What deep generative models can do for you: Opportunities, challenges, and open questions

Giulia Fanti
Carnegie Mellon University

Zinan Lin

Hao Liang

Kiran Thekumparampil

Alankar Jain

Sewoong Oh

Vyas Sekar

Chen Wang

Carnegie Mellon University

UNIVERSITY of WASHINGTON

IBM

Common ML tools in networking

Classification	Classifying network traffic
Reinforcement learning	Traffic engineering
Unsupervised methods	Clustering signals

This talk: Generative models

- What are generative models?
- Why are they relevant now?
- How can they be useful in networking?
- What are the limitations?

What is a generative model?

- Models the joint probability distribution $p(x)$ of a dataset
- Example:

$$x[t] = \boxed{f(x[0{:}\,t-1], \theta) + n[t]}$$

Model

Learned parameters

Noise

How do we pick f?
How to combine noise?

How are they used in the networking community?

Use **domain knowledge** to extract high-level insights

→

Design **parametric model** to model those insights

→

Use **data** to populate parameters

Network traffic has temporal patterns

$$x[t] = \sin(\theta t) + n[t]$$

$\frac{1}{\theta}$ = 1 day

Melamed (1993), Denneulin et al (2004), Swing, BURSE, Hierarchical bundling, Di et al (2014), …

Problems with this approach

Poor flexibility

- Requires new design for every type of data

Poor fidelity

- Doesn’t capture properties that were not explicitly modeled

Deep generative models

Design neural network to produce data of the right dimensionality

Use data to populate parameters

$\theta \in R^d$

Generative Adversarial Networks (GANs): Breakthrough in generative modeling

- Prior approaches
  - Likelihood-based
  - Heavily rely on domain knowledge
- GANs
  - Adversarial learning
  - Limited *a priori* assumptions

Generative Adversarial Networks (GANs)

How can we use these tools in networking?

- Sharing synthetic data
- Discovering malicious inputs to black-box systems
- Understanding complex datasets

Zinan Lin

Alankar Jain

Chen Wang

Vyas Sekar

Sharing synthetic data

Use case 1

github.com/fjxmlzn/DoppelGANger

Key stumbling block: Access to data

Collaborative opportunities go untapped

Unreproducible research
Limited potential

(Not a new) idea: Synthetic data models

Two main problems

Fidelity

Privacy

Existing methods

What kinds of data are we interested in?

Multi-dimensional time series

With metadata

`(U.S., mobile traffic)`

Datasets: Networking, security, and systems

- Cluster traces
  - **Google:** task resource usage logs from 12.k machines (2011)
  - **IBM:** resource usage measurements from 100k containers

- Traffic measurements
  - **Wikipedia web traffic:** # daily views of Wikipedia articles (2016)
  - **FCC Measuring Broadband America:** Internet traffic and performance measurements from consumer devices around the country

DoppelGANger: Time series generation

Part I: RNN + Batched Generation

Challenge: Training on high-dynamic-range time series

Part II: Auto normalization

- Standard normalization: Normalize by **global** min/max

- DoppelGANger: Normalize each timeseries individually
  - Store min/max as “fake” metadata

Mode-collapsed
Normalized page views
1
0
−1
0
100
200
300
400
500
Auto-normalized
Normalized page views
1
0
−1
0
100
200
300
400
500

Challenge: Complex relationships in metadata

- Need to capture relation between metadata and time series
  - E.g., Cable vs Mobile users

- Straw man: Joint generator of metadata and time series
  - Problem: too hard for a single generator

Before:
Single generator

Part III:
Decoupled Generation, Auxiliary Discriminator

- Two stage decoupling
  - Generate metadata (using a standard MLP)
  - Generate measurements conditioned on metadata
- Auxiliary discriminator for metadata alone

Histogram of $\frac{\max+\min}{2}$ per time series

Without auxiliary discriminator

With auxiliary discriminator

Putting it together

Temporal Correlations
Microbenchmark

Predicting job failures in a compute cluster
Downstream task

- Train on synthetic, test on real

Evaluating privacy

- Protecting business secrets
  - Aggregate functions of the data

- User privacy
  - Differential privacy
  - Robustness against membership inference

Differentially-private SGD kills fidelity in GANs

Open questions: Synthetic data generation

- **Fidelity**
  - Long sequences of data
  - Stateful protocols

- **Privacy**
  - Differentially-private GANs
  - New privacy metrics?

