	0.8910	0.8176	0.7785	0.9382	0.9182	0.8264	0.7973	0.9322	0.9017	0.7928	0.7594	0.9518	0.9611	0.8501	0.8153
ET-BERT	0.8907	0.8718	0.8451	0.8500	0.9095	0.8941	0.8637	0.8426	0.8865	0.8632	0.8362	0.8012	0.9370	0.9423	0.8900	0.8875
TFE-GNN	0.9510	0.9291	0.8603	0.8599	0.9518	0.9427	0.8645	0.8511	0.9699	0.9566	0.8970	0.9143	0.9698	0.9576	0.9220	0.9319
FlowPic	0.6867	0.6956	0.5712	0.5945	0.6256	0.6016	0.5976	0.6088	0.6301	0.6196	0.6059	0.6314	0.6338	0.6407	0.6079	0.6289
MH-Net	0.8126	0.8182	0.7631	0.6417	0.8000	0.8138	0.7627	0.6927	0.8342	0.8451	0.6927	0.7145	0.8048	0.8174	0.7876	0.7563
Rosetta	0.9450	0.9003	0.9198	0.8927	0.9399	0.9128	0.9104	0.9114	0.9783	0.9560	0.9516	0.9241	0.9782	0.9895	0.9276	0.9009
AN-Net	0.9274	0.8937	0.9027	0.9094	0.9378	0.9395	0.9072	0.9109	0.9489	0.9382	0.8926	0.9084	0.9471	0.9400	0.9352	0.9467
POC-FPC	0.9854	0.9690	0.9660	0.9500	0.9864	0.9665	0.9492	0.9365	0.9843	0.9624	0.9474	0.9212	0.9857	0.9622	0.9456	0.9345

TABLE V
ABLATION STUDY OF KEY COMPONENTS IN POC-FPC ON ISCX-VPN AND ISCX-NONVPN: CLASSIFICATION PERFORMANCE AND FLOW-LEVEL STABILITY UNDER DYNAMIC PERTURBATIONS.

Method	ISCX-VPN				ISCX-NonVPN			
	Acc↑	F1↑	Agreement↑	Mean-JS↓	Acc↑	F1↑	Agreement↑	Mean-JS↓
Full (POC-FPC)	0.9860	0.9700	0.9210	0.0627	0.8918	0.8865	0.9484	0.0315
Ablations on Partial Observation Consistency (POC)								
w/o MVOG (Train-clean)	0.9173	0.9026	0.8382	0.1335	0.7382	0.7372	0.8736	0.1284
w/o Consistency (Sup. only, view0)	0.9244	0.9200	0.8496	0.1236	0.7553	0.7499	0.8904	0.1027
w/o Overlap-only (Global Consistency)	0.9576	0.9364	0.8803	0.1084	0.8775	0.8573	0.9247	0.0512
w/o Logits Gate (Ungated logits consistency)	0.9764	0.9659	0.9050	0.0809	0.8802	0.8534	0.9371	0.0434
w/o Embed Consistency (Logits-only)	0.9429	0.9238	0.8937	0.0997	0.8523	0.8206	0.9025	0.0877
Ablations on Few-Packet Flow Representation								
w/o Set Aggregation (GRU branch only)	0.9391	0.9117	0.8697	0.1098	0.8627	0.8501	0.9220	0.0396
Avg Fusion (Set+GRU average)	0.9446	0.9264	0.8994	0.0860	0.8702	0.8551	0.9372	0.0404



(a) Cross-view Agreement(↑) (b) Cross-view JS(↓)

Fig. 2. Flow-level cross-view stability under dynamic perturbations.

Agreement decrease on both datasets. This indicates that strict sequential modeling is sensitive to packet loss and reordering, since the hidden-state trajectory changes with packet visibility and order. Simple average fusion of the set and GRU branches does not provide consistent improvement and even causes a larger stability drop on ISCX-VPN. This suggests that combining heterogeneous branches without partial-observation-aware alignment may introduce inconsistent signals. Overall, set-based aggregation and POC learning are complementary: the former reduces order dependence within each view, while the latter aligns reliable shared evidence across views.

