SYNTHETIC DISTRIBUTED TRACES FOR IDENTIFICATION IN LIVE SYSTEMS

Sneh Patel, Yuvraj Sehgal, Mahsa Panahandeh

Naser Ezzati-Jivan, François Tétreault

Brock University • University of Ottawa • Ciena Corporation

Feb 2, 2026

THE CHALLENGE OF PRODUCTION TRACES

Privacy & Compliance Constraints

Production traces contain **sensitive operational details** that cannot be shared across teams or organizations due to privacy regulations and compliance requirements.

Storage Cost & Retention Policies

Traces are **sampled, truncated, or retained for short periods** due to storage costs, limiting the amount of data available for training and evaluation.

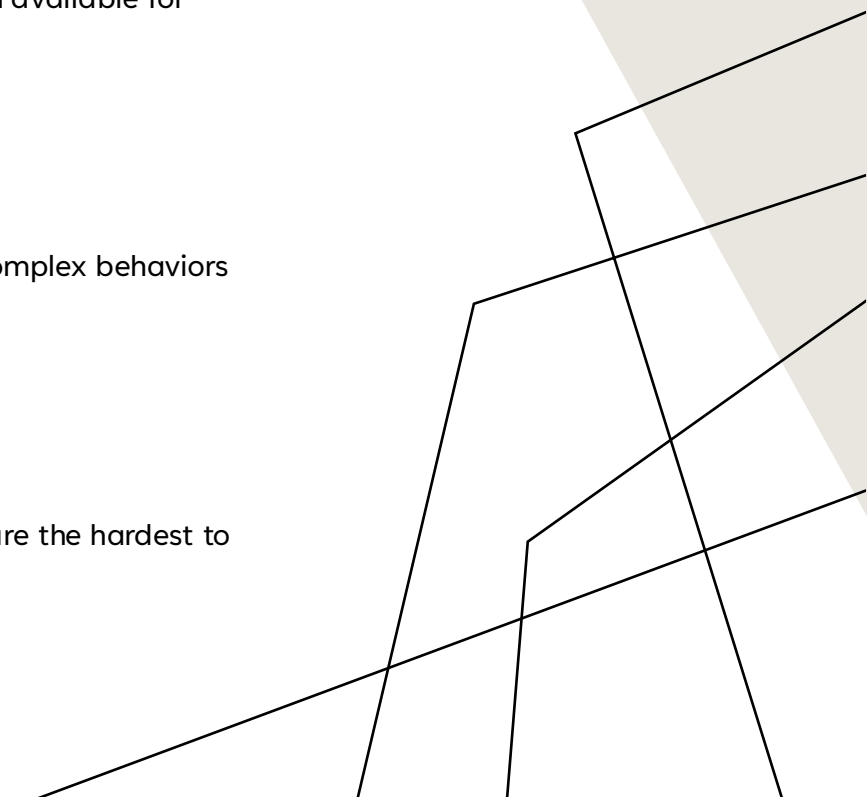
Sampling Bias

Trace repositories are **biased toward frequent, benign executions** , systematically underrepresenting rare or complex behaviors critical for robustness testing.

Long-Tail Underrepresentation

Rare or complex behaviors are often the most informative for debugging and reliability engineering, yet they are the hardest to capture, keep, and share across teams.

Industry context informed by discussions with engineers at Ciena Corporation.



CONTRIBUTIONS

Industry-Driven Need

Articulates the **practical constraints** observed with engineers at Ciena, including data scarcity, privacy limitations, and insufficient trace diversity for training learning-based observability tools.

Data Scarcity | Privacy | Diversity Gap

Hierarchical Framework

Presents a **hierarchical, graph-based generative framework** that models distributed traces as DAGs and separates global execution structure from local span-level behavior, with support for both fixed-size and variable-size generation.

Graph VAE | Hierarchical | DAG Structure

Deployment-Oriented Evaluation

Provides **empirical evaluation** that goes beyond reconstruction accuracy to assess downstream utility, including train-on-synthetic-test-on-real performance, hybrid training, and structural similarity analyses.

TSTR | Hybrid Training | Industrial Guidance

RESEARCH QUESTIONS

RQ1 Fidelity Preservation

How accurately do synthetic traces preserve the **structural and feature-level properties** of real distributed traces? This examines reconstruction accuracy for services, operations, durations, and execution dependencies.

