

Deep Learning IV

Model Evaluation and Open Questions

Russ Salakhutdinov

Machine Learning Department
Carnegie Mellon University
Canadian Institute of Advanced Research



CIFAR
CANADIAN INSTITUTE
for ADVANCED RESEARCH

Talk Roadmap

- Basic Building Blocks:
 - Sparse Coding
 - Autoencoders
- Deep Generative Models
 - Restricted Boltzmann Machines
 - Deep Belief Network, Deep Boltzmann Machines
 - Helmholtz Machines / Variational Autoencoders
- Generative Adversarial Networks
- Model Evaluation

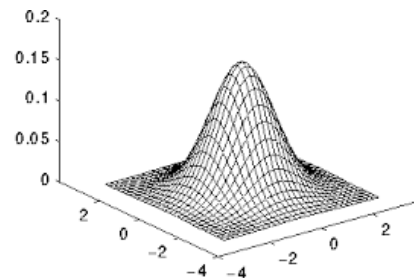
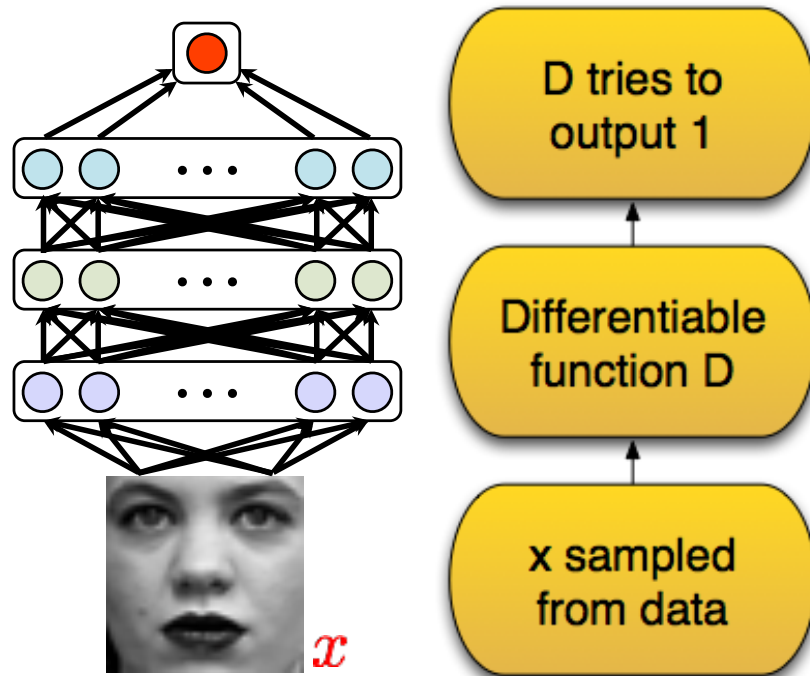
Generative Adversarial Networks

- There is no explicit definition of the density for $p(x)$ – Only need to be able to sample from it.
- No variational learning, no maximum-likelihood estimation, no MCMC. How?
- By playing a game!

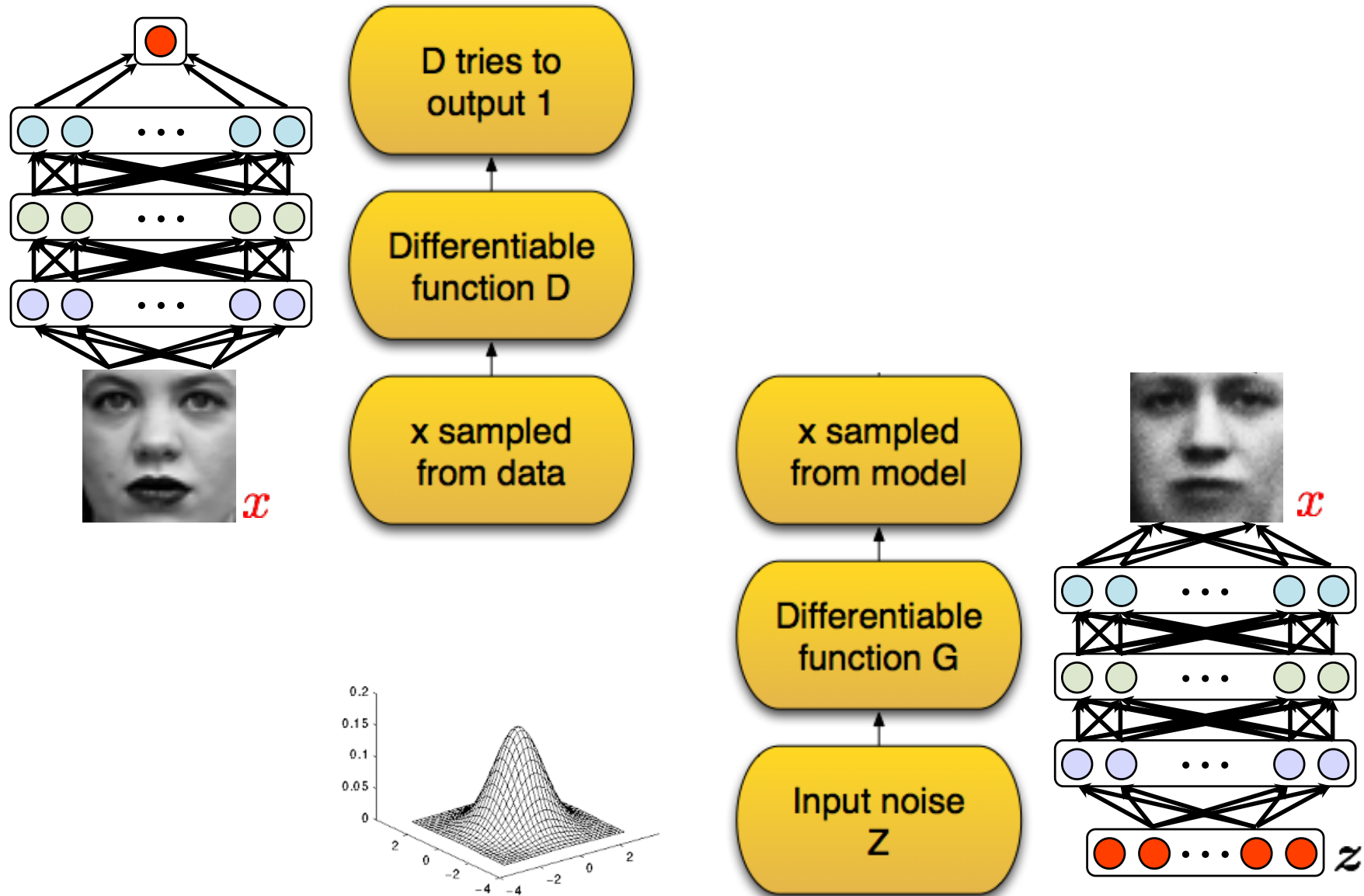
Generative Adversarial Networks

- Set up a game between two players:
 - Discriminator D
 - Generator G
- Discriminator D tries to discriminate between:
 - A sample from the data distribution.
 - And a sample from the generator G.
- The Generator G attempts to “fool” D by generating samples that are hard for D to distinguish from the real data.

