Stochastic Bottleneck:
Rateless Auto-Encoder for Flexible Dimensionality Reduction

Toshiaki Koike-Akino
Ye Wang

June 2020

MITSUBISHI ELECTRIC RESEARCH LABORATORIES (MERL)
Cambridge, Massachusetts, USA
http://www.merl.com
• Motivations
  – Machine learning for real-world data analysis

• Dimensionality reduction
  – Principal component analysis (PCA)
  – Auto-encoder (AE)

• Rateless property
  – Fountain codes

• Stochastic bottleneck
  – Stochastic Width vs. Stochastic Depth
  – TailDrop regularization

• Multi-objective learning

• Experiments
  – MSE
  – SSIM
  – Accuracy

• Summary

How many latent variables required?
Emerging Technologies

- Gartner’s Hype Cycle for Emerging Technologies, 2019 August
Deep Learning for AI

  - 2006 Hinton: Many layers, layer-wise pre-training, massive data sets

- Massively parallel computation
  - Driver: graphic processor units, tensor processor units ...

- Variants:
  - Deep belief networks
  - Deep convolutional networks
  - Deep recurrent networks
  - Deep Boltzmann machines
  - Deep autoencoder
Deep Learning for Media Signal Processing

• Audio & Visual Applications
AI Surpassing Human-Level Performance

Computer won world champion of chess
(Deep Blue) (Garry Kasparov)

May 11th, 1997

DARPA Grand Challenge
Autonomous Vehicle Races

DGC I
Barstow to Primm
March 13, 2004
142 miles
10 hours
$1M

DGC II
Desert Classic
October 8, 2005
132 miles
10 hours
$2M

DGC III
Urban Challenge
November 3, 2007
60 miles
6 hours
$3.5M

Nature

AlphaGo Zero 40 blocks
AlphaGo Master
AlphaGo Lee

© MERL June 2020; T. Koike-Akino and Ye Wang, Stochastic Bottleneck
Applied Deep Learning

• AI has been applied to various fields

- Wireless Communication
- Optical Communication
- Networked Control
- Localization Navigation
- Device / Integrated Circuit
- Tomography Imaging
- Bio-Sensing Human Interface

© MERL June 2020; T. Koike-Akino and Ye Wang, Stochastic Bottleneck
Moore’s Law: Exponential Grow in Applications

- The hit count of articles per year in GoogleScholar; Wireless Communication applications.

![Graph showing exponential growth in articles with Machine Learning and Deep Learning categories.](image-url)

- Machine Learning: 212% annually.
- Deep Learning: 124% annually.

Machine Learning + Wireless Comm
Deep Learning + Wireless Comm
Exponential Prediction
High-Dimensional Real-World Data

• Raw data dimensionality is often extremely large
Curse of Dimensionality

• Data-space volume exponentially increases with dimensionality
Hughes Phenomenon

- Classifier performance drops for high-dimension data with finite training samples

Hughes Phenomenon (Hughes, 1968)

or so called curse of dimensionality, peaking phenomenon

Accuracy Degrades

Higher Dimensionality
Dimensionality Reduction

- Principal component analysis (PCA)
- Kernel PCA
- Independent component analysis (ICA)
- Isomap
- Local linear embedding (LLE)
- Auto-encoder (AE)
- …
Reduced-Dimension Feature

- High-dimensional data may be well-described by lower-dim latent variables
Auto-Encoder (AE): Bottleneck Network

- **Bottleneck** neural network architecture: $M < N$
- Encoder and decoder networks are jointly trained such that latent variables can regenerate original data with smallest distortion

\[
\min_{\theta, \phi} \mathbb{E}_{x \sim \Pr(x)} \left[ \mathcal{L}(x, g_\phi(f_\theta(x))) \right]
\]

Original data \( x \in \mathbb{R}^N \)

Latent variable \( z \in \mathbb{R}^M \)

Encoder network \( z = f_\theta(x) \)

Decoder network \( x' = g_\phi(z) \)

Loss function (e.g. MSE) \( \mathcal{L}(x, x') \)
AE as Nonlinear PCA (NLPCA)

- AE is often called NLPCA due to analogy
- Without nonlinear activations, an optimal AE model coincides with PCA for Gaussian data under MSE distortion (Karhunen-Loeve)