D. Model Sensitivity Analysis (RQ3)

1) Sensitivity to Perturbation Strength:

We analyze the sensitivity of POC-FPC on ISCX-VPN under the strict few-packet setting $K = 6$, considering perturbation strength and key consistency-related hyperparameters.

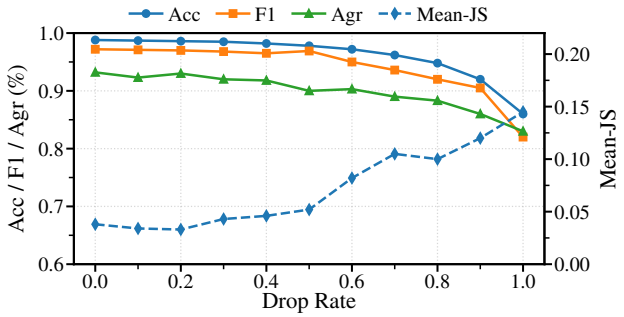
(i) Sensitivity to Drop Rate. As shown in Fig. 3(a), as the packet loss rate increases, Accuracy, F1, and Agreement gradually decrease, while Mean-JS increases. POC-FPC remains stable under moderate loss, but severe loss removes substantial observable evidence and causes clear degradation.

(ii) Sensitivity to Reorder Window. Fig. 3(b) shows that increasing the reorder window leads to milder performance drops, with Mean-JS increasing only moderately. This indicates that order-relaxed set aggregation effectively reduces sensitivity to packet reordering, whereas severe packet loss is more harmful because it directly reduces available semantic evidence.

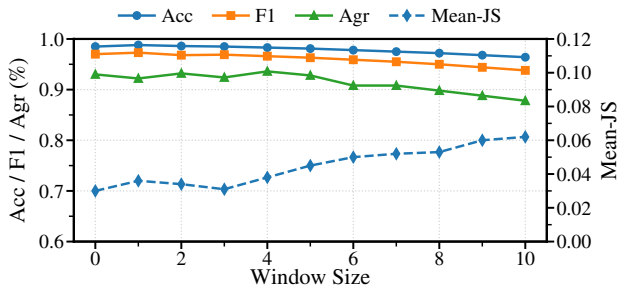
2) Sensitivity to Overlap-Gated Logits Consistency:

As shown in Fig. 4(a), varying the overlap threshold τ shows that moderate gating improves both discrimination and stability: performance peaks around $\tau \approx 0.2$, where Acc and F1 are slightly higher and Mean-JS remains low. When τ increases beyond 0.4, both Agreement and accuracy decline and divergence rises, indicating that overly strict gating weakens effective cross-view regularization.

3) Sensitivity to Consistency Weight:



(a) Drop Rate



(b) Reorder Window

Fig. 3. Sensitivity to Perturbation Strength.

As shown in Fig. 4(b), increasing the consistency weight λ from 0 to 0.1 yields noticeable gains in Acc, F1 and Agreement while reducing Mean-JS, demonstrating that moderate alignment mitigates perturbation-induced drift. Further increasing $\lambda(\geq 0.3)$ leads to gradual performance and stability degradation, suggesting that excessive consistency over-constrains discriminative representations.

4) Sensitivity to Number of Training Views:

Fig. 4(c) shows that increasing the number of training views improves stability initially, but the gain saturates when too many views are used. This suggests that moderate multi-view coverage is sufficient to capture common packet loss and reordering patterns without unnecessary training overhead.

E. Efficiency and Computational Cost Analysis (RQ4)

POC-FPC remains lightweight under the proposed framework. It contains 129.7K trainable parameters with a model size of 0.49MB, indicating that the packet-to-flow representation and partial-observation consistency design introduce limited parameter overhead. This compactness is important for few-packet traffic classification, where early prediction and resource-efficient deployment are often required.

