Poster: NetDPSyn: Synthesizing Network Traces under Differential Privacy

Danyu Sun[†], Joann Qiongna Chen[§], Chen Gong[‡], Tianhao Wang[‡], and Zhou Li[†✉]
[†]University of California, Irvine, [§]San Diego State University, [‡]The University of Virginia
{danyus2, zhou.li}@uci.edu, {jchen27}@sdsu.edu, {fzv6en, tianhao}@virginia.edu,

Abstract

As the utilization of network traces for the network measurement research becomes increasingly prevalent, concerns regarding privacy leakage from network traces have garnered the public's attention. To safeguard network traces, researchers have proposed the *trace synthesis* that retains the essential properties of the raw data. However, previous works also show that synthesis traces with generative models are vulnerable under linkage attacks.

This paper introduces NetDPSyn, the first system to synthesize high-fidelity network traces under privacy guarantees. NetDPSyn is built with the Differential Privacy (DP) framework as its core, which is significantly different from prior works that apply DP when training the generative model. The experiments conducted on three flow and two packet datasets indicate that NetDPSyn achieves much better data utility in downstream tasks like anomaly detection. NetDPSyn is also 2.5 times faster than the other methods on average in data synthesis.

Acknowledgment

This work [1] has been accepted by the ACM Internet Measurement Conference (IMC) 2024. The original abstract and author list are shown above. We post the paper link with the conference version[1] as well as the extended version available on ArXiv[2].

References

[1] D. Sun, J. Q. Chen, C. Gong, T. Wang, and Z. Li. Netdpsyn: Synthesizing network traces under differential privacy. In *Proceedings of the 2024 ACM on Internet Measurement Conference*, IMC '24, page 545–554, New York, NY, USA, 2024. Association for Computing Machinery.

✉ Corresponding authors.

[1] https://dl.acm.org/doi/10.1145/3646547.3689011.

[2] https://arxiv.org/pdf/2409.05249.

NetDPSyn: Synthesizing Network Traces under Differential Privacy

Danyu Sun[1] Joann Qiongna Chen[2] Chen Gong[3] Tianhao Wang[3] Zhou Li[1]

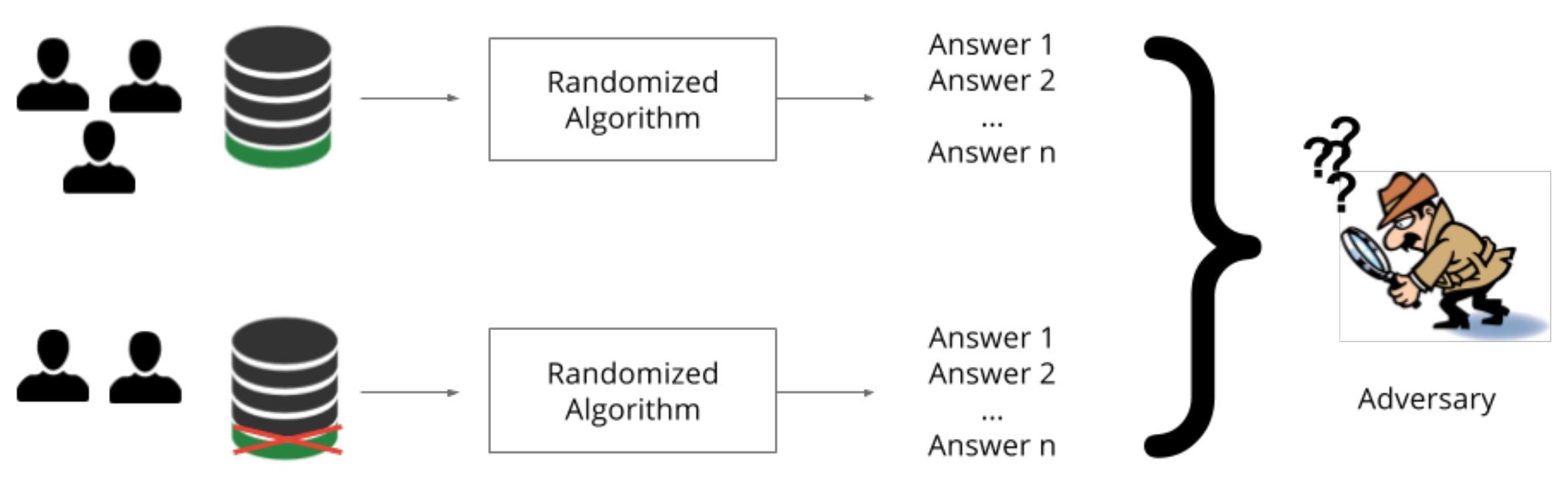
[1]University of California, Irvine [2]San Diego State University [3]University of Virginia