Zinan Lin

Hao Liang

Vyas Sekar

Identifying malicious inputs to black-box systems

Use case 2

Black-box Devices and Systems Abound

IoT Devices

Servers / Routers

Control Units in Vehicles / Manufacturing

NO source code / binary / protocol format / design doc

Identifying Attack Packets is Hard

We want to identify attack packets,
but do NOT have source code or system description

Motivating example

- Packet classification
  - Vamanan et al [SIGCOMM 2010]
  - Singh et al [SIGCOMM 2013]
  - Yingchareonthawornchai et al [TON 2018]
  - Liang et al [SIGCOMM 2019]
  - Rashelbach et al [SIGCOMM 2020]
  - Many more…

Can an attacker identify **many** packets with **high** classification times?

Classification Time

Random packet generation

- NeuroCuts, Liang et al [SIGCOMM 2019]

Can can we generate **many, diverse slow** packets?

2,000 total packets

Common approaches

- Fuzzing tools
  - Random sampling
- Optimization of black-box functions
  - Bayesian optimization
  - Genetic algorithms
  - Simulated annealing

GANs can help!

Approach 1: Vanilla GAN

- Challenge: too little training data

AmpGAN: Training with Feedback

Results

Number of packets

Number of packets

Random packets

AmpGAN

Results

AmpGAN
10x jump
2.5x jump
Fraction of
"slow" packets
10^0
10^{-1}
10^{-2}
10^{-3}
2k
100k
System Calls
Genetic Algorithms
Simulated Annealing
Generalized SA
Bayesian Optimization
AmpGAN

Open questions

- Sequences of inputs

- Can we use this to optimize systems as well as finding attacks
  - E.g., CherryPick [NSDI 2017]

Zinan Lin

Kiran Thekumparampil

Sewoong Oh

Extracting insights from unstructured data

Use case 3

github.com/fjxmlzn/InfoGAN-CR

Disentangled GANs

d input noise

z_1
z_2
...
z_d

Generator

k factors

- Hair color
- Rotation
- Background
- Bangs

- How do z_is control the factors?

Vanilla GANs

z_1
z_2
...
z_d

Factor_1
Factor_2
...
Factor_k

Disentangled GANs

$c_1 \longrightarrow \text{Factor}_1$
$c_2 \longrightarrow \text{Factor}_2$
...
$c_k \longrightarrow \text{Factor}_k$
z_is

Code & Paper: https://github.com/fjxmlzn/InfoGAN-CR

Examples of Disentanglement

* CelebA example is generated by InfoGAN-CR. Dsprites example is synthetic for illustration.

Our Solution: Contrastive Regularizer (CR)

- Use two latent codes $(c_1,\ldots,c_i,\ldots,c_k)$, $(c_1',\ldots,c_i',\ldots,c_k')$ to generate a pair of images

Same i-th latent code

Intuition of Contrastive Regularizer (CR)

- Use two latent codes $(c_1,\ldots,c_i,\ldots,c_k)$, $(c_1',\ldots,c_i',\ldots,c_k')$ to generate a pair of images

Same i-th latent code

Classification task!

InfoGAN-CR

InfoGAN-CR loss: $$\min_{G,Q,H} \max_{D} \underbrace{\underbrace{L_{adv}(G,D)}_{\text{GAN's adversarial loss}} - \underbrace{\lambda I(G,Q)}_{\text{Mutual info loss}}}_{\text{InfoGAN [1]}} - \underbrace{\alpha L_c(G,H)}_{\text{classification accuracy of CR}}$$

[1] InfoGAN. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (NeurIPS 2016)

Key open question

- Can disentanglement help us make sense of networking data?
  - Reverse-engineer protocols
  - Categorize complex data patterns
  - Has not been tried in networking domain
- Time series disentanglement
  - Himberg, Hyvärinen, Esposito (2004)

Take-home messages

- Deep generative models show promise for networking applications
  - Synthetic data generation
  - Identifying malicious packets for black-box systems
  - Extracting structural insights from data

- They often cannot be applied off the shelf
  - New architectures
  - Data pre-processing pipelines
  - Training mechanisms