Generative Adversarial Networks

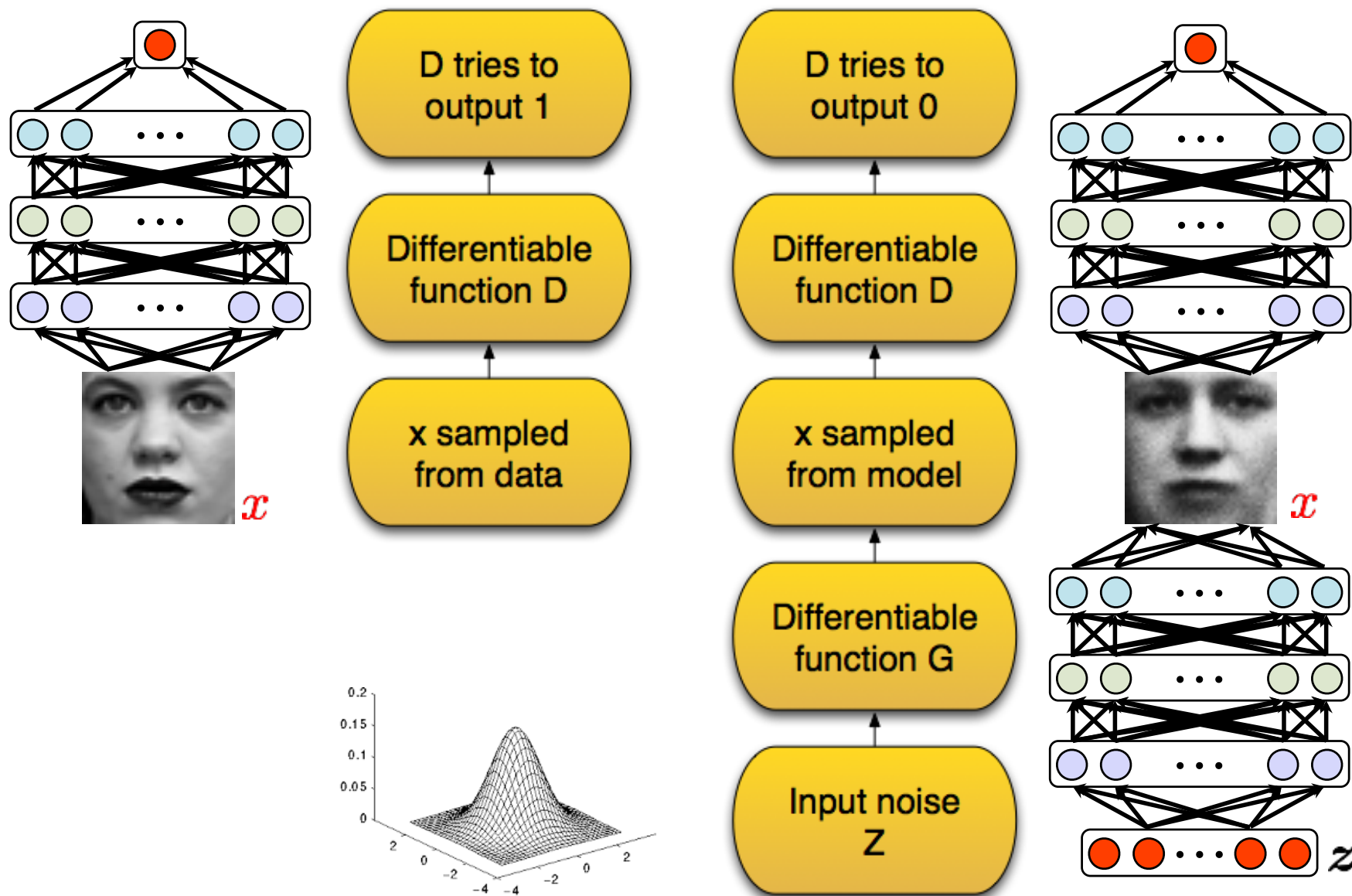


Generative Adversarial Networks



Slide Credit: Ian Goodfellow

Generative Adversarial Networks



Generative Adversarial Networks

- Minimax value function

Generator: generate samples
that D would classify as real

$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))]$$

Discriminator:
Pushes up

Discriminator: Classify
data as being real

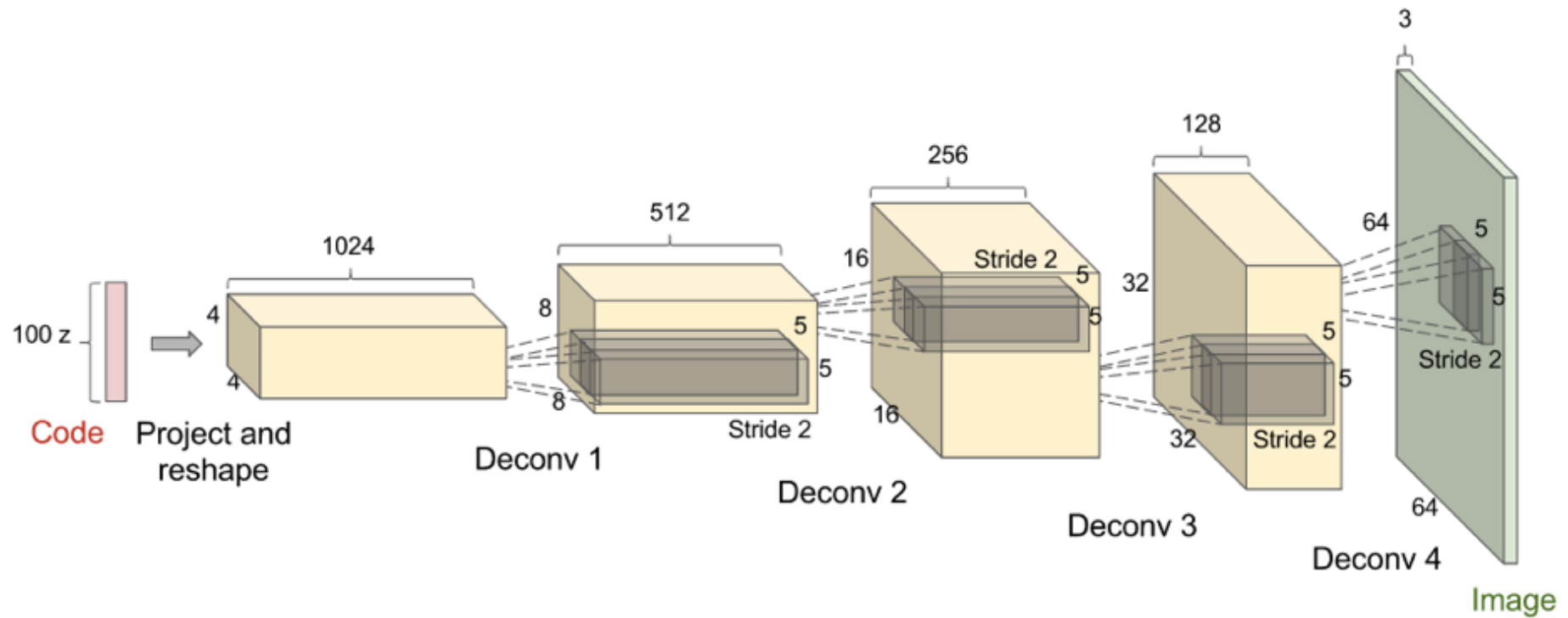
Discriminator: Classify
generator samples as
being fake

Generator:
Pushes down

- Optimal strategy for Discriminator is:

$$D(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{\text{model}}(x)}$$

DCGAN Architecture



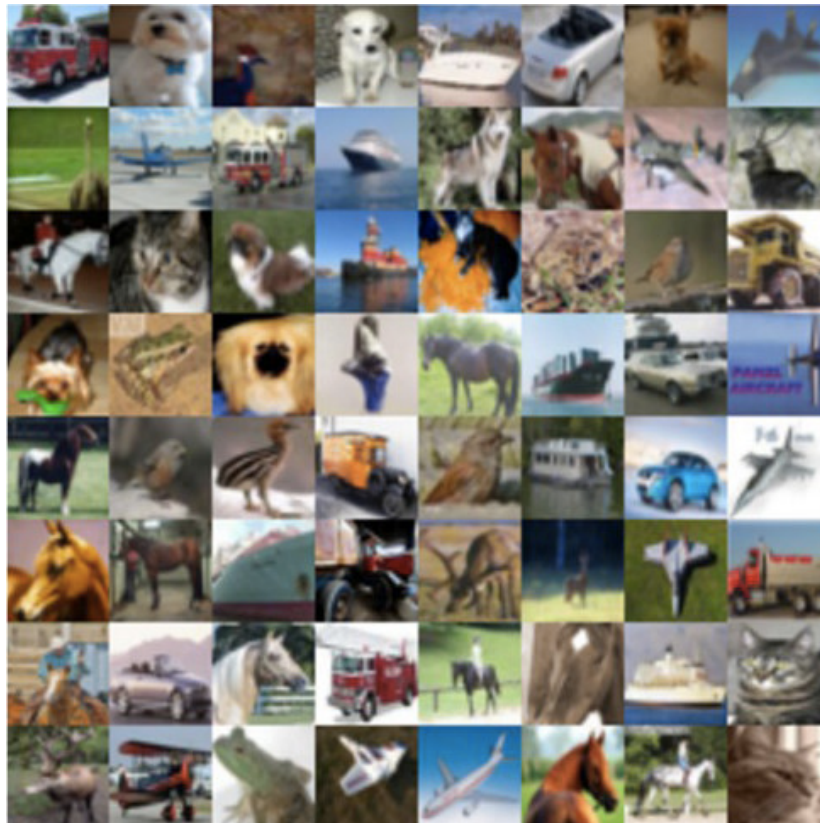
(Radford, Metz, Chintalaet, 2015)

LSUN Bedrooms: Samples

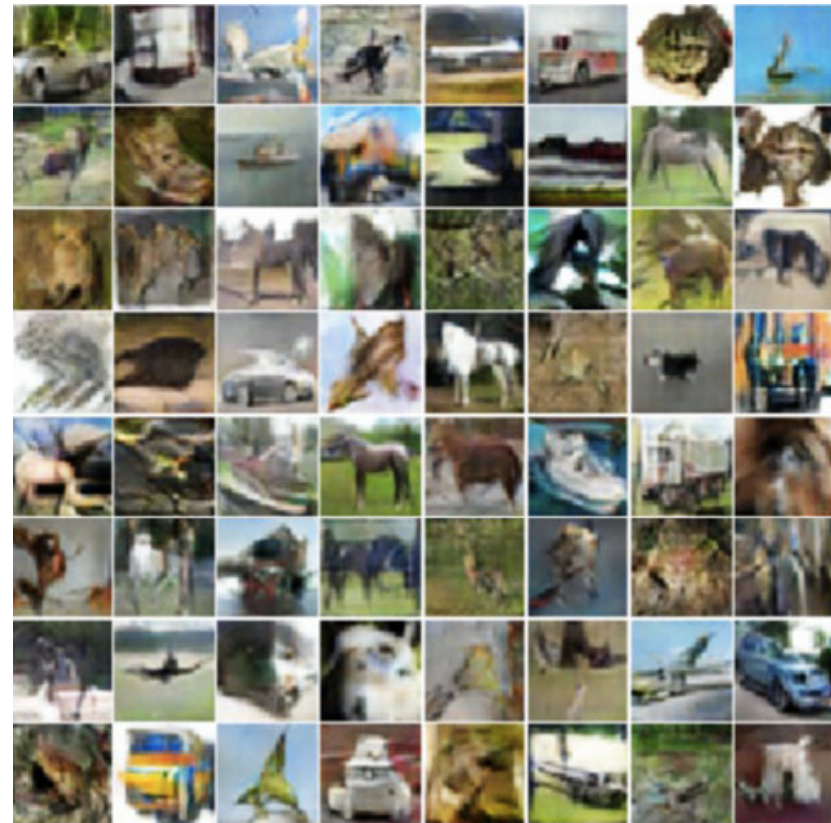


(Radford, Metz, Chintalaet, 2015)

