

Improving Network Traffic Classification Using Metadata

Shahar Katz
Tel Aviv University
shahar.katz@outlook.com

Tal Shapira
The Hebrew University
talshapirala@gmail.com

Yaniv Harel
Tel Aviv University
yaniv10@tauex.tau.ac.il

Yuval Shavitt
Tel Aviv University
shavitt@eng.tau.ac.il

Abstract—Internet traffic classification has attracted increasing attention in the past decade, and deep learning has delivered strong classification performance. The significant effort required to obtain labeled datasets has motivated research to focus on few-shot learning, where only a few labeled samples per class are available. FlowPic, a solution based on visualization of the statistics of packet sizes and arrival times, emerged as a leading solution for both supervised and few-shot learning. However, most existing approaches rely solely on per-flow packet statistics, and therefore miss complementary context from the Internet routing layer.

We propose a multimodal traffic-classification framework that augments FlowPic-based representations with the destination ASN, without requiring additional traffic labels. We fuse the BGP2Vec ASN embedding with FlowPic representations within a contrastive-learning pipeline (SimCLR) that leverages domain-specific traffic augmentations, enabling the model to learn a joint representation that captures both flow dynamics and server context.

Experiments on the UNB ISCX dataset show that incorporating ASN context consistently improves classification accuracy and yields an average gain of 8–12 percentage points in few-shot regimes compared to prior FlowPic-based baselines, reaching over 90% accuracy with only 10 shots per label. The improvement remains statistically significant across the evaluated shot counts.

Index Terms—Traffic Classification, BGP, AS embedding, Few-shot learning, contrastive representation learning.

I. INTRODUCTION

Accurate classification of network traffic, especially encrypted traffic, remains a central problem for modern networks. Traffic classification supports Quality of Service (QoS) management and strengthens security monitoring by enabling early detection of anomalies and policy violations. However, the widespread adoption of encryption, VPNs, and anonymity systems has reduced the effectiveness of content-based approaches such as Deep Packet Inspection (DPI), shifting attention toward methods that infer application types from observable flow behavior.

A large body of work therefore relies on statistical features extracted from packet headers and timing patterns [1], [2]. More recently, a particularly effective direction has been to transform flow dynamics into image-like representations and apply convolutional models, borrowing strengths from visual recognition [3]–[8]. Among these, FlowPic has been repeatedly shown to achieve strong performance while remaining simple to implement [7]–[9]. Follow-up work introduced mini-

FlowPics that reduce representation size and training cost without sacrificing accuracy [10], [11].

Despite these advances, high-performing deep models still depend on labeled data. In practice, unlabeled traffic data is relatively accessible, but reliable application labels are scarce, expensive, and labor-intensive to obtain. This has motivated few-shot and self-supervised pipelines that learn transferable representations from unlabeled traffic. In particular, Horowicz et al. [10], [11] showed that contrastive pretraining (e.g., SimCLR [12]) combined with traffic-aware augmentations can substantially improve few-shot performance. Subsequent replication work highlighted the importance of rigorous evaluation protocols and fair comparisons across splits and datasets [13], while broader studies further emphasized that augmentation choices can meaningfully affect results across datasets [14].

A recent study [15] was the first to use *AS-level context* to improve encrypted traffic classification. Kahana et al. [15] used a two-stage solution, where applications with unique ASNs are classified directly, and the remaining applications are directed to per-ASN classifiers. This requires training of many models, one for each ASN identified during training, and an additional classifier for unseen ASNs. In contrast, we inject AS context through *BGP2Vec* embeddings learned from BGP paths, enabling representation-level fusion with FlowPic features and supporting generalization to ASNs that are unseen in the training set. The BGP2Vec embeddings give our multimodal classifiers valuable information even for ASNs that were not seen during training.

A different direction is *label-conditioned* context: FlowPicClip [16] aligns FlowPic representations with textual descriptions derived from the application label to improve classification accuracy. While effective, this approach inherently depends on labeled data to generate the textual supervision signal, and therefore does not directly address the lack of labeled datasets.