Reconstruction | Structural Fidelity

RQ2 Downstream Utility

To what extent can synthetic traces **replace or supplement real trace data** in downstream analysis tasks? This evaluates train-on-synthetic-test-on-real performance and hybrid training scenarios.

Generalization | TSTR

RQ3 Trace Variability

How does trace variability (**fixed-size vs variable-size graphs**) affect generation fidelity and robustness? This compares uniform structure against heterogeneous execution depths.

Fixed-Size | Variable-Size

RQ4 Similarity & Separability

How similar are synthetic and real traces in the **joint feature space**, and how easily can they be distinguished? This uses clustering, PCA, and discriminative classification analysis.

Clustering | PCA

TRACE REPRESENTATION & DATASET

Graph Representation

Distributed traces are modeled as **directed acyclic graphs (DAGs)** where:

Node (V): Spans representing individual operations, each with feature vector $x_v = (s_v$: Service ID, o_v: Operation ID, d_v: Duration)

Edge (E): Parent-child execution relationships capturing causal dependencies

SocialNetwork (SN)

MicroServices: 21 **| Traces:** 1,244 **| Total Spans:** 11,649 **| Avg Spans/Trace:** 9.36

Services: 12 unique | **Operations:** 59 unique

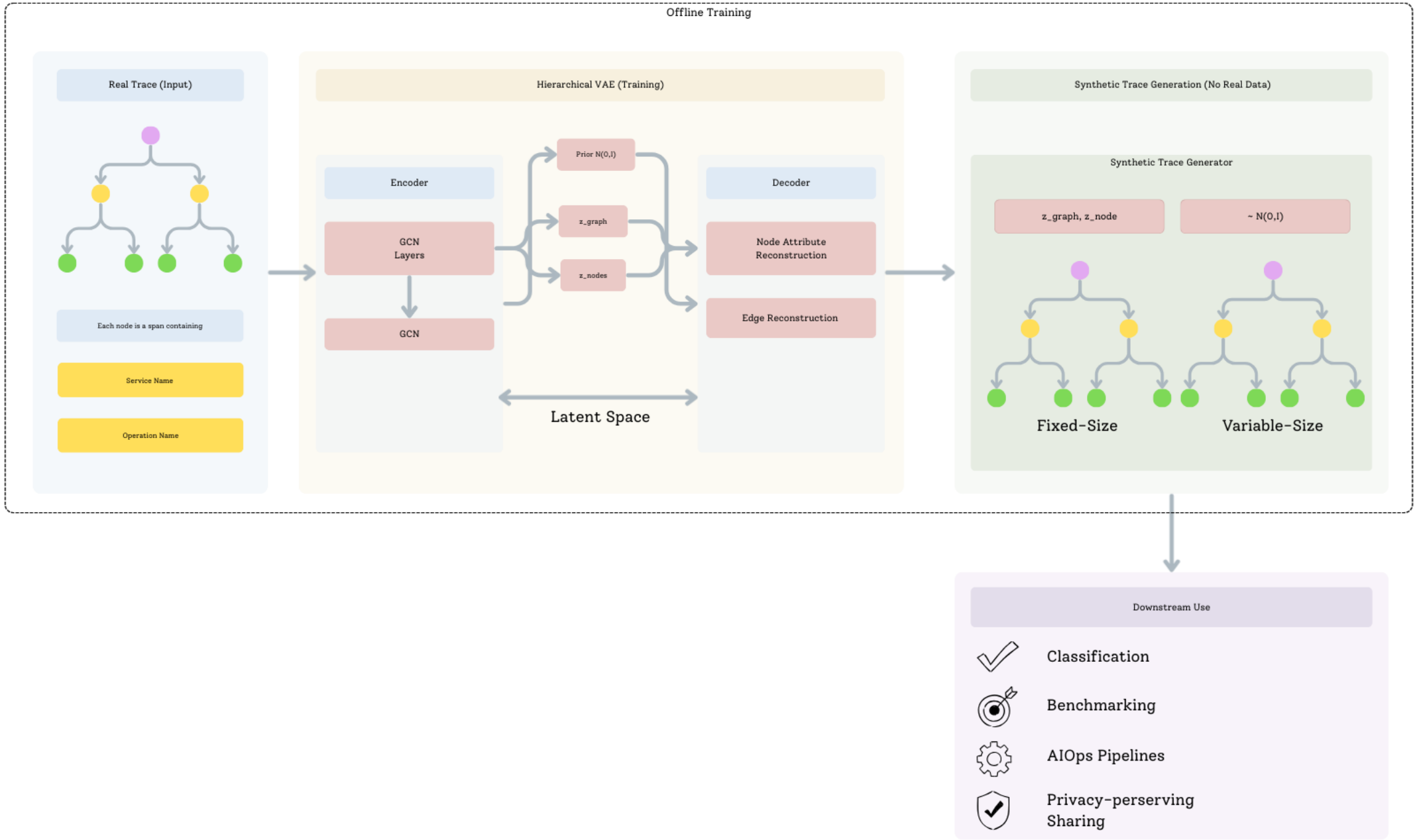
Broadcast-style social networking application