CIFAR

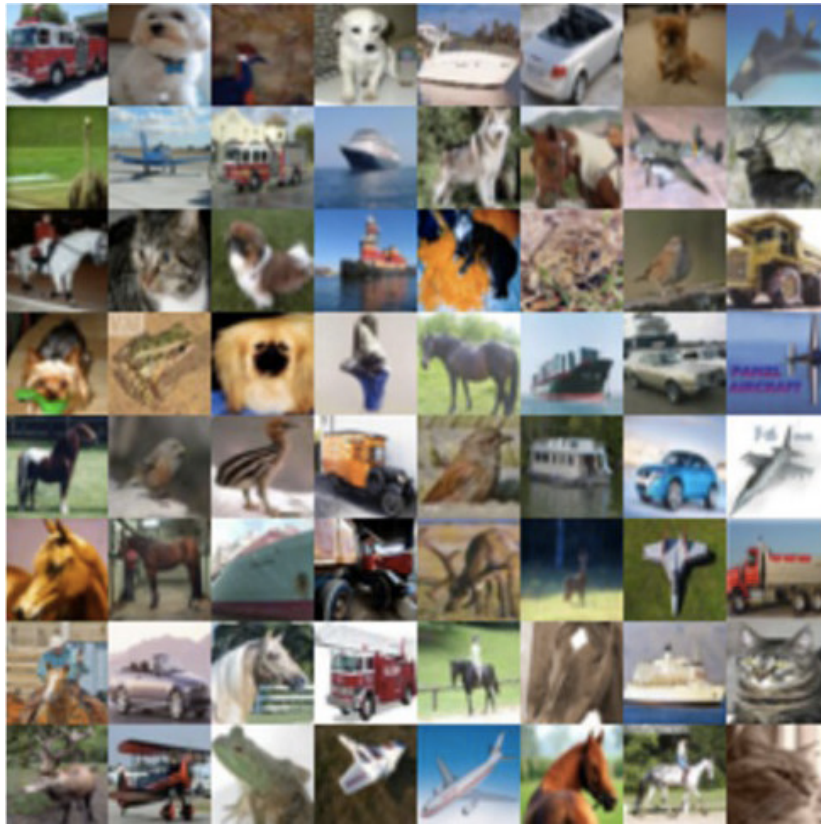


Training

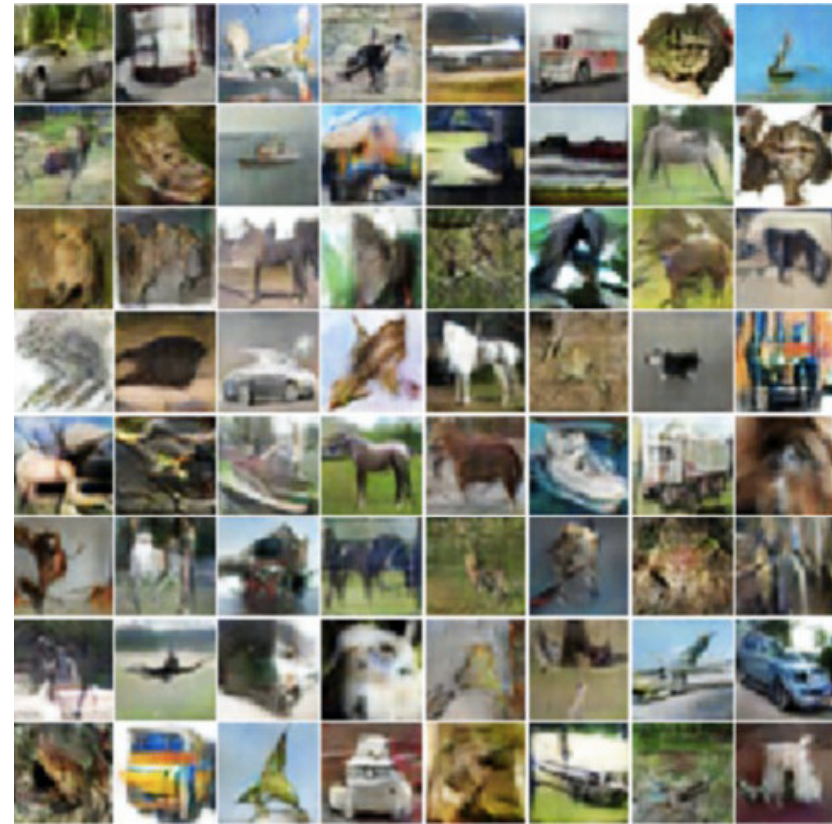


Samples

IMAGENET



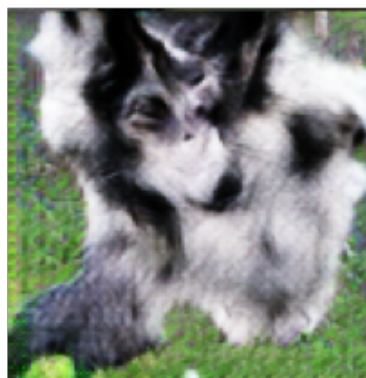
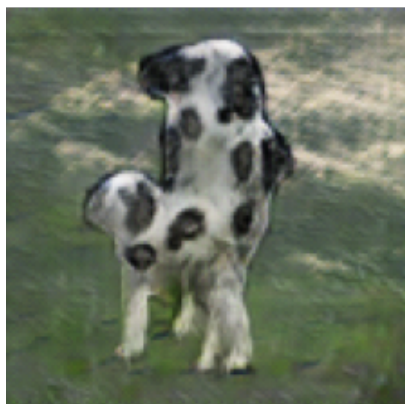
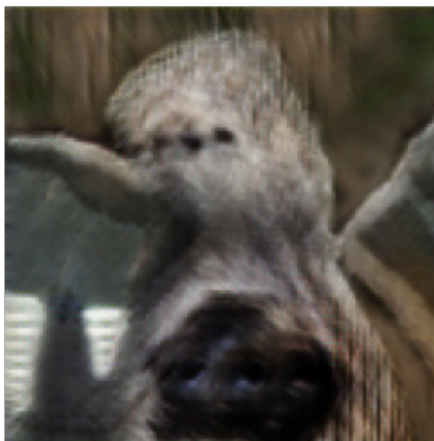
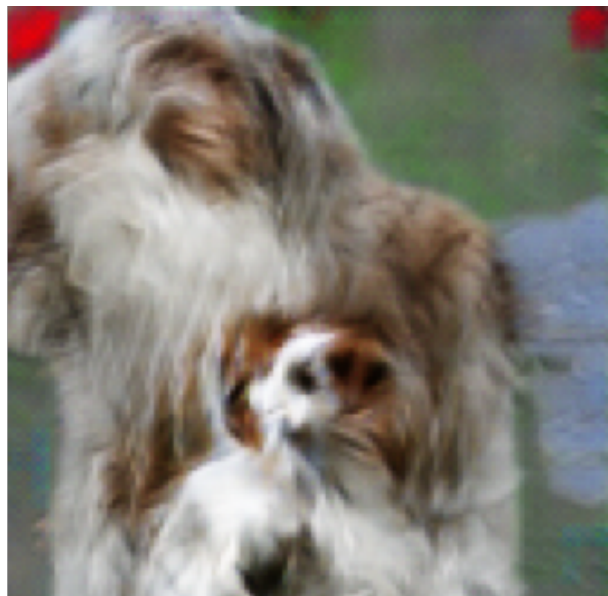
Training



Samples

(Salimans et. al., 2016)

ImageNet: Cherry-Picked Results



- **Open Question:** How can we quantitatively evaluate these models!

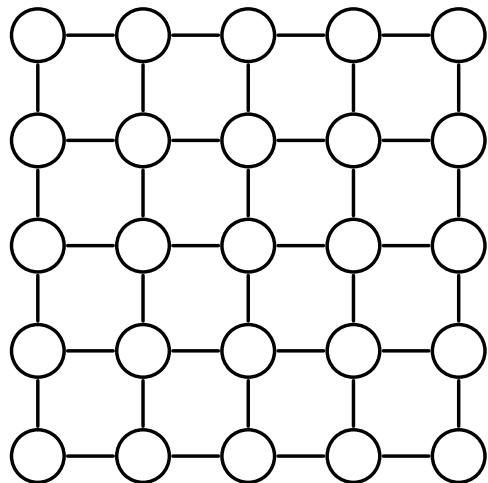
Talk Roadmap

- Basic Building Blocks:
 - Sparse Coding
 - Autoencoders
- Deep Generative Models
 - Restricted Boltzmann Machines
 - Deep Belief Network, Deep Boltzmann Machines
 - Helmholtz Machines / Variational Autoencoders
- Generative Adversarial Networks
- Model Evaluation

Markov Random Fields

Graphical Models: Powerful framework for representing dependency structure between random variables.

$$P_{\theta}(\mathbf{x}) = \frac{1}{\mathcal{Z}(\theta)} \exp(-E(\mathbf{x}; \theta)) = \frac{f_{\theta}(\mathbf{x})}{\mathcal{Z}(\theta)}$$

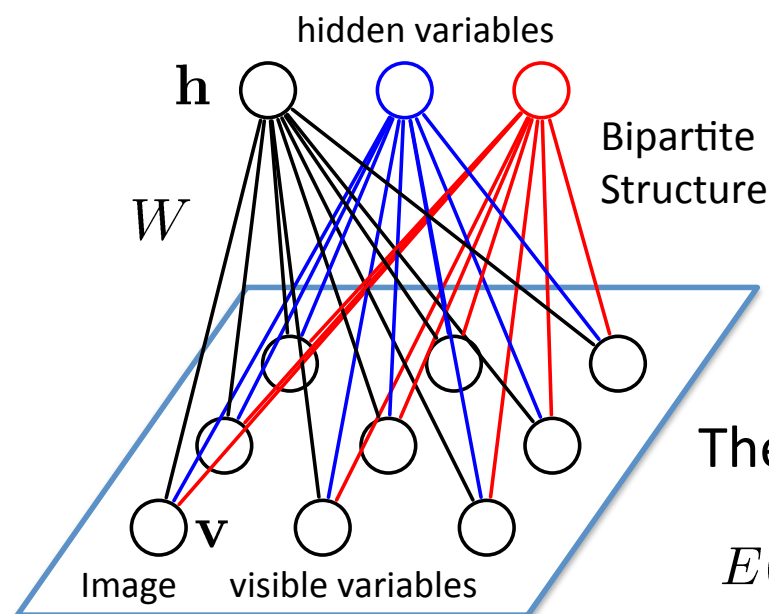


Partition function: difficult to compute

$$\mathcal{Z}(\theta) = \sum_{\mathbf{x}} \exp(-E(\mathbf{x}; \theta))$$

- **Goal:** Obtain good estimates of $\mathcal{Z}(\theta)$.