In this paper, we argue that combining additional information with flow features greatly enhances the context needed for robust generalization. We propose a multimodal traffic-classification framework that uses FlowPic-based representations together with additional metadata such as ASN, address prefix, or Server Name Indication (SNI). We focus on ASNs because prior work has shown that ASes admit informative unsupervised embeddings learned from BGP routing data, making ASN context a natural, lightweight complement

to FlowPic representations [17]–[19]. We obtain the ASN of each flow’s server IP and represent each ASN using *BGP2Vec*, an unsupervised embedding learned from BGP AS path announcements. We then fuse the ASN embedding with the FlowPic representation in a contrastive-learning pipeline (SimCLR), leveraging domain-specific traffic augmentations to learn a joint representation that captures both flow dynamics and network-location context.

We evaluate the proposed method on the UNB ISCX dataset, a widely used dataset in prior research whose packet captures supply endpoint IPs that can be mapped to ASNs. Our results show that adding ASN context consistently improves classification performance, yielding an average improvement of 8–12 percentage points in few-shot learning over prior baselines.

II. DATASETS

To compare our results with previous work, we use the University of New Brunswick (UNB) *ISCX VPN-nonVPN Traffic Dataset* (ISCX-VPN) [20]. The dataset consists of packet-capture (pcap) files labeled by multiple traffic types (e.g., VoIP, Video) and applications (e.g., Facebook, YouTube), and includes both regular (non-VPN) sessions and sessions captured over a VPN.

A key motivation for using the UNB ISCX dataset in this work is that the raw traffic can be reliably parsed into flows defined by the standard 5-tuple $\{source\ IP, source\ port, destination\ IP, destination\ port, protocol\}$. This representation is particularly useful for our setting, since our method augments each flow with Internet routing context by mapping the relevant endpoint IP(s) to an ASN and learning with the resulting ASN-derived embedding. Another reason we decided to use this dataset is to compare our methods with results from previous studies [8], [11], [13].

Following prior work, we filter the dataset to retain only sessions containing at least 100 packets. After filtering, many applications are represented by only a small number of sessions. Fortunately, the VoIP and video applications in ISCX-VPN are typically recorded over several minutes per session, which allows us to extract multiple *non-overlapping* FlowPic samples from a single session without reusing packets across samples. In our experiments, we construct FlowPics using non-overlapping 15-second windows.

We focus on the standard *VoIP and Video Application Identification* task under *non-VPN* encryption. This benchmark contains 10 classes: four VoIP applications (Facebook, Hangouts, Skype, VoipBuster) and six video applications (Facebook, Hangouts, Netflix, Skype, Vimeo, YouTube). To prevent leakage between training and evaluation, we first split the raw sessions into disjoint training and testing groups (i.e., no session contributes FlowPics to both splits). We then sample FlowPics from these groups such that the training set contains 100 FlowPics per class and the test set contains 30 FlowPics per class.

RouteViews BGP Announcements for BGP2Vec Pre-training: For the BGP2Vec ASN embedding, we used a RouteViews OIX file dated January 13, 2025, containing

approximately 3,600,000 BGP paths, 62,525 AS vertices, and 113,400 undirected links.

III. METHODS

A. Overview

Our goal is to improve encrypted traffic classification by combining packet sizes and packet arrival times, represented as a miniFlowPic, with additional relevant information. In this paper, we concentrate on the example of using the ASN of the destination IP (the server) of the flow.

We represent each traffic instance as a miniFlowPic image (32×32) together with AS-level features obtained from a BGP-trained embedding model. We then train a multimodal classifier that fuses the two modalities into a single latent representation for application identification.

B. Traffic Preprocessing and FlowPic Generation

Flow definition and session processing: We extract traffic flows from packet captures using the standard 5-tuple definition $\{source\ IP, source\ port, destination\ IP, destination\ port, protocol\}$. To ensure sufficiently informative visual representations, we follow prior FlowPic/miniFlowPic practice and discard very short flows when constructing the application-identification dataset.

MiniFlowPic construction FlowPic converts a network flow into a 2D histogram, plotting packet sizes against packet arrival times, thereby capturing flow dynamics in an image representation. We use the compact *miniFlowPic* representation (32×32), which reduces the original FlowPic resolution by aggregating packet size into bins on the y -axis and using fewer and longer time slots on the x -axis; the original FlowPic has a resolution of 1500×1500 , whereas the miniFlowPic has a resolution of 32×32 . This change improves efficiency while maintaining strong performance in prior work [10], [11].