Under the main three-view training setting with $K = 6$, POC-FPC takes 36.37 seconds per epoch, achieves 406.23 flows/s throughput, and uses 136.94MB peak allocated GPU memory. Step-level profiling shows that three-view training requires 70.26 ms per step, about 2.42 times the cost of single-view training. The extra cost mainly comes from generating multiple partial observations and performing multi-view forward/backward computation, while the overlap-aware consistency loss itself remains lightweight.

During inference, POC-FPC offers a controllable efficiency–robustness trade-off. Single-view inference requires only 0.124 ms per flow with 8060.82 flows/s throughput and 24.80MB peak memory. Three-view fusion increases the profiled latency to 0.872 ms per flow and reduces throughput to 1146.40 flows/s, while peak memory remains low at 25.36MB. It also maintains stable predictions under perturbations, achieving 95.95% accuracy, 93.36% Macro-F1, 91.51% Agreement, and 0.0471 Mean-JS. These results suggest that the multi-view mechanism introduces manageable overhead while improving robustness to packet loss and reordering.

V. Conclusion

This paper studies few-packet traffic classification under dynamic network conditions, where packet loss and reordering lead to inconsistent partial observations of the same flow. We formalize this problem as Partial Observation Consistency (POC) and propose POC-FPC, which integrates multi-view observation generation, order-relaxed packet-to-flow representation, and overlap-aware consistency learning. By aligning only shared and reliable evidence, POC-FPC reduces noisy supervision from unobservable packets and improves prediction stability under perturbations.

Experiments on multiple encrypted traffic datasets show that POC-FPC achieves robust classification performance across diverse packet loss and reordering settings. Ablation and sensitivity analyses further verify the necessity of selective consistency and order-relaxed aggregation. Overall, this work improves the robustness of few-packet traffic classification under dynamic network conditions and provides a principled basis for extending partial-observation inconsistency modeling to complex perturbations and cross-environment generalization.

VI. Acknowledgment

This work was funded by the National Key R&D Program of China under Grant 2024YFB2906900, the Beijing Nova Program under Grant 2023140, the National Natural Science Foundation of China for Youth Science Foundation Project (Class B) under Grant 62522203, and the Key Program of the Beijing Natural Science Foundation (Haidian Original Innovation Joint Fund) under Grant L252013.

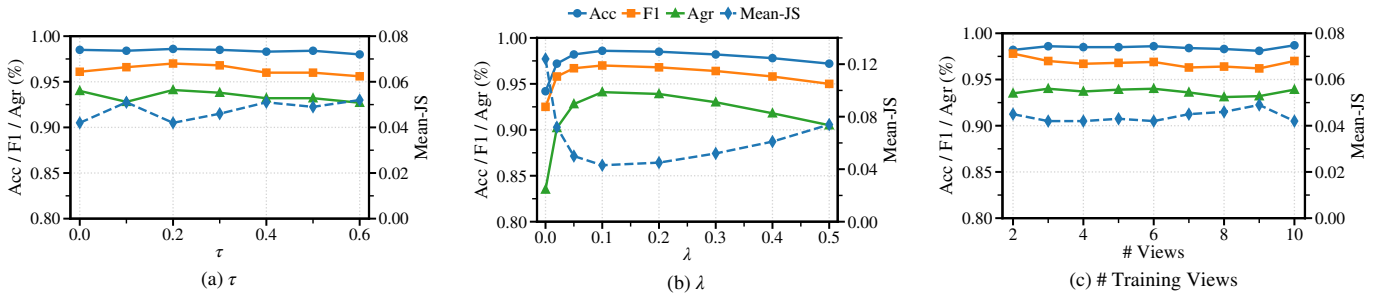
References

[1] R. Xie, J. Cao, E. Dong, M. Xu, K. Sun, Q. Li, L. Shen, and M. Zhang, "Rosetta: Enabling robust TLS encrypted traffic classification in diverse network environments with TCP-Aware traffic augmentation," in *32nd USENIX Security Symposium (USENIX Security 23)*, Anaheim, CA, 2023, pp. 625–642.