Background

- **Network Data and Privacy:** We consider header fields of network *packet* or *flows* as the target for data synthesis. Releasing the header without payload still raises privacy concerns, and the main solutions include data anonymization and synthesis.
- **Differential Privacy:** A randomized mechanism $\mathcal{A}$ satisfies (ε, δ)-differential privacy ($\varepsilon > 0$ and $\delta > 0$), if and only if, for any two neighboring datasets D and D', it holds that,

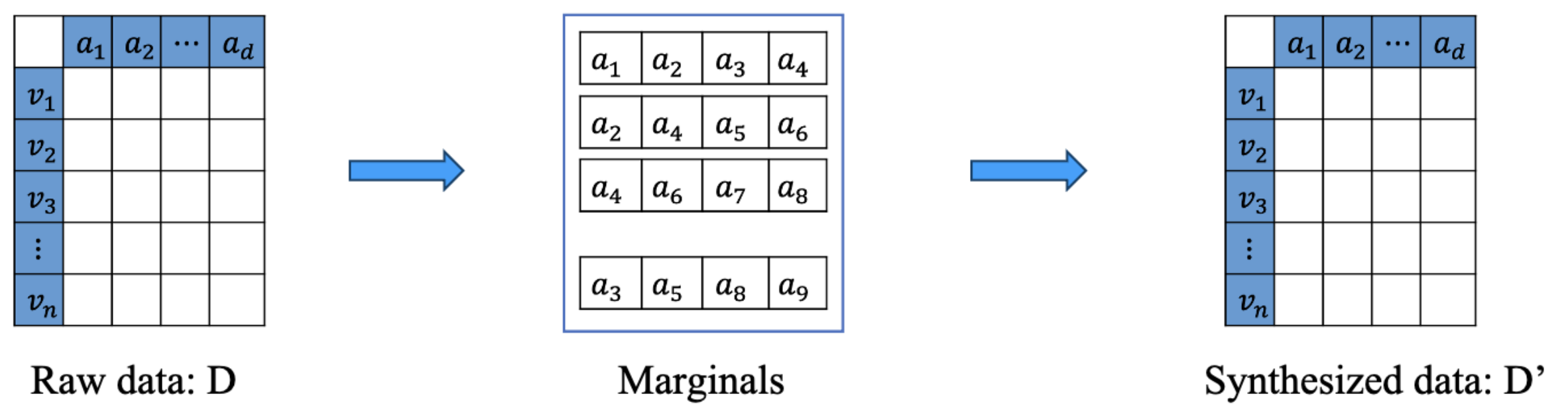
$$\Pr[\mathcal{A}(D) \in \mathcal{O}] \leq e^{\varepsilon} \Pr[\mathcal{A}(D') \in \mathcal{O}] + \delta \tag{1}$$

DP ensures the privacy of an individual's data within a dataset is preserved data processing. It guarantees that the result of any computation, such as a database query, remains essentially unchanged whether or not any single individual's data is included or excluded.