ISCX application-identification protocol For the ISCX VPN-nonVPN application-identification setting, we use the 10-class configuration containing VoIP and video applications, and follow the same protocol as previous studies [8], [11], [16]: (i) filtering sessions to contain at least 100 packets; (ii) extracting multiple *non-overlapping* 15-second FlowPics from longer recordings, with at least 10 packets per flow; and (iii) splitting flows by session to avoid overlap between training and test traffic. We adopt the standard sampling described in Horowicz et al. [10], [11] (100 FlowPics per class for training and 30 FlowPics per class for testing).

C. ASN Embeddings

FlowPic captures packet sizes and arrival behavior, but it does not encode all the data the pcap files offer. To include the ASN as additional context, we associate each flow with AS-level information and embed ASNs into a vector space.

1) Training BGP2Vec on RouteViews: We train an AS embedding model using BGP routing data, following the BGP2Vec methodology [17], [18]. In this approach, BGP paths are treated as sequences of ASNs, and an AS-level graph is constructed from the observed paths. The BGP routes

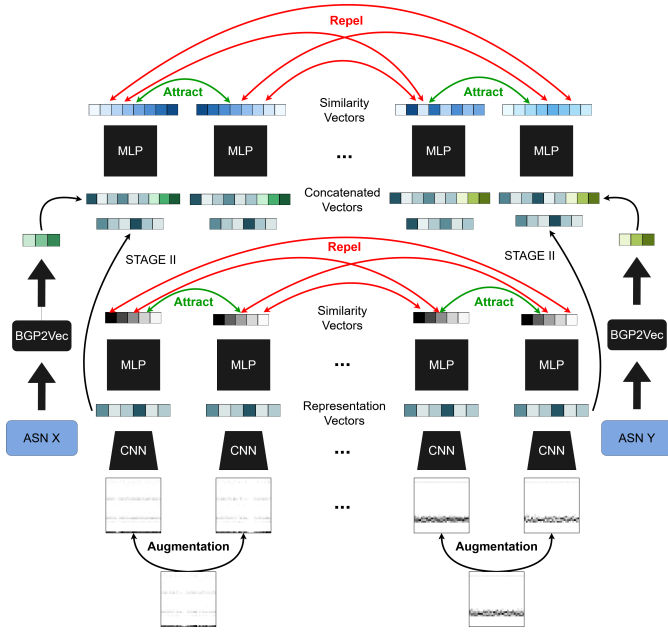


Figure 1. Joint multimodal model overview.

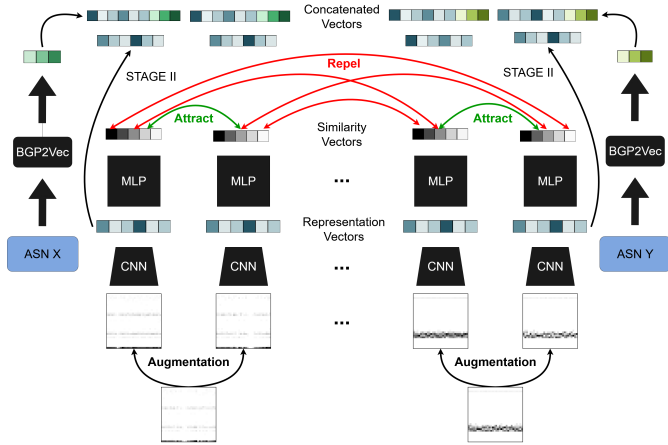


Figure 2. Frozen multimodal model overview.

were obtained from RouteViews file from January 13, 2025. We adopt the original BGP2Vec embedding dimensionality ($d=32$) for the learned AS vectors.

2) *Per-flow AS feature construction*: For each flow, we extract the relevant AS identifiers (e.g., origin AS for source or destination IPs) and look up their pretrained BGP2Vec embeddings. The resulting AS embedding is used as an additional input to the classifier.

D. Architecture and Training

1) *Stage I: SimCLR backbone (FlowPic encoder)*: We follow the SimCLR pipeline of Horowicz et al. [10], [11] *without modification* to obtain a FlowPic representation extractor. At each training step, we sample a minibatch of N unlabeled

samples $\{x_k\}_{k=1}^N$ and independently sample two augmentations $t, t' \sim \mathcal{A}$. Each sample produces two augmented views:

$$\tilde{x}_{2k-1} = t(x_k), \quad \tilde{x}_{2k} = t'(x_k),$$

forming a batch of $2N$ views. The two views originating from the same x_k constitute a positive pair, while all other views in the batch act as negatives.