Restricted Boltzmann Machines



Stochastic binary visible variables $\mathbf{v} \in \{0, 1\}^D$ are connected to stochastic binary hidden variables $\mathbf{h} \in \{0, 1\}^F$.

The energy of the joint configuration:

$$E(\mathbf{v}, \mathbf{h}; \theta) = - \sum_{ij} W_{ij} v_i h_j - \sum_i b_i v_i - \sum_j a_j h_j$$

$\theta = \{W, a, b\}$ model parameters.

Probability of the joint configuration is given by the Boltzmann distribution:

$$P_{\theta}(\mathbf{v}) = \frac{1}{\mathcal{Z}(\theta)} \underbrace{\sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}; \theta))}_{\text{Tractable}} = \underbrace{\frac{f_{\theta}(\mathbf{v})}{\mathcal{Z}(\theta)}}_{\text{Intractable}}.$$

Markov random fields, Boltzmann machines, log-linear models.

Generative Model

- Which model is a better generative model?

Model A



Model B



Model Selection

- More generally, how can we choose between models?



RBM samples



Mixture of Bernoulli's

Compare $P(\mathbf{x})$ on the validation set: $P(\mathbf{x}) = f(\mathbf{x})/Z$.

Need an estimate of Partition Function Z

Model Selection

- More generally, how can we choose between models?



RBM samples



Mixture of Bernoulli's

MoB, test log-probability:


-137.64 nats/digit

RBM, test log-probability:

-86.35 nats/digit

Difference of about 50 nats!

Simple Importance Sampling

- Two distributions defined on \mathcal{X} with probability distribution functions $p_{\text{ini}}(\mathbf{x}) = f_{\text{ini}}(\mathbf{x})/\mathcal{Z}_0$ and $p_{\text{tgt}}(\mathbf{x}) = f_{\text{tgt}}(\mathbf{x})/\mathcal{Z}_{\text{tgt}}$


Proposal, easy to sample from distribution

Intractable, target distribution

- Under mild conditions:

$$\mathcal{Z}_{\text{tgt}} = \sum_{\mathbf{x}} f_{\text{tgt}}(\mathbf{x}) = \sum_{\mathbf{x}} \frac{f_{\text{tgt}}(\mathbf{x})}{p_{\text{ini}}(\mathbf{x})} \times p_{\text{ini}}(\mathbf{x})$$

- Get unbiased estimate of using Monte Carlo approximation:

$$\mathcal{Z}_{\text{tgt}} \approx \frac{1}{M} \sum_{m=1}^M \frac{f_{\text{tgt}}(\mathbf{x}^{(m)})}{p_{\text{ini}}(\mathbf{x}^{(m)})} = \frac{1}{M} \sum_{m=1}^M w^{(m)} \quad \mathbf{x}^{(m)} \sim p_{\text{ini}}$$

- In high-dimensional spaces, the variance will be high (or infinite).

Annealed Importance Sampling

- Consider a sequence of intermediate distributions:

p_0, p_1, \dots, p_K with $p_0 = p_{\text{ini}}$ and $p_K = p_{\text{tgt}}$.

- One general way is to use **geometric averages**:

$$p_\beta(\mathbf{x}) = f_\beta(\mathbf{x}) / \mathcal{Z}_\beta = f_{\text{ini}}(\mathbf{x})^{1-\beta} f_{\text{tgt}}(\mathbf{x})^\beta / \mathcal{Z}_\beta$$

with $0 = \beta_0 < \beta_1 < \dots < \beta_K = 1$ chosen by the user.

- If p_{ini} is the uniform distribution, then:

$$p_\beta(\mathbf{x}) = f_{\text{tgt}}(\mathbf{x})^\beta / \mathcal{Z}_\beta$$

inverse temperature

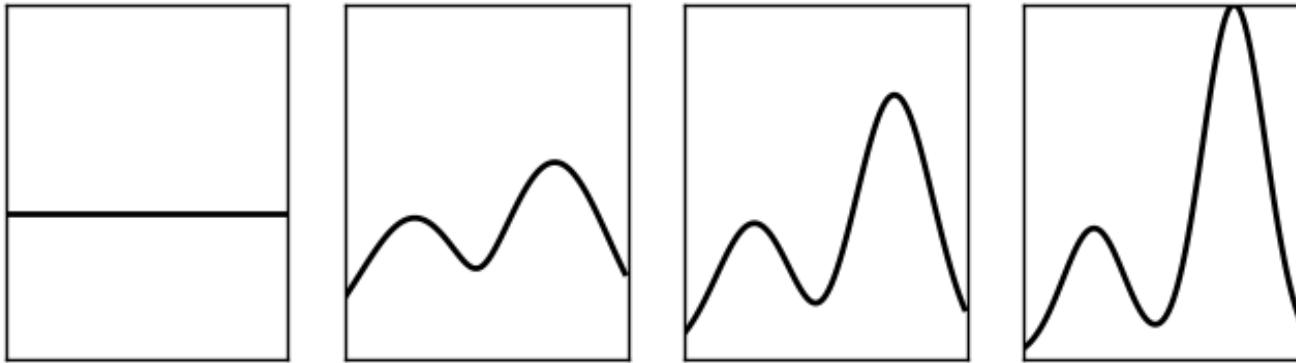
Annealing by Averaging Moments,
Grosse et al., NIPS, 2013

hence the term annealing.

(Neal, Statistics and Computing, 2001)

Annealed Importance Sampling

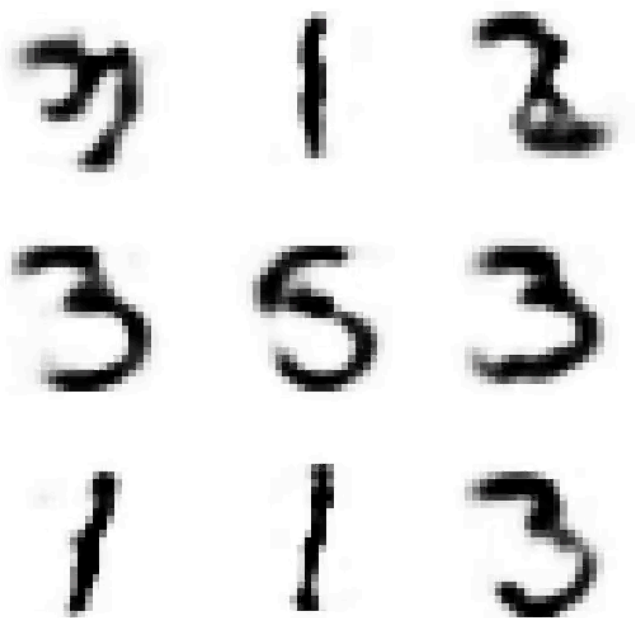
- Move gradually from hotter distribution to colder distribution:



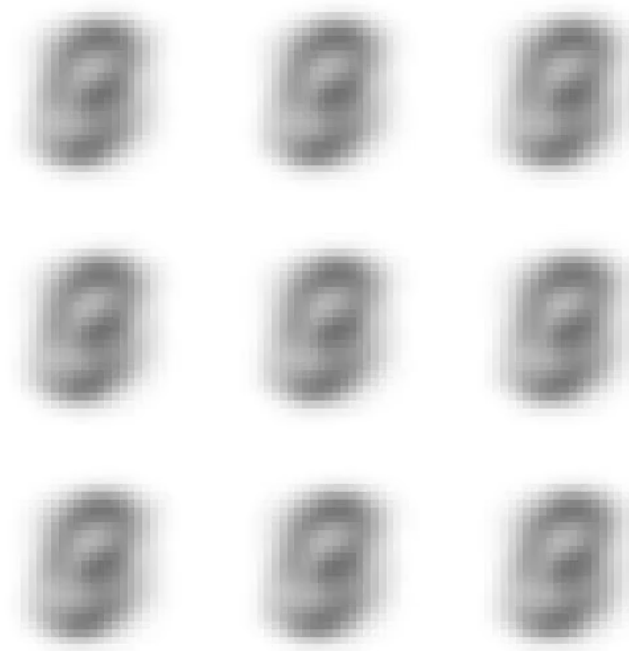
- Need to define transition operator $T_k(\mathbf{x}'|\mathbf{x})$ that leaves p_k invariant (e.g. Gibbs sampling) – Easy to implement!

RBM with Geometric Averages

- Restricted Boltzmann Machines trained on MNIST.