Encoder Affine Transform

\[ f_\theta(x) = Wx + b \]

Decoder Affine Transform

\[ g_\phi(z) = W'z + b' \]

Consider Gaussian data

\[ x \sim \mathcal{N}(m, C) \]

Covariance EVD

\[ C = \Phi \Lambda \Phi^T \]

Eigen projection gives minimum MSE

\[ \Lambda = \text{diag} [\lambda_1, \lambda_2, \ldots, \lambda_N] \]

\[ \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \geq 0 \]

\[ \mathcal{L}_M = \mathbb{E}_x \left[ \| W'(Wx + b) + b' - x \|^2 \right] = \sum_{n=M+1}^{N} \lambda_n \]

© MERL June 2020; T. Koike-Akino and Ye Wang, Stochastic Bottleneck
PCA: Eigen-Spectrum

- Random matrix theorem:
  If covariance matrix follows i.i.d. Gaussian Gram matrix, eigenvalue distribution follows Marchenko-Pastur distribution

\[ \mu(A) = \begin{cases} (1 - \frac{1}{\lambda})1_{0 \in A} + \nu(A), & \text{if } \lambda > 1 \\ \nu(A), & \text{if } 0 \leq \lambda \leq 1, \end{cases} \]

\[ \bar{L}_M = \mathbb{E}_x \left[ \|W'(Wx + b) + b' - x\|^2 \right] = \sum_{n=M+1}^{N} \lambda_n \]

Cumulative is well approximated by exponential
Rateless Property

- PCA universally achieves best MSE for all dimensionality $1 < M < N$ under Gaussian datasets
- The downstream users can freely change the dimensionality by discarding the least-principal components or appending the most-principal components without changing encoder and decoder models
- The MSE is gracefully improved by increasing the compression rate $M/N$
- We do not need to pre-determine the dimensionality when training the model
- This rateless property can resolve the issue:

How many latent variables do we need for training the AE model?
Coding Theory: Rateless Channel Codes

• Capacity approaching codes need to pre-determine code rates under the knowledge of channel capacity.

- Single Rateless Code
- Multiple LDPC codes with different rates
Rateless Fountain Codes

• Continue sending more redundant parity until the user satisfies
  – Luby-Transform (LT) codes [2002], Online codes [2002],
    Raptor codes [2006], Tornado codes [2004]

• We do not pre-determine the code rates

• Rateless codes are capacity-achievable

• We introduce “rateless” AE which does not have to determine the dimensionality beforehand
Dimensionality Flexibility

- For PCA, principal components are sorted in significance, thus scalable
  - For AE, latent variables are equally important, thus not adaptable

- Once AE is learned with pre-determined dims, it requires another learning to reduce or expand dims
  - **Hierarchical AE** (hAE) to append dim for residual reconstruction
  - **Stacked AE** (sAE) to further reduce dimensionality

- Conditional update for progressive learning usually does not work best and often finite-tuning is required while flexibility is compromised

- We propose a very simple **dropout** mechanism to realize ratelessness
Dropout Regularization

- Dropout is an effective method to prevent **over-training** by regularizing over-parameterized networks.

- It can be viewed as **Bayesian** approximation [Gal2016].

- There are many different regularization techniques: DropConnect, DropBlock, StochasticDepth, DropPath, ShakeDrop, SpatialDrop, ZoneOut, Shake-Shake, etc.
Proposal: Stochastic Bottleneck

- Simple idea: Non-uniform dropout mechanism
Non-uniform dropout has been used in StochasticDepth for ResNet.

Not only depth direction, we use width direction to concentrate important feature in upper neurons.

(a) Stochastic Depth

(b) Stochastic Width (Independent)

(c) Stochastic Width (Tail Drop)

Drop random burst length of tail.
We tested various eigenspectrum model: Poisson, Laplacian, exponential, sigmoid, Lorentzian, polynomial, and Wigner distribution.

Power cumulative mass function (CMF) showed a good tradeoff between distortion and compression rate.