Let f_ϕ be the CNN encoder and g_ψ the projection head. Each view is encoded into a representation vector

$$h_i = f_\phi(\tilde{x}_i) \in \mathbb{R}^{120},$$

and projected to the similarity space where the contrastive loss is applied:

$$z_i = g_\psi(h_i) \in \mathbb{R}^{30}.$$

We use cosine similarity $\text{sim}(u, v) = \frac{u^\top v}{\|u\|_2 \|v\|_2}$ and the NT-Xent loss. For a positive pair (i, j) :

$$\mathcal{L}_{\text{SimCLR}}(i, j) = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{1}[k \neq i] \exp(\text{sim}(z_i, z_k)/\tau)},$$

where $\mathbf{1}[k \neq i]$ is an indicator function that equals 1 if $k \neq i$ and 0 otherwise, and τ denotes a temperature parameter. The final loss is computed across all positive pairs in the minibatch in both directions, i.e., (i, j) and (j, i) , and the parameters of $f_\phi(\cdot)$ and $g_\psi(\cdot)$ are updated accordingly. After contrastive pretraining, we keep $f_\phi(\cdot)$ as the representation extractor and discard the projection head $g_\psi(\cdot)$.

RTT augmentation: Consistent with Horowicz et al., we apply RTT scaling by multiplying packet arrival times by a random factor $\alpha \sim \mathcal{U}(0.5, 1.5)$.

2) *Stage II: ASN fusion head (contrastive refinement)*: To isolate the effect of the ASN context, we introduce BGP2Vec *only after* Stage I. For each view $\tilde{x}_i$, we compute

$$h_i = f_\phi(\tilde{x}_i) \in \mathbb{R}^{120}.$$

Let $e_{\text{AS}} \in \mathbb{R}^{32}$ be the fixed pretrained BGP2Vec embedding for the flow's ASN. We concatenate at the representation level:

$$u_i = [h_i; e_{\text{AS}}] \in \mathbb{R}^{152}.$$

We then apply a two-layer MLP head p_θ to map the fused vector into a 30-dimensional contrastive space:

$$y_i = p_\theta(u_i) \in \mathbb{R}^{30}.$$

Stage II uses the same NT-Xent loss, replacing z_i by y_i :

$$\mathcal{L}_{\text{StageII}}(i, j) = -\log \frac{\exp(\text{sim}(y_i, y_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{1}[k \neq i] \exp(\text{sim}(y_i, y_k)/\tau)}$$

BGP2Vec embeddings remain fixed throughout training.

3) *Frozen vs. Joint Training*: **Frozen training**. In the Frozen variant, shown in Figure 2, we first train Stage I to obtain f_ϕ , then freeze f_ϕ and combine it with the frozen BGP2Vec model. This variant performs no additional training after combining the ASN and FlowPic embeddings.

Joint training. In the Joint variant, shown in Figure 1, we train the fusion head p_θ together with the FlowPic encoder f_ϕ end-to-end using $\mathcal{L}_{\text{StageII}}$. Gradients are backpropagated through p_θ and into f_ϕ , while BGP2Vec remains fixed.

IV. EXPERIMENTS AND RESULTS

A. Experimental Setup and Evaluation Protocol

To check whether our idea is viable, we first need to verify a basic premise: on ISCX, adding ASN information should *not* make the prediction task trivial, e.g., if each application is served on a unique set of ASNs. We then (i) replicate the SimCLR training and evaluation protocol of Horowicz et al. [10], [11], (ii) reproduce BGP2Vec as described by Shapira and Shavitt [17]–[19], (iii) evaluate our two proposed multimodal variants to determine which training strategy performs better, and (iv) compare them with previous work.

a) *Models compared:* We evaluate three variants: (i) **Base:** the SimCLR FlowPic representation (no ASN) used in prior work [10], [11], [13]. (ii) **Frozen:** the Base representation concatenated with a fixed BGP2Vec ASN embedding and passed through a *frozen* multimodal head (i.e., the SimCLR backbone is not updated during evaluation). (iii) **Joint:** an *end-to-end* variant in which the Stage II head and the SimCLR backbone are trained jointly.

b) *Few-shot evaluation:* We sample n labeled examples per class from the training set, $n \in \{1, 3, 5, 7, 10, 20, 100\}$, train a supervised linear classifier on the resulting representations, and evaluate on the fixed test set.