Samples from target
distribution



beta = 0.00

AIS with geometric
averages

Problems with Undirected Models

- AIS provides an unbiased estimator: $\mathbb{E}[\hat{\mathcal{Z}}_{\text{tgt}}] = \mathcal{Z}_{\text{tgt}}$. In general, we are interested in estimating $\log \mathcal{Z}_{\text{tgt}}$

- By Jensen's inequality:

$$\mathbb{E}[\log \hat{\mathcal{Z}}_{\text{tgt}}] \leq \log \mathbb{E}[\hat{\mathcal{Z}}_{\text{tgt}}] = \log \mathcal{Z}_{\text{tgt}}$$

- By Markov's inequality: very unlikely to overestimate $\log \mathcal{Z}_{\text{tgt}}$

$$\Pr(\log \hat{\mathcal{Z}}_{\text{tgt}} > \log \mathcal{Z}_{\text{tgt}} + b) \leq e^{-b}$$

Stochastic lower
bound!

- Compute log-probability on the test set:

$$\log p(\mathbf{x}) = \log f(\mathbf{x}) - \log \mathcal{Z}_{\text{tgt}}$$

↑
overestimate

↑
underestimate

Motivation: RBM Sampling

Run Markov chain (alternating Gibbs Sampling):

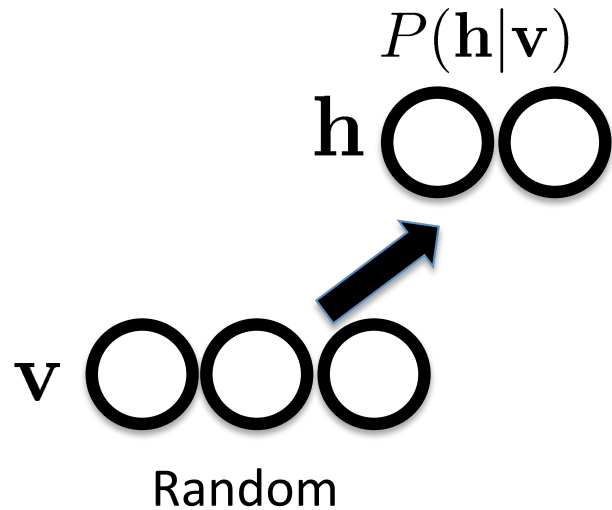
Motivation: RBM Sampling

Run Markov chain (alternating Gibbs Sampling):



Motivation: RBM Sampling

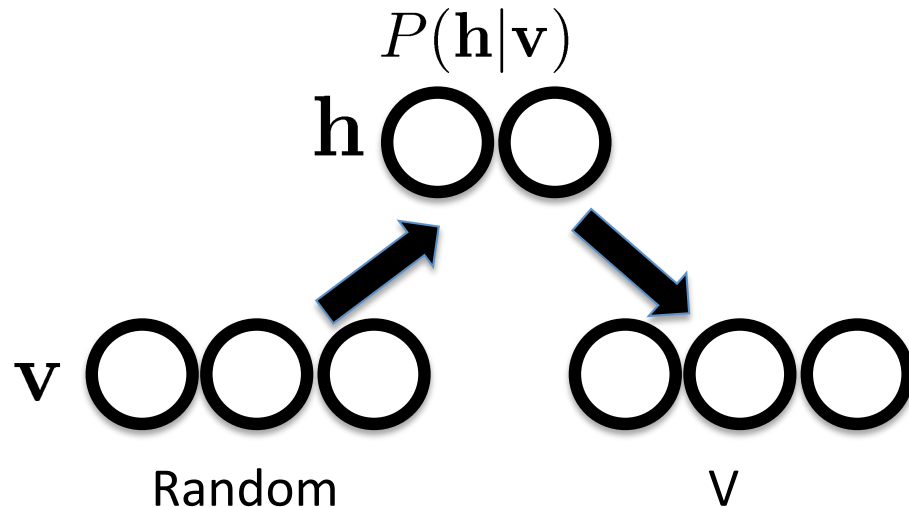
Run Markov chain (alternating Gibbs Sampling):



$$P(\mathbf{h}|\mathbf{v}) = \prod_j P(h_j|\mathbf{v}) \quad P(h_j = 1|\mathbf{v}) = \frac{1}{1 + \exp(-\sum_i W_{ij}v_i - a_j)}$$

Motivation: RBM Sampling

Run Markov chain (alternating Gibbs Sampling):

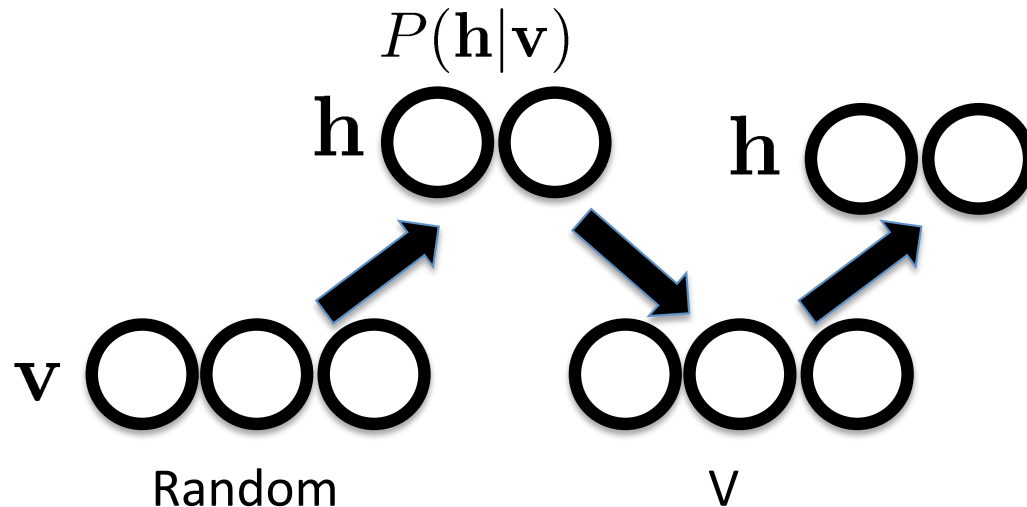


$$P(\mathbf{h}|\mathbf{v}) = \prod_j P(h_j|\mathbf{v}) \quad P(h_j = 1|\mathbf{v}) = \frac{1}{1 + \exp(-\sum_i W_{ij}v_i - a_j)}$$

$$P(\mathbf{v}|\mathbf{h}) = \prod_i P(v_i|\mathbf{h}) \quad P(v_i = 1|\mathbf{h}) = \frac{1}{1 + \exp(-\sum_j W_{ij}h_j - b_i)}$$

Motivation: RBM Sampling

Run Markov chain (alternating Gibbs Sampling):

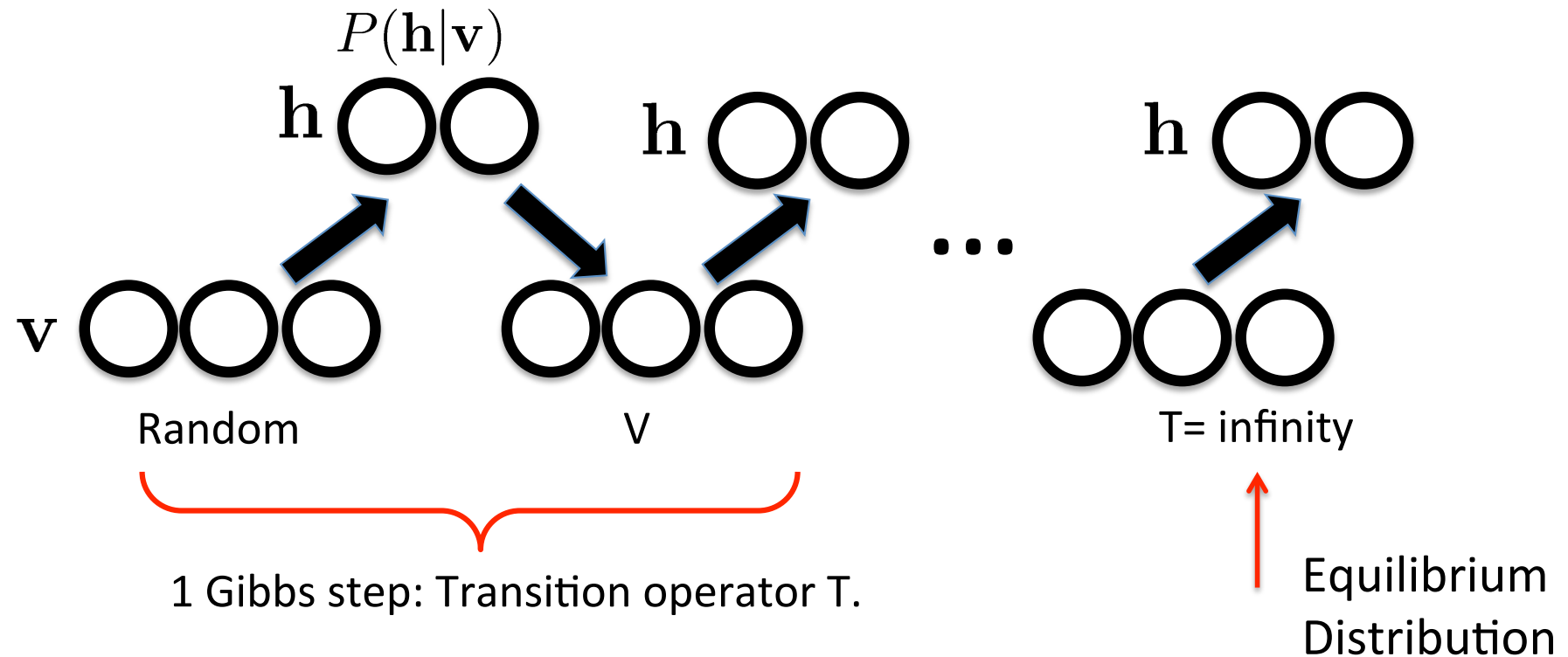


$$P(\mathbf{h}|\mathbf{v}) = \prod_j P(h_j|\mathbf{v}) \quad P(h_j = 1|\mathbf{v}) = \frac{1}{1 + \exp(-\sum_i W_{ij}v_i - a_j)}$$

$$P(\mathbf{v}|\mathbf{h}) = \prod_i P(v_i|\mathbf{h}) \quad P(v_i = 1|\mathbf{h}) = \frac{1}{1 + \exp(-\sum_j W_{ij}h_j - b_i)}$$