Best power order parameter is chosen dependent on datasets.

\[ \Pr(D < \tau M) = \tau^\beta \]
Multi-Objective Learning

• Rateless objective is multi-task learning

Single:
\[
\min_{\theta, \phi} \mathbb{E}_{x \sim \text{Pr}(x)} \left[ \mathcal{L}(x, g_\phi(f_\theta(x))) \right]
\]

Multi:
\[
\min_{\theta, \phi} \left[ \tilde{\mathcal{L}}(\theta, \phi; 1), \tilde{\mathcal{L}}(\theta, \phi; 2), \ldots, \tilde{\mathcal{L}}(\theta, \phi; M) \right]
\]

\( \tilde{\mathcal{L}}(\theta, \phi; L) \): Expected loss when the first \( L \) latent variables retained by user

Simple multi-objective optimization with weighted sum:
\[
\min_{\theta, \phi} \sum_{L=1}^{M} \omega_L \tilde{\mathcal{L}}(\theta, \phi; L)
\]

\[
\text{Pr}(D = M - L) = \omega_L
\]
e.g.) balanced weights: \( \omega_L \approx 1/\tilde{\mathcal{L}}^*(\theta, \phi; L) \)

Weighted metric method, ...
Toy Experiments

• AE architecture
  – 3 layers 1024 or 2048 nodes
  – Adam (0.001)
  – Mini-batch 100
  – Max 500 epochs
  – Power CMF TailDrop

• Datasets
  – MNIST
  – CIFAR-10
  – FMNIST
  – KMNIST
  – SVHN
  – CIFAR-100
MSE Distortion Measure (MNIST)

- Conventional Sparse AE
- Proposed Rateless AE
- Linear PCA
- Proposed
- PCA
- Conv.
MSE vs. Structural Similarity (SSIM)

- MSE does not fully tell perceptual distortion

Original

MSE=0, SSIM=1

MSE=309, SSIM=0.928

MSE=309, SSIM=0.987

MSE=309, SSIM=0.580

MSE=309, SSIM=0.641

MSE=309, SSIM=0.730
SSIM Distortion Measure (MNIST)

![Graph showing SSIM distortion measure for different methods.](Image)

- Negative SSIM Index vs. Latent Dimension
- Methods compared:
  - Conventional Sparse AE
  - Proposed Rateless AE
  - Linear PCA
  - Proposed

PCA and Conv. markers indicate points for Principal Component Analysis and conventional methods, respectively.
Reconstructed Image Snapshots

Original

64-dim

4-dim

Conventional AE

Proposed AE

© MERL June 2020; T. Koike-Akino and Ye Wang, Stochastic Bottleneck
Support Vector Machine (SVM) Classification (MNIST)

Conventional Sparse AE
Proposed Rateless AE
Linear PCA

Classification Error Rate

Latent Dimension
Latent Space Geometry

• The first 2 latent variables
### CIFAR-10

- 32x32 color images
- 10-class natural photos
- 50,000 training
- 10,000 test

<table>
<thead>
<tr>
<th>Category</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>airplane</td>
<td><img src="image" alt="Airplane examples" /></td>
</tr>
<tr>
<td>automobile</td>
<td><img src="image" alt="Automobile examples" /></td>
</tr>
<tr>
<td>bird</td>
<td><img src="image" alt="Bird examples" /></td>
</tr>
<tr>
<td>cat</td>
<td><img src="image" alt="Cat examples" /></td>
</tr>
<tr>
<td>deer</td>
<td><img src="image" alt="Deer examples" /></td>
</tr>
<tr>
<td>dog</td>
<td><img src="image" alt="Dog examples" /></td>
</tr>
<tr>
<td>frog</td>
<td><img src="image" alt="Frog examples" /></td>
</tr>
<tr>
<td>horse</td>
<td><img src="image" alt="Horse examples" /></td>
</tr>
<tr>
<td>ship</td>
<td><img src="image" alt="Ship examples" /></td>
</tr>
<tr>
<td>truck</td>
<td><img src="image" alt="Truck examples" /></td>
</tr>
</tbody>
</table>
MSE Distortion (CIFAR-10)
MNIST vs. CIFAR-10

- MNIST data is gray-scale image, but nearly binary (white or black) whose statistics are far from Gaussian distribution

- CIFAR-10 uses color natural photos. Such photos are well modeled by Gauss-Markov random field (GMRF)