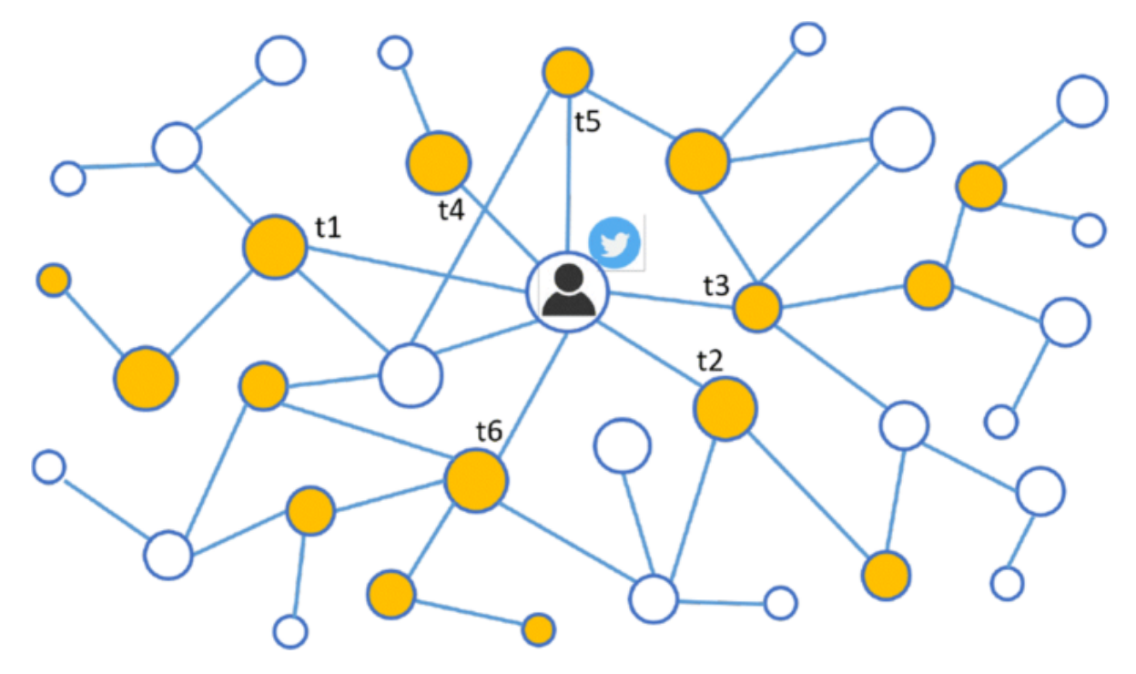
Motivation

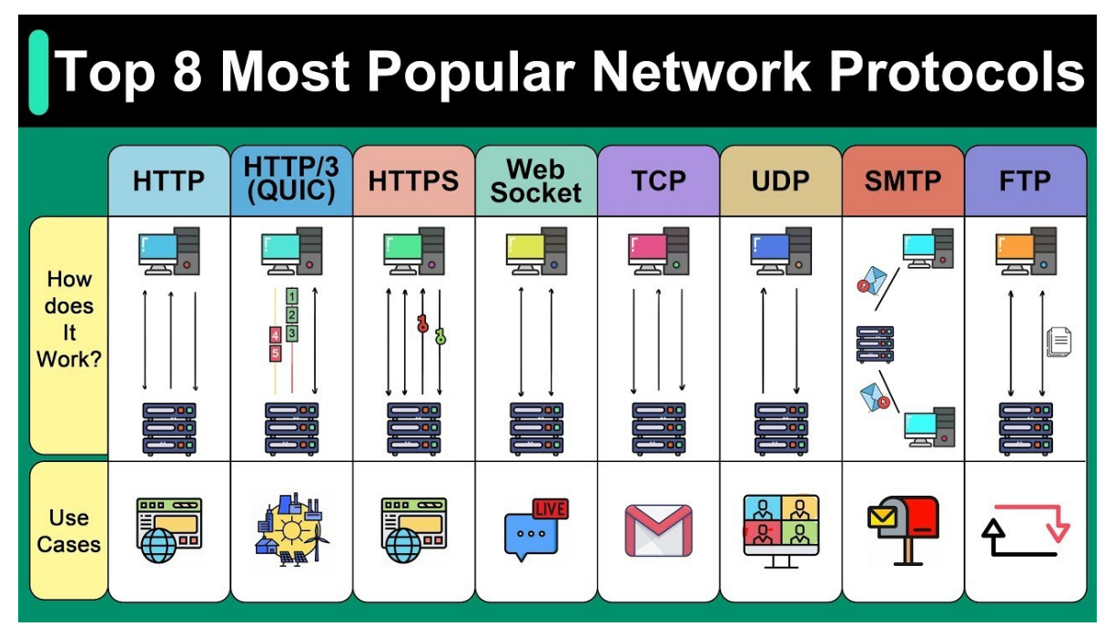
- To achieve formal privacy guarantees, the training of the generative model can be hardened with DP. We found **Only two** works that synthesize network traces applied DP [1, 3]. However, the **data utility** will be significantly worse.
- We pursue a different direction to capture the **underlying distribution** of the original data and then synthesizing network records **after** they are protected by DP.
- **PrivSyn** [4] handles high-dimensional datasets by automatically selecting and constructing noisy marginal tables that captures the data distribution.