Motivation: Sampling

Run Markov chain (alternating Gibbs Sampling):

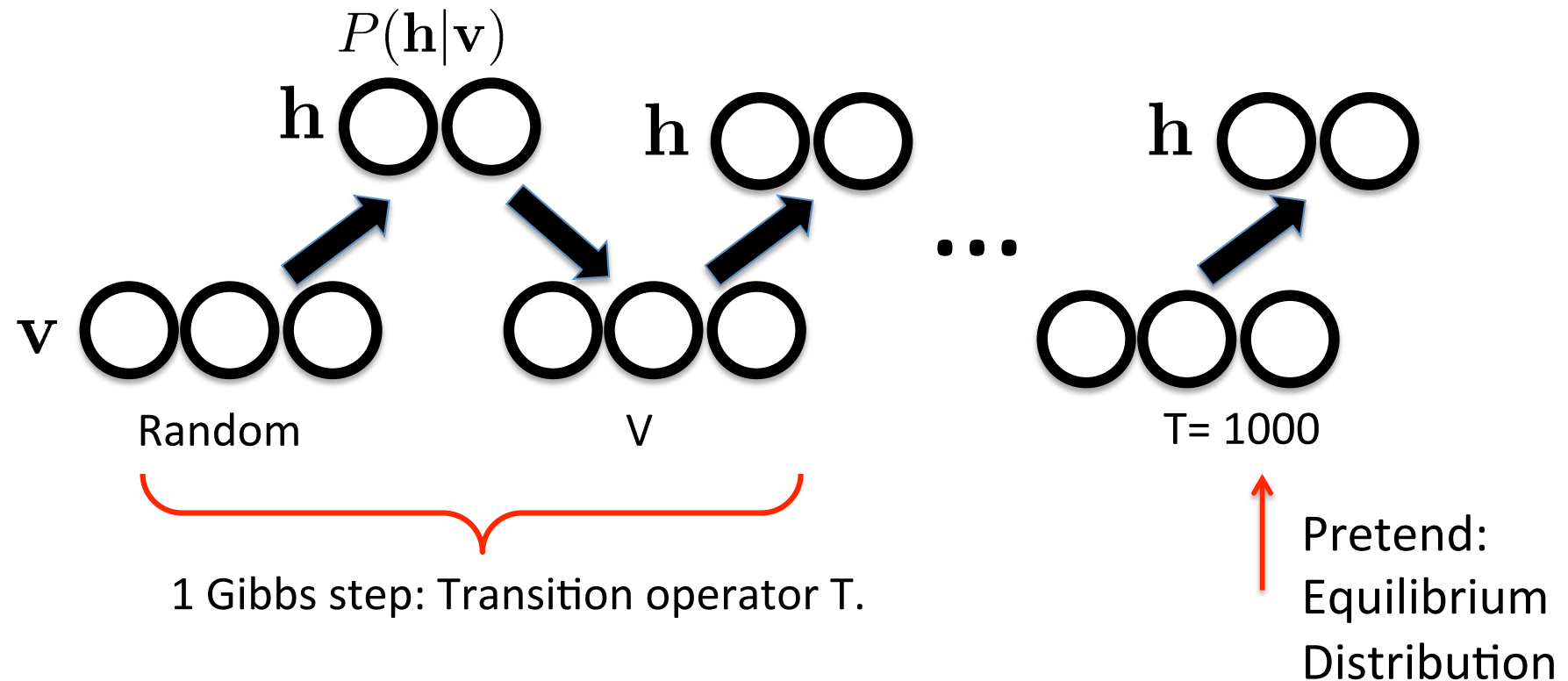


$$P(\mathbf{h}|\mathbf{v}) = \prod_j P(h_j|\mathbf{v}) \quad P(h_j = 1|\mathbf{v}) = \frac{1}{1 + \exp(-\sum_i W_{ij}v_i - a_j)}$$

$$P(\mathbf{v}|\mathbf{h}) = \prod_i P(v_i|\mathbf{h}) \quad P(v_i = 1|\mathbf{h}) = \frac{1}{1 + \exp(-\sum_j W_{ij}h_j - b_i)}$$

Motivation: Sampling

Run Markov chain (alternating Gibbs Sampling):

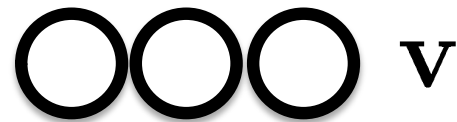


$$P(\mathbf{h}|\mathbf{v}) = \prod_j P(h_j|\mathbf{v}) \quad P(h_j = 1|\mathbf{v}) = \frac{1}{1 + \exp(-\sum_i W_{ij}v_i - a_j)}$$

$$P(\mathbf{v}|\mathbf{h}) = \prod_i P(v_i|\mathbf{h}) \quad P(v_i = 1|\mathbf{h}) = \frac{1}{1 + \exp(-\sum_j W_{ij}h_j - b_i)}$$

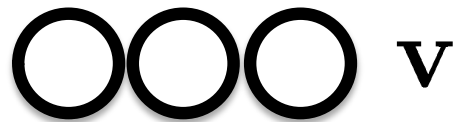
Unrolled RBM as a Deep Generative Model

Random (uniform)



Unrolled RBM as a Deep Generative Model

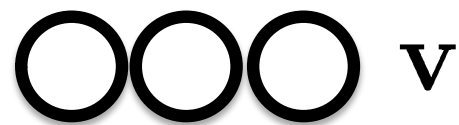
Random (uniform)



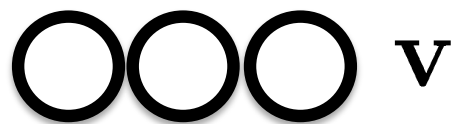
...

Unrolled RBM as a Deep Generative Model

Random (uniform)



...

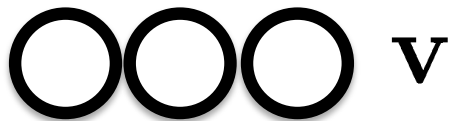


Unrolled RBM as a Deep Generative Model

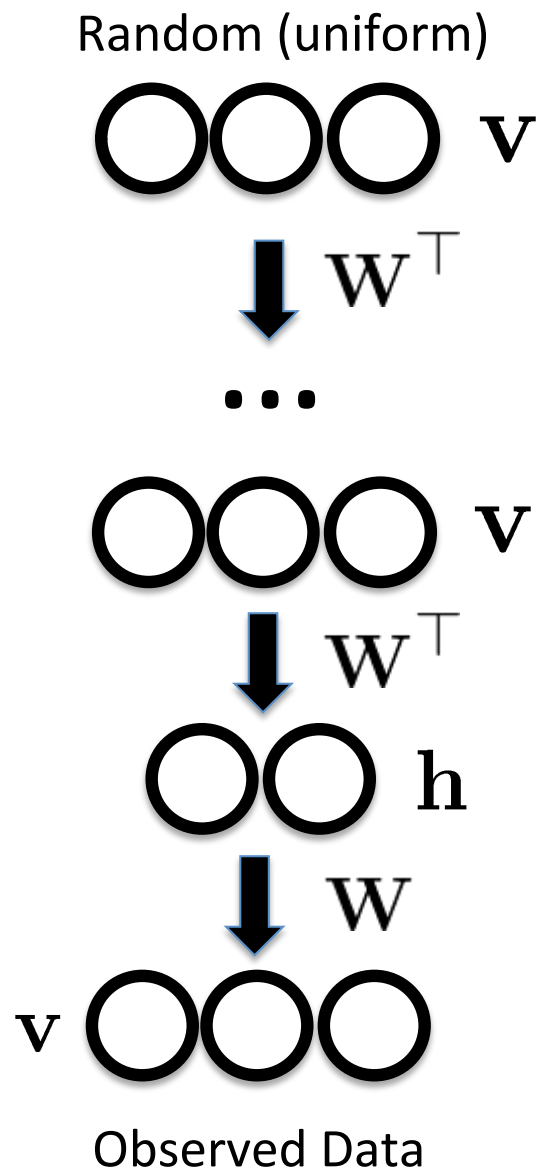
Random (uniform)