B. Sufficiency of ASN alone

A natural concern when introducing AS context is that classification becomes trivial because the server ASN almost determines the application label. Recent work supports the intuition that *AS-level context* is useful for encrypted traffic classification. Kahana et al. [15] propose a hierarchical alternative to the standard flat classifier by first grouping flows according to the server IP address, and then training specialized classifiers that distinguish only among the applications observed within each group. One variant is *ASN-based grouping*, where popular server IPs are mapped to their ASNs and separate classifiers are trained per major ASN group, with a flat fallback model for connections whose server IP is not in any major group. Empirically, this hierarchy improves over a flat baseline (e.g., Random Forest F1 increases from 87.92 to 92.58 with up to 10 packets per flow, and from 88.07 to 93.42 with up to 50 packets).

To check whether ASN alone is sufficient in our setting, we run tests on ISCX using the few-shot protocol employed throughout the paper. First, we train tree-based models, such as Gradient Boosting, LightGBM, and XGBoost, using *ASN and coarse per-flow statistics*, meaning our covariates are: (ASN, packet count, mean packet size, and packet-size variance); this directly probes the “ASN makes it trivial” hypothesis.

Second, we train the same models using only *coarse per-flow statistics* (packet count, mean packet size, and packet-size variance), *without* ASN. We do that to check the direct impact of having the ASN as a covariate in our model. Third, we train the same models again using only *ASN* as our sole covariate. We do that to check the impact of the *coarse per-flow statistics* covariates on accuracy.

Table I

TREE-BASED RESULTS ON ISCX (10 CLASSES) AT $n=10$ (100 TRAINING FLOWS). **ASN-ONLY** USES ASN AS THE SOLE FEATURE. **STATS** USES ONLY COARSE PACKET STATISTICS: PACKET COUNT, MEAN PACKET SIZE, AND PACKET-SIZE VARIANCE. **BOTH** USES ASN TOGETHER WITH THESE STATISTICS. VALUES ARE MEAN ACCURACY (%) ACROSS RUNS.

Model	ASN-only (%)	Stats (%)	Both (%)	Δ pp
Grad. Boosting	39.67 $\pm$ 2.36	71.33 $\pm$ 2.83	73.33 $\pm$ 1.41	+2.00
LightGBM	36.33 $\pm$ 0.47	61.50 $\pm$ 4.01	76.50 $\pm$ 3.06	+15.00
XGBoost	39.67 $\pm$ 2.36	69.00 $\pm$ 1.41	73.00 $\pm$ 6.60	+4.00

Overall, Table I shows that *ASN + coarse packet statistics* are not sufficient to reliably classify applications on ISCX, but it also shows that ASN is an informative covariate that can improve classification accuracy, making it a promising signal to incorporate *within* a representation-learning pipeline.

C. ASN Mapping

Mapping ISCX endpoints to ASNs is occasionally ambiguous due to noisy session metadata (e.g., cases where the local lab ASN appears as source, destination, or both). We therefore employ a simple, conservative heuristic: we prefer the destination ASN when it is mapped and not the local lab ASN (ASN 611); otherwise we fall back to the source ASN under the same condition. If neither endpoint maps to a non-611 ASN, we retain ASN 611 when available, and use a reserved value (ASN 0) when no mapping is possible. Importantly, the Frozen model remains effective despite this noise, suggesting that representation-level fusion is robust even when AS attribution is imperfect.

D. Main Few-Shot Results (Multimodal vs. Base)

We evaluate whether incorporating AS context through BGP2Vec improves few-shot generalization when fused with FlowPic representations.

Table II shows that adding ASN context yields consistent gains of roughly 8 to 12 percentage points. Figure 3 visualizes this trend, showing that the Frozen and Joint variants, along with their confidence intervals, remain consistently above the FlowPic-only Base model across all shot counts.

To contextualize performance at the standard $n=10$ regime, Table III compares our Base/Frozen results to previously reported reference points. We include Horowicz et al. [11] at $n=10$, and the reported FlowPicClip [16] results at $n \in \{5, 7, 10\}$ from the FlowPicClip study.