Challenges

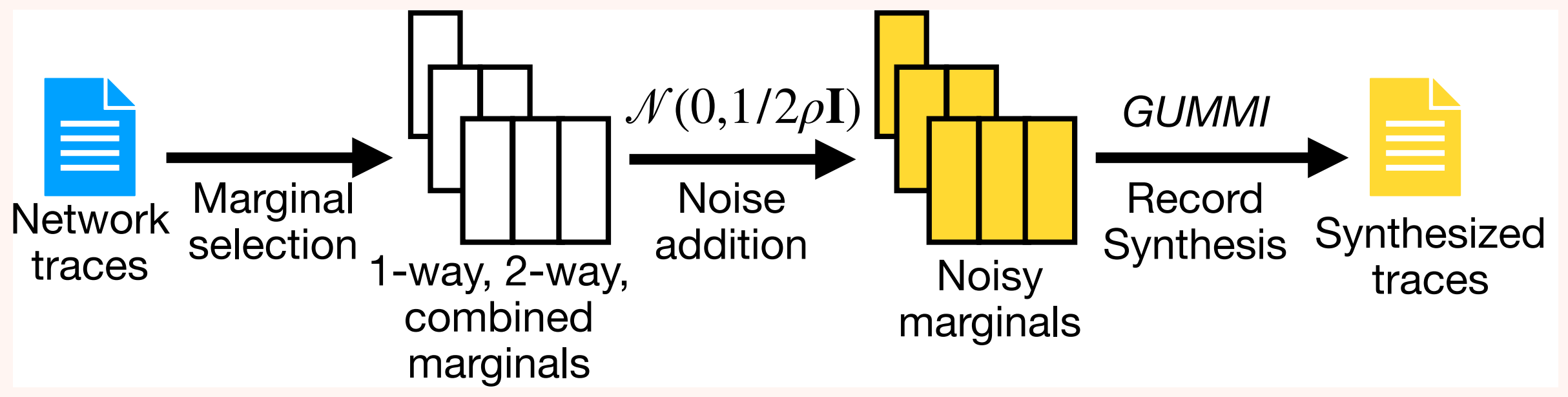
- **Large Domain Size:** E.g., the number of bytes per flow and long-tail distribution.
- **Temporal Patterns:** It is essential to downstream applications. However, such patterns cannot be synthesized using marginals tables.
- **Inherent Constraints of Network Protocol:** E.g., FTP communications use TCP and the destination ports are 20 and 21.
- **Time Consuming:** PrivSyn [4] uses the Gradually Update Method (GUM) to synthesize records from noisy marginal tables, but this method can take a long time to converge for large network datasets.





NetDPSyn [2]

- **Input**: Private Network Traces.
- **Output**: Synthesized Data.



Pre-Process

- **IP:** Bin the low-count IP address by the /30 prefix.
- **Port:** Keep common port under 1024, bin higher by 10.
- **Integer and float-point:** Log transformation.
- **Timestamp:** Group-wise differences between timestamp.
- **Category:** No need to be binned.

Marginal Selection

- **Marginal Tables:** Marginal tables example for `dstport` and `type` computed on TON dataset.

v	$M_{\mathrm{d}}(v)$
⟨53, *⟩	82828
⟨80, *⟩	68748
⟨15600, *⟩	27255

(a) 1-way marginal for dstport.

v	$M_{\mathrm{t}}(v)$
⟨∗, normal⟩	166494
⟨∗, injection⟩	15951

(b) 1-way marginal for type.

v	$M_{\mathrm{dt}}(v)$
⟨53, normal⟩	74547.08
⟨53, injection⟩	554.71
⟨80, normal⟩	12297.88
⟨80, injection⟩	15396.66
⟨15600, normal⟩	27247.02
⟨15600, injection⟩	20.09

(c) 2-way noisy marginal before marginal post-processing.