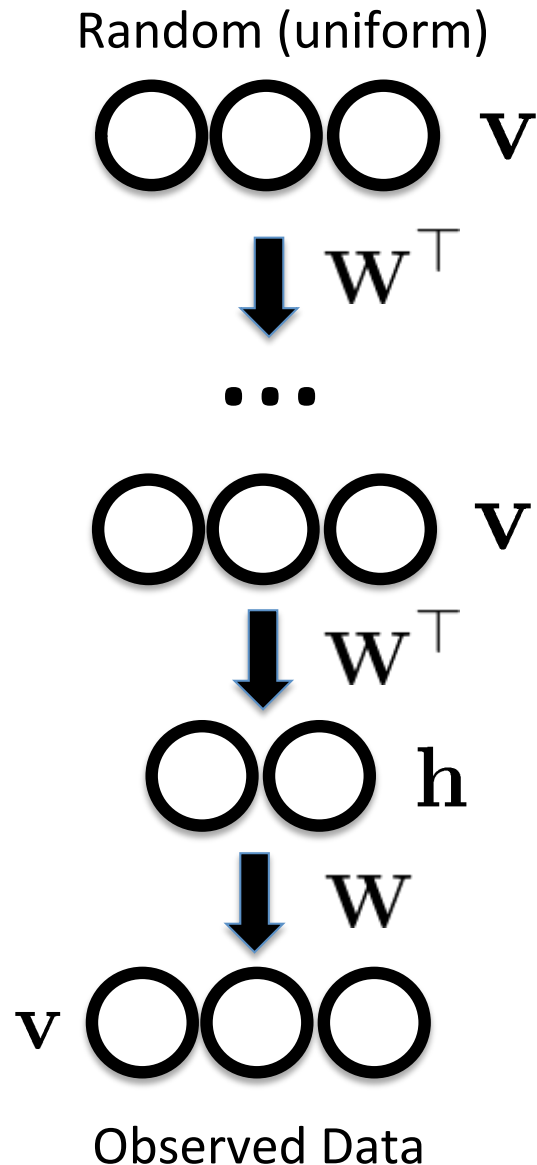
...



Unrolled RBM as a Deep Generative Model



Unrolled RBM as a Deep Generative Model



- If we use infinite number of layers, then:

$$P_{gen}(\mathbf{v}) = P_{RBM}(\mathbf{v})$$

- Otherwise, deep generative model is just an approximation to an RBM.

Reverse AIS Estimator (RAISE)



$$T_k(\mathbf{x}_k | \mathbf{x}_{k-1})$$

...



- Let us consider $\mathbf{x} = \{\mathbf{v}, \mathbf{h}\}$ where \mathbf{v} is observed and \mathbf{h} is unobserved.

- Define the following generative process (*sequence of AIS distributions*):

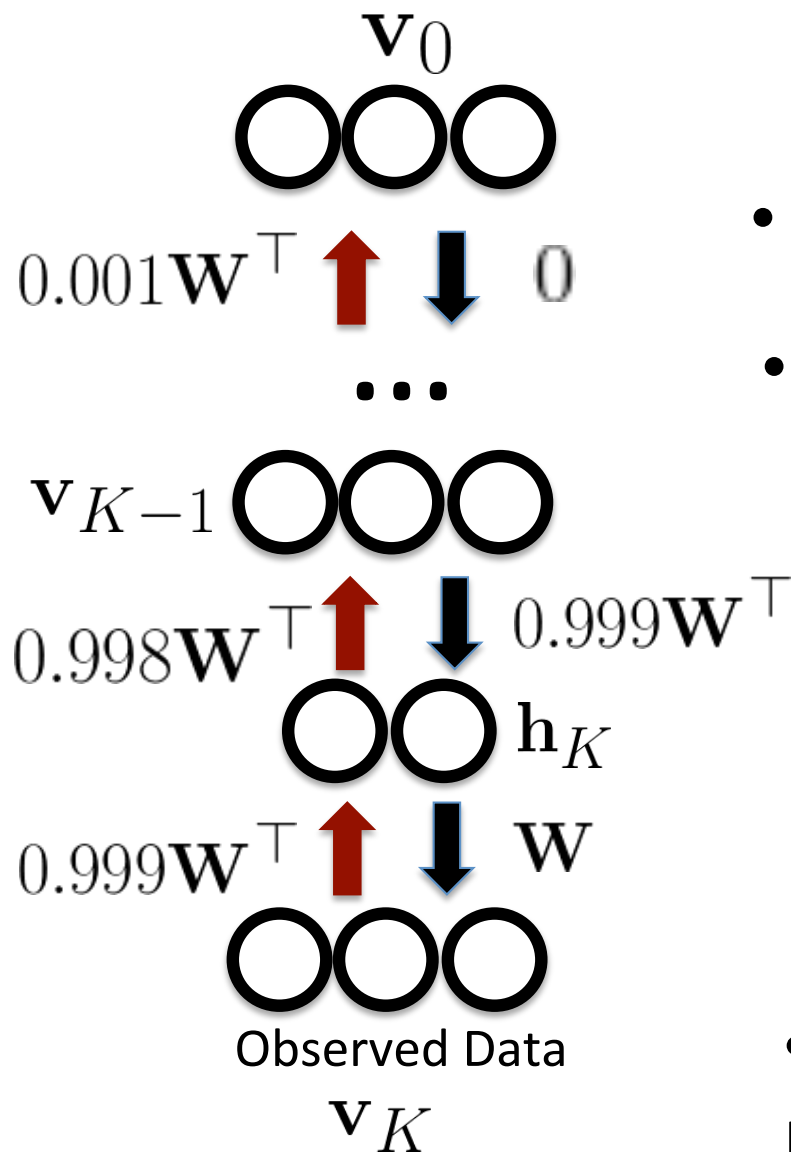
$$p_{\text{fwd}}(\mathbf{x}_{0:K}) = p_0(\mathbf{x}_0) \prod_{k=1}^K T_k(\mathbf{x}_k | \mathbf{x}_{k-1})$$

- Generative model, that we call the **annealing model**:

$$p_{\text{ann}}(\mathbf{v}_K) = \sum_{\mathbf{x}_{0:K-1}, \mathbf{h}_K} p_{\text{fwd}}(\mathbf{x}_{0:K-1}, \mathbf{h}_K, \mathbf{v}_K)$$

(Burda, Grosse, Salakhutdinov, AISTATS 2015)

Reverse AIS Estimator (RAISE)



- As K goes to infinity:

$$P_{\text{ann}}(\mathbf{x}) = P_{\text{RBM}}(\mathbf{x})$$

- We would like to estimate $p(\mathbf{v}_{\text{test}})$.

- We use reverse chain as our proposal:

$$q_{\text{rev}}(\mathbf{x}_{0:K-1}, \mathbf{h}_K | \mathbf{v}_{\text{test}}) = p_{\text{tgt}}(\mathbf{h}_K | \mathbf{v}_{\text{test}}) \prod_{k=1}^K \tilde{T}_k(\mathbf{x}_{k-1} | \mathbf{x}_k)$$



Assume tractable, which is the case for RBMs

- Can be easily extended to non-tractable posteriors, e.g. DBMs, DBNs.

Reverse AIS Estimator (RAISE)

- We now have our generative model (theoretical construct):

$$p_{\text{fwd}}(\mathbf{x}_{0:K}) = p_0(\mathbf{x}_0) \prod_{k=1}^K T_k(\mathbf{x}_k | \mathbf{x}_{k-1})$$

- Proposal starts at the data and melts the distribution:

$$q_{\text{rev}}(\mathbf{x}_{0:K-1}, \mathbf{h}_K | \mathbf{v}_{\text{test}}) = p_{\text{tgt}}(\mathbf{h}_K | \mathbf{v}_{\text{test}}) \prod_{k=1}^K \tilde{T}_k(\mathbf{x}_{k-1} | \mathbf{x}_k)$$

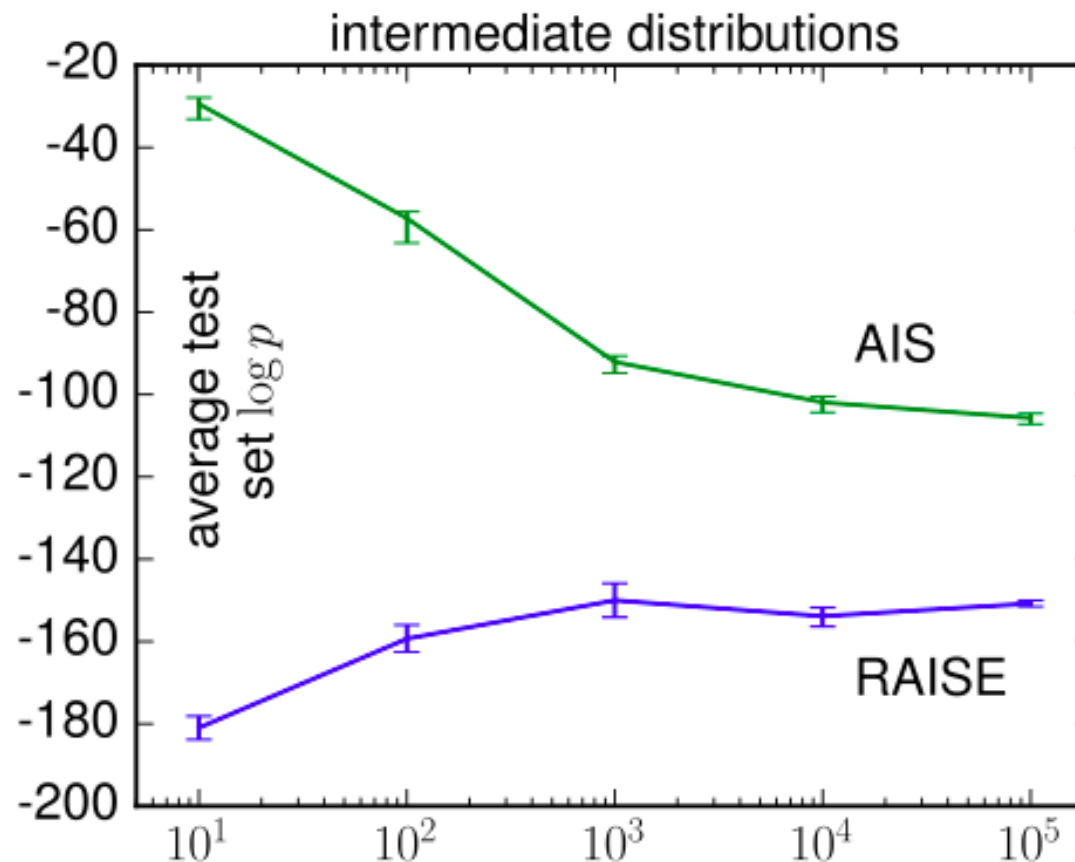
- We then obtain:

$$\begin{aligned} P_{\text{ann}}(\mathbf{v}_{\text{test}}) &= \mathbb{E}_{q_{\text{rev}}} \left[\frac{f_{\text{fwd}}}{q_{\text{rev}}} \right] \\ &= \mathbb{E}_{q_{\text{rev}}} \left[\frac{f_{\text{tgt}}(\mathbf{v}_{\text{test}})}{\mathcal{Z}_0} \prod_{k=1}^{K-1} \frac{f_k(\mathbf{x}_k)}{f_{k+1}(\mathbf{x}_k)} \right] = \mathbb{E}_{q_{\text{rev}}} [w] \end{aligned}$$

- Tends to underestimate rather than overestimate log-probs!