- Hence, PCA surprisingly performs well for CIFAR-10 if we consider MSE distortion

- However, SSIM and accuracy measure ...
SSIM Measure (CIFAR-10)
## Reconstructed Image Snapshots (CIFAR-10)

![Reconstructed Image Snapshots](image)

### Conventional AE vs. Proposed AE

<table>
<thead>
<tr>
<th>Dimensionality $L$</th>
<th>64</th>
<th>54</th>
<th>44</th>
<th>34</th>
<th>24</th>
<th>14</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MSE (dB)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conv. AE</td>
<td>-5.92</td>
<td>-4.96</td>
<td>-3.96</td>
<td>-2.97</td>
<td>-1.91</td>
<td>-0.96</td>
<td>0.92</td>
</tr>
<tr>
<td>Prop. AE</td>
<td>-6.19</td>
<td>-6.43</td>
<td>-6.30</td>
<td>-5.82</td>
<td>-5.11</td>
<td>-4.05</td>
<td>-1.92</td>
</tr>
<tr>
<td><strong>SSIM Index</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conv. AE</td>
<td>0.64</td>
<td>0.61</td>
<td>0.57</td>
<td>0.53</td>
<td>0.48</td>
<td>0.44</td>
<td>0.37</td>
</tr>
<tr>
<td>Prop. AE</td>
<td>0.66</td>
<td>0.67</td>
<td>0.67</td>
<td>0.64</td>
<td>0.60</td>
<td>0.54</td>
<td>0.44</td>
</tr>
<tr>
<td><strong>SVM Acc.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conv. AE</td>
<td>0.47</td>
<td>0.47</td>
<td>0.46</td>
<td>0.44</td>
<td>0.40</td>
<td>0.32</td>
<td>0.20</td>
</tr>
<tr>
<td>Prop. AE</td>
<td>0.47</td>
<td>0.48</td>
<td>0.47</td>
<td>0.48</td>
<td>0.46</td>
<td>0.42</td>
<td>0.29</td>
</tr>
</tbody>
</table>
Example Use Case

Rateless:
- Single unified AE model regardless of dimensionality for application invariant

Moderate dimension we need to diagnose!

Conventional:
- What purpose?
- Dimensionality?
- Which AE models?

All dimensions we need to analyze!

We do not care many but final results
We introduced a new rateless concept in auto-encoder design

We proposed **Stochastic Bottleneck** architecture
- Non-identical dropout rates for **Stochastic Width** and Depth

New regularization called **TailDrop** was investigated

Proposed AE offers an excellent trade-off between distortion and compression rates
- Benefits in MSE, SSIM, and SVM accuracy were confirmed

Demonstrated the benefit for various benchmark datasets

Questions?
- koike@merl.com
- More results in arXiv 2005.02870
More Results in ArXiv

- MNIST
- CIFAR-10
- FMNIST
- KMNIST
- SVHN
- CIFAR-100
Fashion MNIST (FMNIST)

- 28x28 gray-scale images
- 10-class fashion photos
- 60,000 train
- 10,000 test
Snapshots (FMNIST)

Conventional AE

Proposed AE
Kuzushiji MNIST (KMNIST)

- 28x28 gray-scale images
- 10-class ancient Japanese letters
- 60,000 training data
- 10,000 test data
MSE Measure (KMNIST)
Snapshots (KMNIST)

Conventional AE

Proposed AE
Street-View House Numbers (SVHN)

- 32x32 color images
- 10-class cropped digits
- 73,257 training
- 26,032 test
MSE Measure (SVHN)

-3
-4
-5
-6
-7
-8
-9
-10
-11
-12

MSE (dB)

Latent Dimension

1 2 4 8 16 32 64

Conventional Sparse AE
Proposed Rateless AE
Linear PCA
Proposed

Conv.

PCA
Snapshots (SVHN)

Conventional AE

Proposed AE
CIFAR-100

• 32x32 color images
• 100-class natural photos (20 super-classes)
• 50,000 training data, 10,000 test data
MSE Measure (CIFAR-100)

- Conv.
- Proposed
- PCA

Latent Dimension

MSE (dB)

Conventional Sparse AE
Proposed Rateless AE
Linear PCA
Snapshots (CIFAR-100)

Conventional AE

Proposed AE