TrainTicket (TT)

MicroServices: 41 **| Traces:** 1,244 **| Total Spans:** 7,912 **| Avg Spans/Trace:** 6.36

Services: 13 unique | **Operations:** 16 unique

Online railway ticketing platform


Offline Training
Real Trace (Input)
Each node is a span containing
Service Name
Operation Name
Hierarchical VAE (Training)
Encoder
GCN Layers
GCN
Prior N(0,I)
z_graph
z_nodes
Latent Space
Decoder
Node Attribute Reconstruction
Edge Reconstruction
Synthetic Trace Generation (No Real Data)
Synthetic Trace Generator
z_graph, z_node
~ N(0,I)
Fixed-Size
Variable-Size
Downstream Use
Classification
Benchmarking
AIOps Pipelines
Privacy-perserving Sharing

TRAINING OBJECTIVE & GENERATION

Multi-Component Loss

1. Node Reconstruction (L_node)

Weighted cross-entropy for service/operation + MSE for duration

$$\mathcal{L}_{node} = \mathcal{L}_{service} + \mathcal{L}_{op} + \mathcal{L}_{duration}$$

2. Edge Reconstruction (L_edge)

Binary cross-entropy for parent-child dependency prediction

$$\mathcal{L}_{edge} = \mathrm{BCE}(\hat{E}, E)$$

3. KL Regularization (L_KL)

Separate regularization for graph and node latents with β_G > β_N

$$\mathcal{L}_{KL} = \beta_G \,\mathrm{KL}\big(q(z_G \mid G) \,\|\, \mathcal{N}(0, I)\big) + \beta_N \sum_{v \in V} \mathrm{KL}\big(q(z_v \mid G) \,\|\, \mathcal{N}(0, I)\big),$$

Total Objective: $\mathcal{L}_{total} = \mathcal{L}_{node} + \lambda_{edge}\mathcal{L}_{edge} + \mathcal{L}_{KL},$

Synthetic Trace Generation

Sampling from Prior

Sample latents from standard Gaussian (no encoder needed):

Decoding Process

Decoder reconstructs node attributes and edges from latent variables

Graph Construction

Edge logits thresholded to form DAG; causal ordering enforced

Generation Modes

Fixed-Size Generation

|V| constant — isolates node attribute quality and dependency reconstruction

Variable-Size Generation

|V| sampled from empirical distribution — evaluates structural coherence across heterogeneous depths

RQ1: FIDELITY ANALYSIS:
RECONSTRUCTION FIDELITY RESULTS

Service Accuracy

100% SocialNetwork

99.90% TrainTicket

Operation Accuracy

99.91% SocialNetwork

99.92% TrainTicket

Edge F1-Score

97.65% SocialNetwork

86.10% TrainTicket

Detailed Performance Metrics

Metric	SN	TT
Total Nodes Evaluated	11,649	7,912
Service Accuracy	**100%**	**99.90%**
Operation Accuracy	**99.91%**	**99.92%**
Duration MAE	1,653,559 ms	1,561 ms
Edge Precision	95.40%	78.28%
Edge Recall	100%	95.65%
Edge F1-Score	**97.65%**	**86.10%**

Key Findings

- The encoder captures **compact yet expressive representations** of both node attributes and execution dependencies
- **High reconstruction fidelity** demonstrates the model preserves information necessary for structurally valid traces
- Edge-level metrics validate the decoder's ability to reconstruct **directed execution dependencies**

RQ1 Finding

The hierarchical VAE preserves **essential structural and feature-level properties** with high fidelity, providing a reliable foundation for synthetic trace generation.

