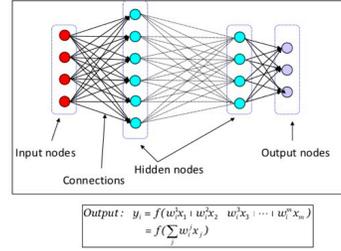


Analyzing the Training Processes of Deep Generative Models

Mengchen Liu, Jiaxin Shi, Kelei Cao, Jun Zhu, Shixia Liu

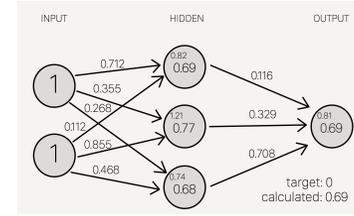
Presented by:
Gursimran Singh
Department of Computer Science
University of British Columbia

Background – deep neural networks



Background – dnn training

• Training in deep neural networks



What to visualize - data

- Network diagram of neurons and its characteristics
 - activations, gradients, weights of each neuron [MILLIONS]
- Summary statistics of the model
 - loss/ accuracy of the model
- Dynamic time-series data [MILLIONS]
 - how activations/ gradients/ weights changes over time
 - how loss/ accuracy change over time

Why visualize – after iterative user studies

- Identify root cause of a failed training process
 - Loss function can become NaN or Inf
 - Multiple sources -> inappropriate network structure, bug in code, lack of numerical stability
- R1: Connecting the overall statistics with detailed training dynamics
 - Summary statistic serves as entry point
 - Detailed training dynamic can be used for root cause analysis

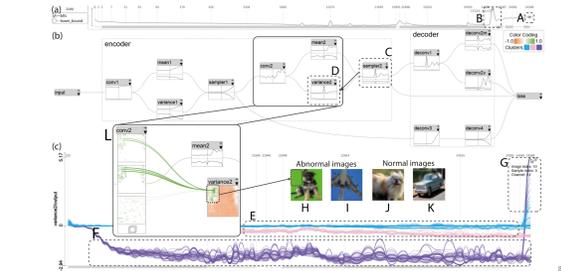
Why visualize – after iterative user studies

- R2: Examining how data flows through the network
 - Different layers has different roles in DGM
 - A failed training process is usually caused by a specific layer
- R3: Facilitating the detection of outliers
 - Outliers in training process is a potential reason for failed training process
- R4: Examining how neurons interact with each other
 - Poor understanding on how neurons interact with each other in a DGM
 - Even after identification of abnormal; neuron, it is hard to identify the root cause.

How to visualize

- Data flow visualization module
 - How data flows in a DGM (R2)
 - How other neurons influence the output of the neuron of interest (R4)
 - Credit assignment problem
- Training dynamics analysis module
 - Loss as a function of time (R1)
 - Samples time series to preserve outliers (R3)
 - Blue-noise polyline sampling to reduce visual clutter

Overview



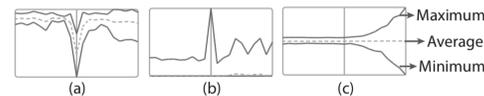
Snapshot level analysis

- Idea - which layer caused the issue?
- DAG - illustrates how layers are connected
 - Hierarchical structure to reduce clutter in case of 100s of layers



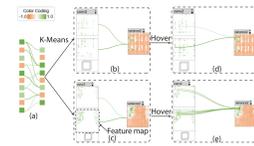
Snapshot level analysis

- Line charts – how data flows
 - Averaged training dynamics within a certain configurable window
 - Center line represents the focus snapshot

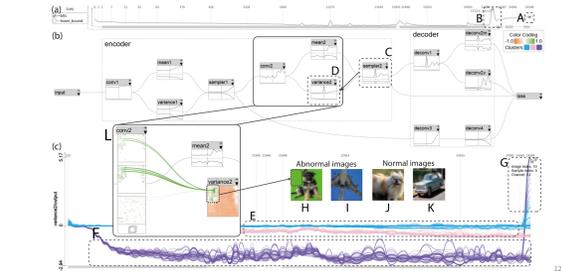


Neuron level analysis

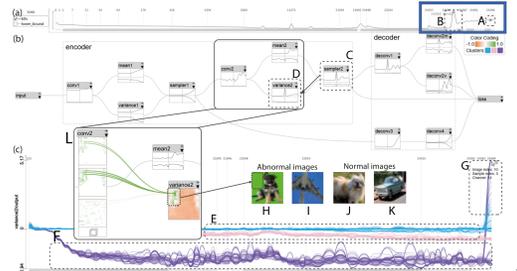
- Idea - Root cause analysis
- Credit assignment
 - Influenced by both previous and next layer
 - Use forward and backward propagation to find neurons of interest