Figure 4 provides a visualization of the representation space, comparing the FlowPic-only Base model with the Frozen multimodal representation. Compared with the FlowPic-only Base representation, the Frozen multimodal representation exhibits visually tighter within-class clusters and clearer separation between several application classes. This qualitative improvement complements the accuracy gains reported in Table II.

E. Statistical Significance and Model Comparison

Having shown that adding ASN context improves over the FlowPic-only Base model, we next ask whether the more

Table II

FEW-SHOT ACCURACY ON ISCX (10 CLASSES): BASE VS. FROZEN VS. JOINT. VALUES ARE MEAN ACCURACY (%) WITH 95% CI ACROSS 25 EXPERIMENTS. Δ IS THE ABSOLUTE GAIN IN PERCENTAGE POINTS (PP) FOR **FROZEN** – **BASE**, REPORTED WITH ITS 95% CI.

#Shots	Base %	Frozen %	Joint %	Δ % Points
1	55.44 $\pm$ 1.76	63.33 $\pm$ 1.94	62.52 $\pm$ 1.89	+7.89 $\pm$ 2.19
3	71.60 $\pm$ 1.55	80.63 $\pm$ 1.33	79.04 $\pm$ 2.23	+9.03 $\pm$ 1.35
5	75.65 $\pm$ 0.97	86.93 $\pm$ 1.26	85.91 $\pm$ 1.40	+11.28 $\pm$ 1.31
7	79.07 $\pm$ 0.99	89.09 $\pm$ 0.80	87.99 $\pm$ 1.26	+10.03 $\pm$ 1.31
10	81.41 $\pm$ 1.28	90.48 $\pm$ 0.65	90.41 $\pm$ 1.16	+9.07 $\pm$ 1.12
20	83.68 $\pm$ 0.83	91.68 $\pm$ 0.61	91.77 $\pm$ 0.89	+8.00 $\pm$ 0.81
100	87.27 $\pm$ 1.14	95.73 $\pm$ 0.73	95.71 $\pm$ 0.83	+8.47 $\pm$ 0.87

Table III

REFERENCE COMPARISON ON ISCX AT CANONICAL FEW-SHOT POINTS. BASE/FROZEN SHOW 95% CI ACROSS $N=25$ EVALUATION RUNS. OTHER ROWS ARE REPRODUCED FROM THE CITED WORKS AS REPORTED.

Method	#Shots	Accuracy (%)
Frozen (ours)	5	86.93 $\pm$ 1.26
Frozen (ours)	7	89.09 $\pm$ 0.80
Frozen (ours)	10	90.48 $\pm$ 0.65
FlowPicClip [16]	5	79.23 $\pm$ 1.66
FlowPicClip [16]	7	82.60 $\pm$ 2.50
FlowPicClip [16]	10	86.05 $\pm$ 0.52
Horowicz et al. [11]	10	79.83

Table IV

JOINT VS. FROZEN PER- n COMPARISON WITH BONFERRONI CORRECTION OVER $|\mathcal{N}| = 7$ TESTS ($\alpha = 0.05$). Δ DENOTES JOINT – FROZEN IN PERCENTAGE POINTS (PP).

n	Δ mean (pp)	SE (pp)	95% CI (pp)	p_{bonf}	Sig.
1	-0.81	1.17	[-3.24, +1.61]	1.0000	×
3	-1.59	1.27	[-4.21, +1.04]	1.0000	×
5	-1.03	0.95	[-2.98, +0.93]	1.0000	×
7	-1.11	0.70	[-2.55, +0.34]	1.0000	×
10	-0.07	0.61	[-1.32, +1.19]	1.0000	×
20	+0.09	0.48	[-0.90, +1.09]	1.0000	×
100	-0.03	0.38	[-0.82, +0.76]	1.0000	×

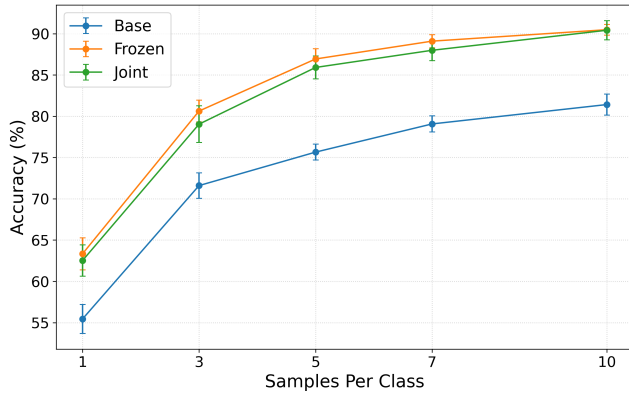


Figure 3. Model accuracy as a function of training set size with 95% confidence intervals.

complex Joint training procedure provides an additional benefit over the simpler Frozen variant. To test this, we compare Joint and Frozen across the same few-shot options.