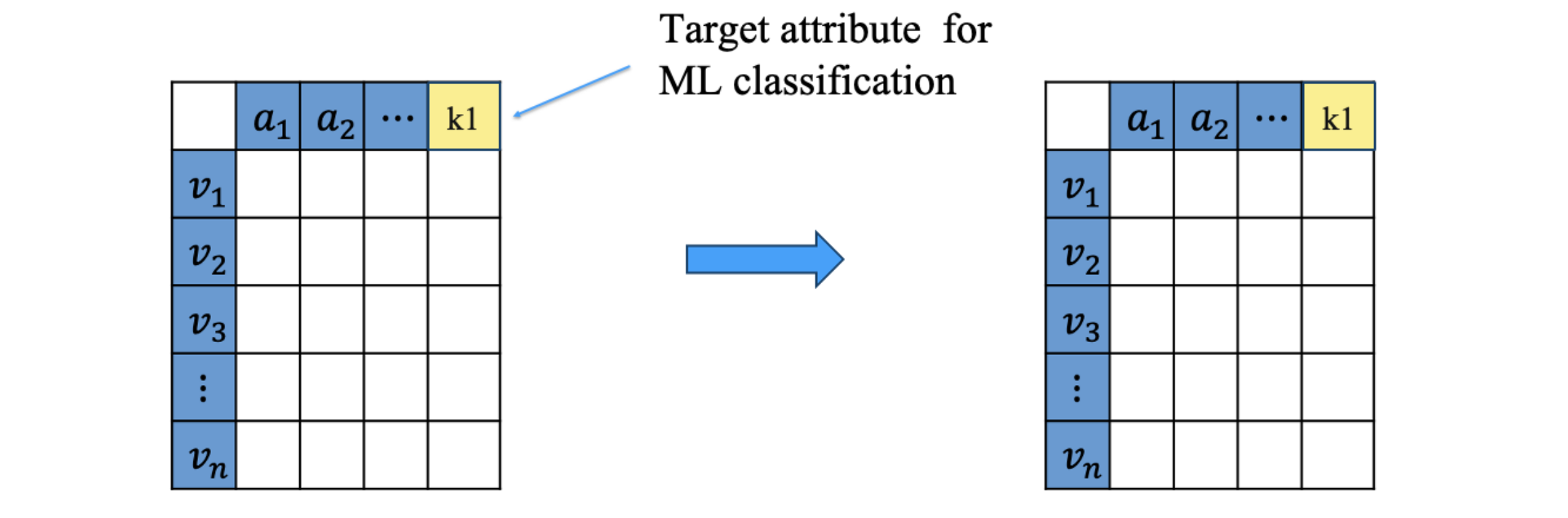
v	$\tilde{M}_{\mathrm{dt}}(v)$
⟨53, normal⟩	74566
⟨53, injection⟩	558
⟨80, normal⟩	12308
⟨80, injection⟩	15364
⟨15600, normal⟩	27255
⟨15600, injection⟩	0

(d) Actual 2-way marginal.

- **Marginal Selection:** DenseMarg [4] formalizes the marginal selection problem as an optimization problem that balances dependency error (error caused by missing a marginal) and noise error (error caused by adding noises to a selected marginal).

$$\text{minimize} \sum_{i=1}^{m} [\overbrace{\psi_i \underbrace{x_i}_{\textit{selected two-way marginal}}}^{\textit{noise error}} + \overbrace{\phi_i \underbrace{(1 - x_i)}_{\textit{non-selected two-way marginal}}}^{\textit{dependency error}}] \text{ s.t. } x_i \in \{0, 1\}$$