[2] L. Yang, Y. Wang, L. Liu, J. Huang, and S. Fu, "Expmd: an explainable framework for traffic identification based on multi-domain features," in *2024 IEEE/ACM 32nd International Symposium on Quality of Service (IWQoS)*, 2024, pp. 1–10.

[3] Y. Zhao, G. Dettori, M. Boffa, L. Vassio, and M. Mellia, "The sweet danger of sugar: Debunking representation learning for encrypted traffic classification," in *Proceedings of the ACM SIGCOMM 2025 Conference*, 2025, pp. 296–310.

[4] J. Wang, Y. Zhang, and W. Wang, "Resolving congestion packet losses for small flows in datacenter networks with link-local retransmission," in *2025 34th International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2025, pp. 1–9.



(a) τ (b) λ (c) # Training Views

Fig. 4. Sensitivity analysis of POC-FPC to key consistency hyperparameters: (a) τ (Overlap Gate Threshold), (b) λ (Consistency Weight), and (c) # Training Views.

[5] J. Sherry, P. X. Gao, S. Basu, A. Panda, A. Krishnamurthy, C. Maciocco, M. Manesh, J. Martins, S. Ratnasamy, L. Rizzo *et al.*, "Rollback-recovery for middleboxes," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, 2015, pp. 227–240.

[6] D.-Y. Yu, Y. Zhang, X. Wang, J. Wang, W. Wu, H. Yao, W. Wang, and K. Xu, "Lcmp: Distributed long-haul cost-aware multi-path routing for inter-datacenter rdma networks," in *Proceedings of the 21st European Conference on Computer Systems*, ser. EUROSYS '26, New York, NY, USA, 2026, p. 862–878.

[7] J. Huang, W. Lyu, W. Li, J. Wang, and T. He, "Mitigating packet reordering for random packet spraying in data center networks," *IEEE/ACM Transactions on Networking*, vol. 29, no. 3, pp. 1183–1196, 2021.

[8] Y. Wang, H. He, Y. Lai, and A. X. Liu, "A two-phase approach to fast and accurate classification of encrypted traffic," *IEEE/ACM Transactions On Networking*, vol. 31, no. 3, pp. 1071–1086, 2022.

[9] T. Shapira and Y. Shavitt, "Flowpic: A generic representation for encrypted traffic classification and applications identification," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1218–1232, 2021.

[10] R. J. Babaria, M. Lyu, G. Batista, and V. Sivaraman, "Fastflow: Early yet robust network flow classification using the minimal number of time-series packets," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 9, no. 2, pp. 1–27, 2025.

[11] O. Salman, I. H. Elhajj, A. Chehab, and A. Kayssi, "Towards efficient real-time traffic classifier: A confidence measure with ensemble deep learning," *Computer Networks*, vol. 204, p. 108684, 2022.

[12] X. Lin, G. Xiong, G. Gou, Z. Li, J. Shi, and J. Yu, "Et-bert: A contextualized datagram representation with pre-training transformers for encrypted traffic classification," in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 633–642.

[13] H. Zhang, H. Yue, X. Xiao, L. Yu, Q. Li, Z. Ling, and Y. Zhang, "Revolutionizing encrypted traffic classification with mh-net: A multi-view heterogeneous graph model," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025, pp. 1048–1056.

[14] H. Zhang, L. Yu, X. Xiao, Q. Li, F. Mercaldo, X. Luo, and Q. Liu, "Tfe-gnn: A temporal fusion encoder using graph neural networks for fine-grained encrypted traffic classification," in *Proceedings of the ACM web conference 2023*, 2023, pp. 2066–2075.

[15] S. Cui, X. Han, D. Han, Z. Wang, W. Wang, B. Jiang, B. Liu, and Z. Lu, "Fg-sat: Efficient flow graph for encrypted traffic classification under environment shifts," *IEEE Transactions on Information Forensics and Security*, 2025.