Because we evaluate the comparison at *seven* different shot values, we control the familywise error rate using a Bonferroni correction with overall significance level $\alpha = 0.05$. For each $n \in \{1, 3, 5, 7, 10, 20, 100\}$, we test the one-sided hypothesis

that Joint improves accuracy over Frozen:

$$H_0 : \mu_{\Delta(\text{Joint}-\text{Frozen})}(n) \leq 0 \quad \text{vs.} \quad H_1 : \mu_{\Delta(\text{Joint}-\text{Frozen})}(n) > 0.$$

Thus, significance is declared only when the Bonferroni-corrected result satisfies $p \leq \alpha/7$.

Table IV reports the per- n mean accuracy differences, standard errors, confidence intervals, and Bonferroni-corrected p -values. Across all few-shot settings, Joint does not significantly improve over Frozen. Moreover, the mean differences are small and often negative, indicating that the added complexity of end-to-end Joint training does not yield a reliable performance advantage. We therefore use the Frozen variant as our main model in this paper, since it is simpler, easier to train, and statistically indistinguishable from Joint under our evaluation.

F. Why BGP2Vec Enables Robust ASN Use

A practical advantage of using BGP2Vec embeddings is that ASN information is incorporated as a *vector representation* rather than as a categorical identifier. This allows the model to assign meaningful representations to ASNs that may not appear in the labeled few-shot set, supporting generalization in the presence of sparse labels. In contrast, classifiers that treat ASN as a raw categorical feature can only exploit AS identities that are sufficiently represented during training.

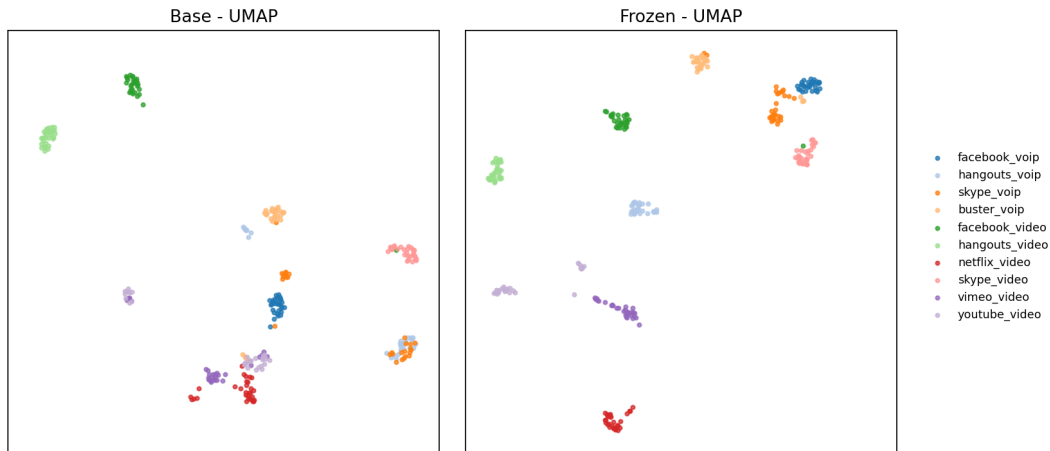


Figure 4. UMAP visualization of frozen representations: Base (FlowPic only) vs. Frozen (FlowPic + ASN embedding).

V. CONCLUSIONS

We introduced a FlowPic-based traffic classification framework that integrates *Autonomous System* (AS) context into contrastive representation learning. Our approach preserves the SimCLR training and few-shot evaluation protocol of Horowicz et al. [10], [11] and concatenates the learned FlowPic representation with *BGP2Vec* embeddings [17]–[19]. By representing each ASN with an *unsupervised* BGP2Vec embedding trained on BGP data, we can use ASN context *without requiring every ASN to be observed during training*.