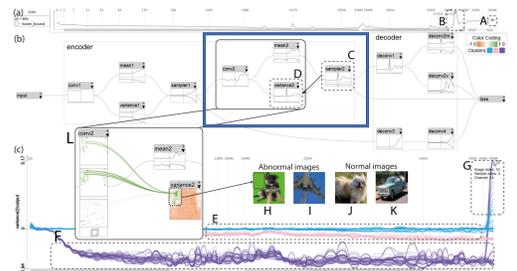
Stitching it all together



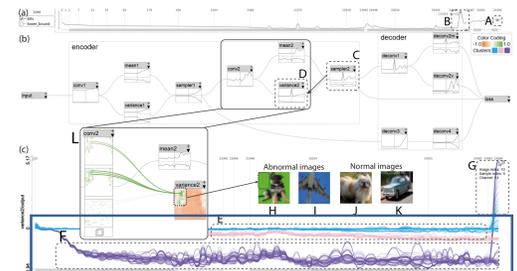
Stitching – Examine the loss function



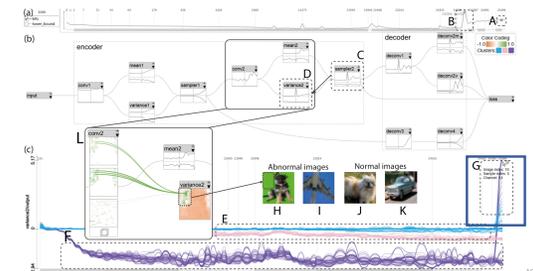
Stitching – Examine the layers (snapshot)



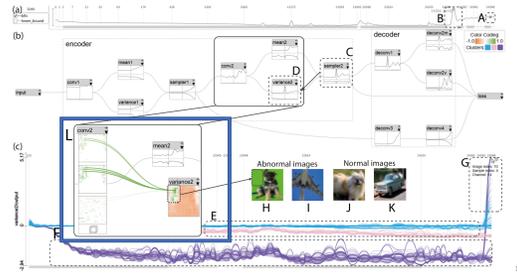
Stitching – Training dynamics of the layer



Stitching – Training dynamics of the layer



Stitching – Neuron level root cause analysis



Analysis - scalability

- Millions of neurons in each layer and 100s of layers
 - In snapshot level view, use expandable hierarchy groups
 - In layer level view, use filtering, clustering
 - In neuron level view, use filtering and clustering
- Large amount of training dynamics data [5TBs]
 - Keep a few in memory
- Millions of snapshot-level data come as time-series
 - Use sampling (blue-noise polyline)

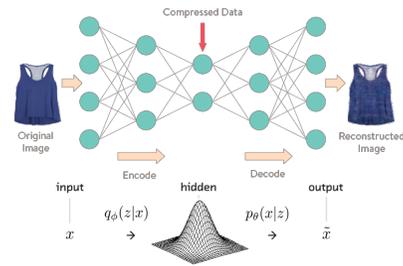
Analysis - critique

- Pros
 - Case studies to prove that it works
 - User study to understand the requirements
 - Extensible to CNNs and MLPs
- Cons
 - Does not work in case neural networks have cycles
 - Works only for offline analysis
- My views
 - Not sure of responsiveness
 - Not sure what is the level of change required for other NNs

Conclusion/ Highlights

- Effective tool to diagnose DGMs
- Three level of analysis in DGMs
 - Snapshot level
 - Layer level
 - Neuron level
- Idioms
 - focus+context, filtering, aggregation, interaction
 - blue-noise polyline sampling to reduce clutter

DGMs – VAE (variational auto encoders)



What to visualize - challenges

- Handle large amount of time series data from the training process
 - Activation/ gradient/ weight changes over time (training dynamics)
 - Millions of time series (as DGMs has millions of above)
 - Simple viewing => visual clutter

Overview

- Examine loss changes to identify abnormal snapshot
- High level averaged statistics of each layer to identify layer of interest (hybrid viz) at the **snapshot level**
- Print the training dynamics of the layer of interest (**layer level analysis**)
- Interactively select a set of neurons and explore how other neurons are related through the **data-flow visualization**

Neuron level analysis

- Idea - Root cause analysis
- Credit assignment
 - Influenced by both previous and next layer
 - Use forward and backward propagation to find neurons of interest
- Other ideas for conv/ deconv layers
 - Relative position of the image patch that each neuron is influenced
 - Represent neurons as feature maps - neurons which share the same weights
 - By tracing behind, we identify the corresponding image patch.

Neuron Level analysis

- Visualization
 - Cluster neurons and show them in a grid form
 - Show only which highly contribute to the output