RQ2: DOWNSTREAM UTILITY:

TRAIN-ON-SYNTHETIC-TEST-ON-REAL RESULTS

Synthetic-Only Training

72-78%

Accuracy on Real Data

Hybrid Training (10% Real)

99-100%

Accuracy on Real Data

Performance Improvement

~25%

With Minimal Real Data

Hybrid Training Performance (10% Real + Synthetic)

Metric	Fixed-Size	Variable-Size
Test Loss	0.0121	0.0136
Test Accuracy	**99.8%**	**99.9%**
SN Accuracy	99.6%	99.7%
TT Accuracy	100%	100%
SN Precision/Recall/F1	1.00/0.996/0.998	1.00/0.997/0.999
TT Precision/Recall/F1	0.996/1.00/0.998	0.997/1.00/0.999
False Positives (SN→TT)	4	3
False Negatives (TT→SN)	0	0

Evaluated on held-out test sets under distributional shift

Key Insights

Generalization Under Distribution Shift

Models trained on synthetic data **generalize to unseen real traces** from different workloads without retraining

Small Real Data Anchoring Effect

Just **10% real traces** combined with synthetic data yields near-perfect classification performance

Workload-Discriminative Structure

Synthetic traces preserve **stable execution patterns** rather than dataset-specific artifacts

RQ2 Finding

Synthetic traces can **effectively replace real traces** for initial model training and substantially reduce real data requirements, particularly when combined with limited production traces.

RQ3 & RQ4: ROBUSTNESS & SEPARABILITY ANALYSIS:

VARIABILITY & SEPARABILITY ANALYSIS

Fixed-Size vs Variable-Size Comparison

Metric	Fixed	Variable
Overall Accuracy	82.6%	83.4%
SN Precision	78.8%	86.8%
SN Recall	89.3%	78.7%
SN F1	83.7%	82.5%
TT Precision	81.4%	87.7%
TT Recall	84.1%	80.5%
TT F1	**82.5%**	**88.0%**

RQ3 Finding: Variable-size generation improves downstream robustness and class balance

Key Observations

Trade-off: Fixed-size promotes stability; variable-size improves robustness across heterogeneous workloads

Exposure: Variable-size exposes algorithms to broader range of execution depths and branching

Clustering Analysis (NMI Scores)

Lower NMI = stronger mixing between real and synthetic samples

SN Fixed-Size — **NMI = 0.072**

TT Fixed-Size — **NMI = 0.117**

SN Variable-Size — **NMI = 0.142**

TT Variable-Size — **NMI = 0.142**

Instance-Level Similarity

Average Overall Similarity:

20.6%	26.2%	30.5%	63.9%
SN Var	TT Var	SN Fixed	TT Fixed

RQ4 Finding:

Strong feature-space overlap; difficult to distinguish using unsupervised techniques

PRACTICAL IMPACT:
INDUSTRIAL DEPLOYMENT & LESSONS LEARNED

Key Lessons Learned

Bootstrap
Use synthetic when real data is limited

Combine
Add small fraction of real data

Prefer Variable
When robustness is priority

Avoid Over-Optimization
Don't sacrifice diversity for indistinguishability

Deployment Scenarios

Cold-Start Training

Bootstrap models when **insufficient production traces** have been collected. Synthetic-only training achieves non-trivial generalization.

Privacy-Preserving Sharing

Enable **cross-team collaboration** without exposing sensitive execution details. Synthetic traces preserve behavioral characteristics safely.

Testing & Benchmarking

Provide **controllable yet realistic workloads** for evaluating AIOps pipelines. Variable-size generation tests robustness to heterogeneous patterns.

Conclusion:
Future Work

Key Findings

- **Hierarchical VAE** provides viable foundation for privacy-aware trace synthesis
- **Synthetic traces generalize** effectively to real-world data under distribution shift
- **Hybrid training** (10% real) yields near-optimal 99-100% performance
- **Variable-size generation** improves robustness across heterogeneous workloads

Future Directions

Richer Temporal Dynamics

Incorporate resource metrics, cross-trace dependencies for performance diagnosis

Online Integration

Integrate into continual learning pipelines for evolving workloads

Production Deployment

Large-scale validation with industry partners under real operational constraints

Overall: Hierarchical generative modeling provides a **viable and scalable foundation** for privacy-aware distributed trace synthesis, with clear applicability to real-world observability and AIOps pipelines.

Source code available at

github.com/sneh2001patel/distributed_trace_research