On the ISCX VoIP/Video application identification task, we showed that fusing ASN context yields consistent and significant improvements over the FlowPic-only baseline. Beyond the setting studied here, we have also observed significant improvements when applying the same metadata-driven approach to a substantially larger classification task with more than 100 classes. A full evaluation of this setting is left for future work.

Acknowledgment: This work was partially funded by the Israeli DDR&D and the INCD.

REFERENCES

- [1] M. Crotti, M. Dusi, F. Gringoli, and L. Salgarelli, “Traffic classification through simple statistical fingerprinting,” *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 1, pp. 5–16, 2007.
- [2] S. Rezaei and X. Liu, “How to achieve high classification accuracy with just a few labels: A semi-supervised approach using sampled packets,” in *Industrial Conference on Data Mining (ICDM)*, 2019.
- [3] G. Aceto, D. Ciunzo, A. Montieri, and A. Pescapé, “Mobile encrypted traffic classification using deep learning: Experimental evaluation, lessons learned, and challenges,” *IEEE Transactions on Network and Service Management*, vol. 16, Jun. 2019.
- [4] H. Huang, H. Deng, J. Chen, L. Han, and W. Wang, “Automatic multi-task learning system for abnormal network traffic detection,” *International Journal of Emerging Technologies in Learning*, vol. 13, no. 4, 2018.
- [5] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, “Network traffic classifier with convolutional and recurrent neural networks for internet of things,” *IEEE Access*, vol. 5, pp. 18 042–18 050, 2017.
- [6] S. Rezaei and X. Liu, “Multitask learning for network traffic classification,” in *2020 29th International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2020, pp. 1–9.
- [7] T. Shapira and Y. Shavitt, “FlowPic: Encrypted internet traffic classification is as easy as image recognition,” in *IEEE Workshop on Network Intelligence: Machine Learning for Networking (NI)*, 2019, pp. 680–687.
- [8] —, “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.
- [9] L. Yang, A. Finamore, F. Jun, and D. Rossi, “Deep learning and zero-day traffic classification: Lessons learned from a commercial-grade dataset,” *IEEE Transactions on Network and Service Management*, vol. 18, no. 4, pp. 4103–4118, 2021.
- [10] E. Horowicz, T. Shapira, and Y. Shavitt, “A few shots traffic classification with mini-FlowPic augmentations,” in *The 22nd ACM Internet Measurement Conference (IMC)*, 2022, pp. 647–654.
- [11] —, “Self-supervised traffic classification: Flow embedding and few-shot solutions,” *IEEE Transactions on Network and Service Management*, vol. 21, pp. 3054 – 3067, 2024.
- [12] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.
- [13] A. Finamore, C. Wang, J. Krolikowski, J. M. Navarro, F. Chen, and D. Rossi, “Replication: Contrastive learning and data augmentation in traffic classification using a Flowpic input representation,” in *ACM Internet Measurement Conference (IMC)*, 2023, pp. 36–51.
- [14] C. Wang, A. Finamore, P. Michiardi, M. Gallo, and D. Rossi, “Data augmentation for traffic classification,” in *Passive and Active Network Measurement (PAM)*, 2024, pp. 159–186.
- [15] Y. Kahana, Y. J. Stein, and O. Glickman, “Fast real-world classification of ECH-enabled applications,” *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 4, pp. 337–350, 2026.
- [16] D. Shalev, T. Shapira, and Y. Shavitt, “FlowPicClip: Improving network traffic classification using language supervision,” 10 2025, pp. 1–6.
- [17] T. Shapira and Y. Shavitt, “Unveiling the type of relationship between autonomous systems using deep learning,” in *IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2020, pp. 1–6.
- [18] —, “BGP2Vec: Unveiling the latent characteristics of autonomous systems,” *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 4516–4530, 2022.
- [19] —, “A deep learning approach for IP hijack detection based on asn embedding,” in *ACM SIGCOMM Workshop on Network Meets AI & ML (NetAI’20)*, 2020, p. 35–41.
- [20] G. Draper-Gil, A. H. Lashkari, M. S. I. 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 - Volume 1: ICISPP, INSTICC*. SciTePress, 2016, pp. 407–414.