[16] G. Zhou, X. Guo, Z. Liu, T. Li, Q. Li, and K. Xu, "Trafficformer: an efficient pre-trained model for traffic data," in *2025 IEEE symposium on security and privacy (SP)*. IEEE, 2025, pp. 1844–1860.

[17] S. Yang, X. Zheng, J. Li, J. Xu, X. Wang, and E. C. Ngai, "Recda: Concept drift adaptation with representation enhancement for network intrusion detection," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 3818–3828.

[18] X. Deng, Q. Li, and K. Xu, "Robust and reliable early-stage website fingerprinting attacks via spatial-temporal distribution analysis," in *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, 2024, pp. 1997–2011.

[19] X. Deng, Y. Wang, and Z. Xue, "An-net: an anti-noise network for anonymous traffic classification," in *Proceedings of the ACM Web Conference 2024*, 2024, pp. 4417–4428.

[20] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatian, "Traffic classification on the fly," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 2, pp. 23–26, 2006.

[21] L. Peng, B. Yang, and Y. Chen, "Effective packet number for early stage internet traffic identification," *Neurocomputing*, vol. 156, pp. 252–267, 2015.

[22] X. Wang, Y. Wang, Y. Lai, Z. Hao, and A. X. Liu, "Reliable open-set network traffic classification," *IEEE Transactions on Information Forensics and Security*, 2025.

[23] C. Liu, L. He, G. Xiong, Z. Cao, and Z. Li, "Fs-net: A flow sequence network for encrypted traffic classification," in *IEEE INFOCOM 2019-IEEE Conference On Computer Communications*. IEEE, 2019, pp. 1171–1179.

[24] X. Xiao, S. Wang, G. Hu, Q. Li, K. Mao, X. Luo, B. Zhang, and S. Xia, "Rbljan: Robust byte-label joint attention network for network traffic classification," *IEEE Transactions on Dependable and Secure Computing*, 2024.

[25] Z. Diao, G. Xie, X. Wang, R. Ren, X. Meng, G. Zhang, K. Xie, and M. Qiao, "Ec-gcn: A encrypted traffic classification framework based on multi-scale graph convolution networks," *Computer Networks*, vol. 224, p. 109614, 2023.

[26] X. Qiu, G. Cheng, W. Zhu, D. Niu, and N. Fu, "Dual-channel interactive graph transformer for traffic classification with message-aware flow representation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 1, 2025, pp. 685–693.

[27] X.-Y. Chen, L. Han, D.-C. Zhan, and H.-J. Ye, "Miett: Multi-instance encrypted traffic transformer for encrypted traffic classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 15, 2025, pp. 15 922–15 929.

[28] S. Li, Z. Gan, M. Xiao, P. Hu, X. Cheng, C. Wang, and F. Li, "Mota: Mixture of traffic agents for robust network traffic classification," in *2025 IEEE/ACM 33rd International Symposium on Quality of Service (IWQoS)*. IEEE, 2025, pp. 1–10.

[29] Y. Qing, Q. Yin, X. Deng, X. Zhang, P. Li, Z. Liu, K. Sun, K. Xu, and Q. Li, "Training robust classifiers for classifying encrypted traffic under dynamic network conditions," in *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*, 2025, pp. 3564–3578.

[30] G. D. Gil, A. H. Lashkari, M. Mamun, and A. A. Ghorbani, "Characterization of encrypted and vpn traffic using time-related features," in *Proceedings of the 2nd international conference on information systems security and privacy (ICISSP 2016)*. SciTePress Setúbal, Portugal, 2016, pp. 407–414.

[31] A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of tor traffic using time based features," in *International conference on information systems security and privacy*, vol. 2. SciTePress, 2017, pp. 253–262.

[32] S. Dadkhah, H. Mahdikhani, P. K. Danso, A. Zohourian, K. A. Truong, and A. A. Ghorbani, "Towards the development of a realistic multidimensional iot profiling dataset," in *2022 19th Annual International Conference on Privacy, Security & Trust (PST)*. IEEE, 2022, pp. 1–11.