GUMMI

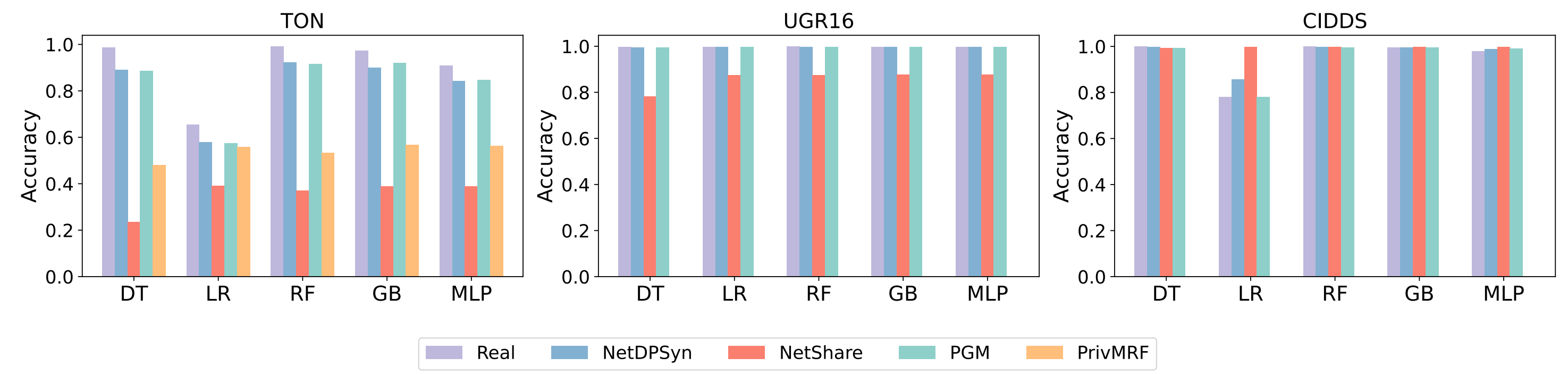

Target attribute for ML classification
a_1 a_2 ⋯ k1
v_1 v_2 v_3 ⋮ v_n
a_1 a_2 ⋯ k1
v_1 v_2 v_3 ⋮ v_n

Step 1: Initialize dataset that **contains key** to the downstream tasks, e.g. label

Step 2: Using all noisy marginal to update the initialized dataset.

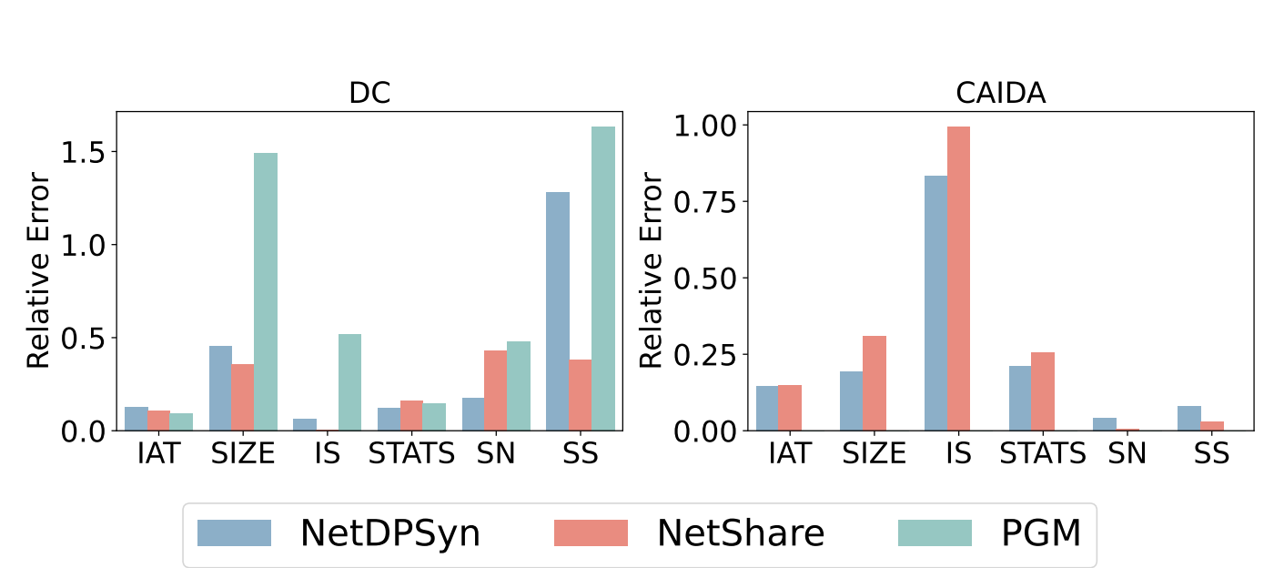
Evaluation Results

- Machine Learning Classification Tasks on 3 Flow Data:

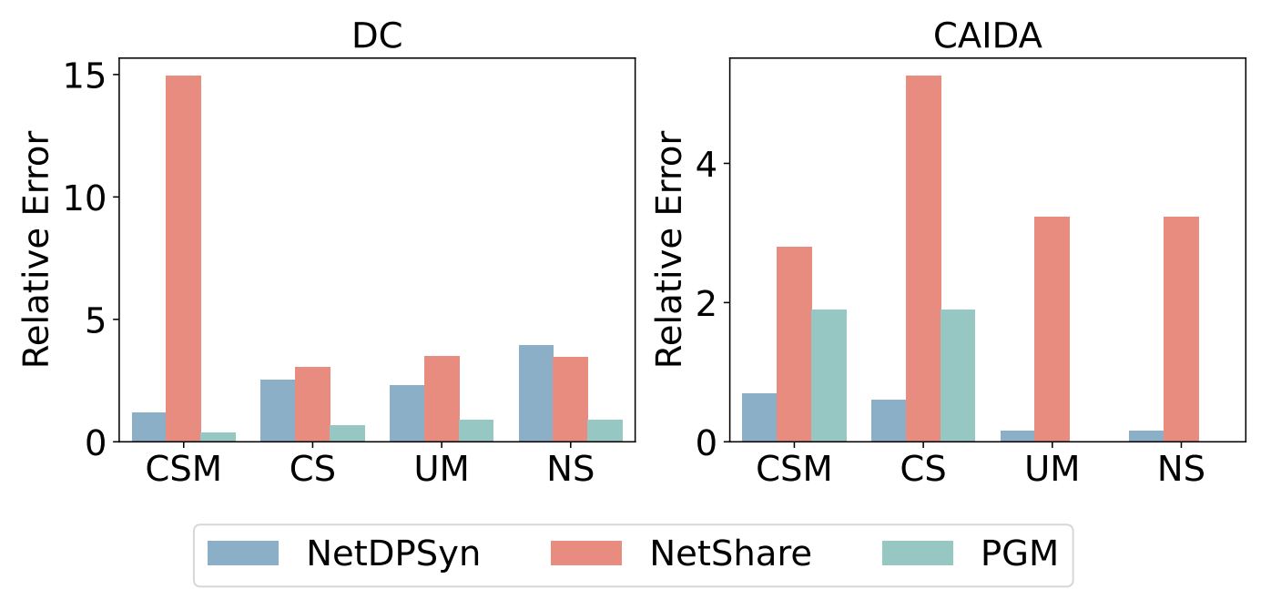


Accuracy of 5 ML Tasks.

- Anomaly Detection and Sketch Algorithms on 2 Packet Data:



Relative Error of NetML.



Relative Error of Various Sketch Algorithms.

- Running Time of Each Method in Minutes:

	NetDPSyn	NetShare	PGM	PrivMRF
TON	10	27	70	240
CIDDS	20	100	55	N/A
UGR16	40	94	55	N/A
CAIDA	35	30	54	N/A
DC	20	100	24	N/A

Conclusions and Future Work

Conclusions:

- Synthesize **high-fidelity** network traces under privacy guarantee.
- Achieve **better data utility** in downstream task.
- **2.5 times** faster than other methods.

Future Work:

- Model the temporal pattern in complex representation.
- Cover all types of network environment and data type.
- Evaluate advanced downstream tasks like graph-based anomaly detection.

Project Resources

Paper Link

GitHub Repository

Danyu Sun's Homepage

References

[1] Liyue Fan and Akarsh Pokkunuru.
Dpnet: Differentially private network traffic synthesis with generative adversarial networks.
In *IFIP Annual Conference on Data and Applications Security and Privacy*, pages 3–21. Springer, 2021.

[2] Danyu Sun, Joann Qiongna Chen, Chen Gong, Tianhao Wang, and Zhou Li.
Netdpsyn: Synthesizing network traces under differential privacy.
In *Proceedings of the 2024 ACM on Internet Measurement Conference*, IMC '24, page 545–554, New York, NY, USA, 2024. Association for Computing Machinery.

[3] Yucheng Yin, Zinan Lin, Minhao Jin, Giulia Fanti, and Vyas Sekar.
Practical gan-based synthetic ip header trace generation using netshare.
In *Proceedings of the ACM SIGCOMM 2022 Conference*, pages 458–472, 2022.

[4] Zhikun Zhang, Tianhao Wang, Ninghui Li, Jean Honorio, Michael Backes, Shibo He, Jiming Chen, and Yang Zhang.
{PrivSyn}: Differentially private data synthesis.
In *30th USENIX Security Symposium (USENIX Security 21)*, pages 929–946, 2021.

To appear in ACM IMC 2024

Poster for Network and Distributed System Security Symposium (NDSS) 2025

Presenter: Danyu Sun (danyus2@uci.edu)