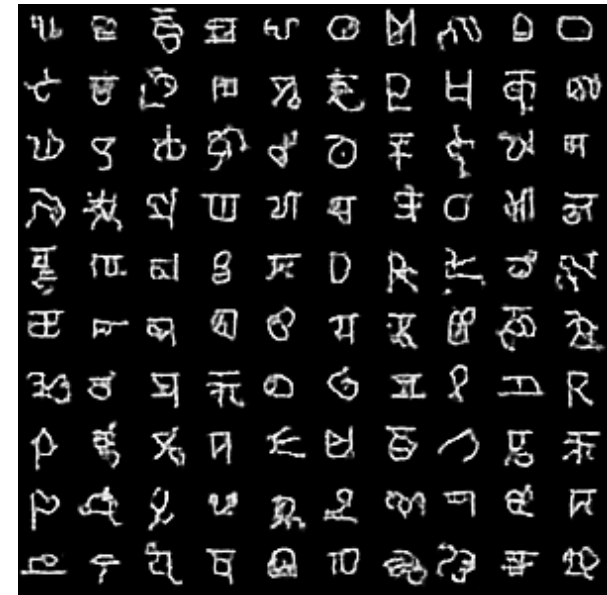
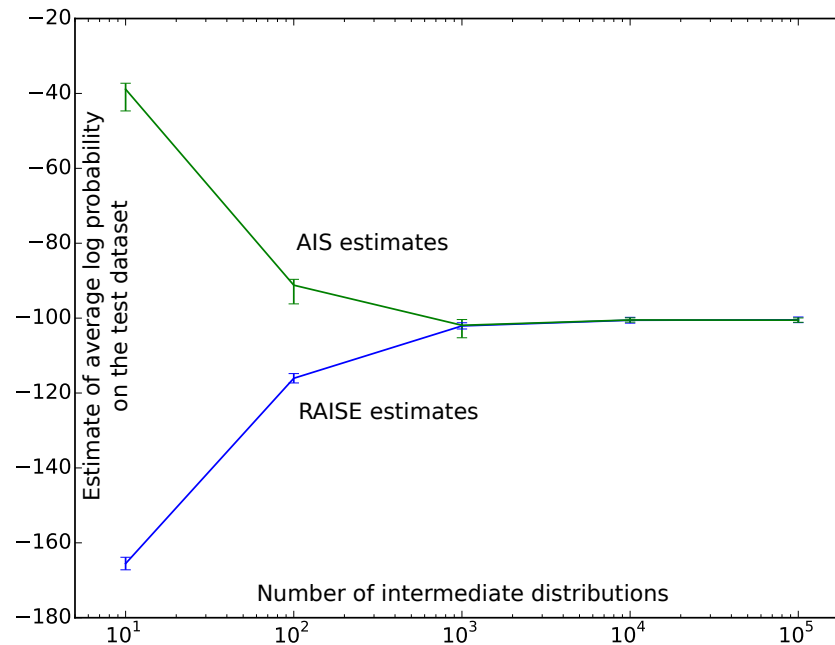
MNIST

- RBM with 500 hidden units trained on MNIST.
- Initial distribution is uniform: AIS is off by 20 nats!



Omniglot Dataset

- RBM with 500 hidden units trained on Omniglot.



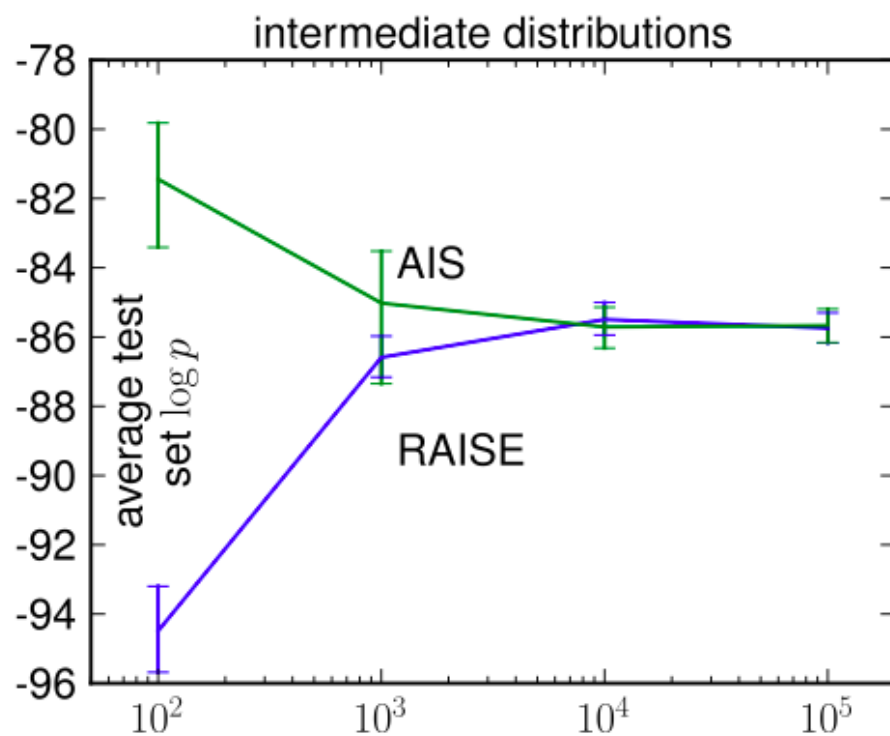
MNIST and Omniglot Results

Model	exact	CSL	RAISE	uniform	gap
				AIS	
mnistCD1-20	-164.50	-185.74	-165.33	-164.51	0.82
mnistPCD-20	-150.11	-152.13	-150.58	-150.04	0.54
mnistCD1-500	—	-566.91	-150.78	-106.52	44.26
mnistPCD-500	—	-138.76	-101.07	-99.99	1.08
mnistCD25-500	—	-145.26	-88.51	-86.42	2.09
omniPCD-1000	—	-144.25	-100.47	-100.45	0.02

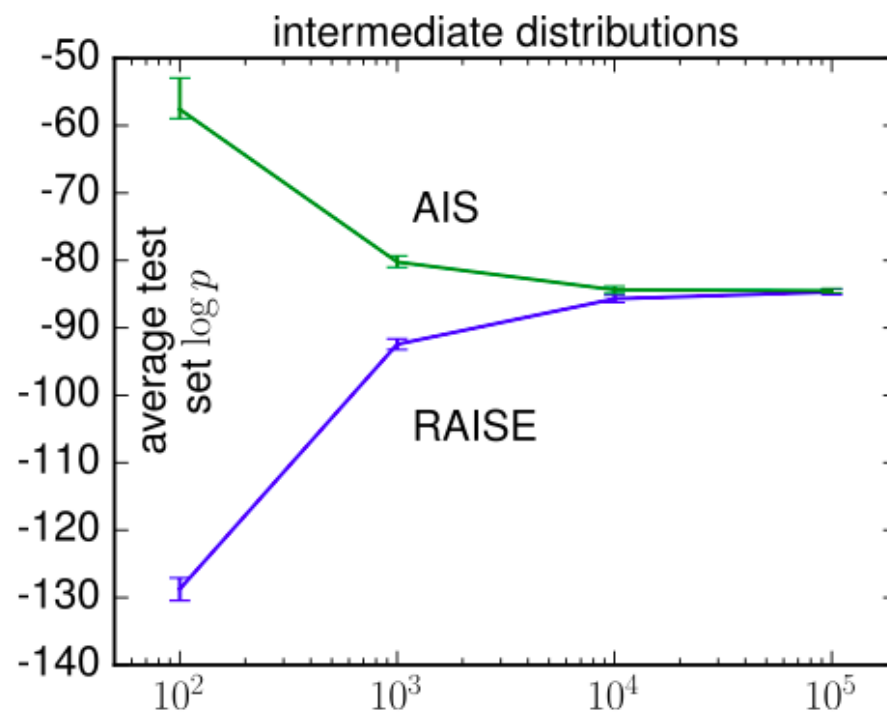
- RAISE errs on the side of underestimating the log-likelihood.
- Note that the gap is very small.
- CSL: Conservative Sampling-based Log-likelihood (CSL) estimator of Bengio et. al.

DBMs and DBNs

Deep Boltzmann Machine



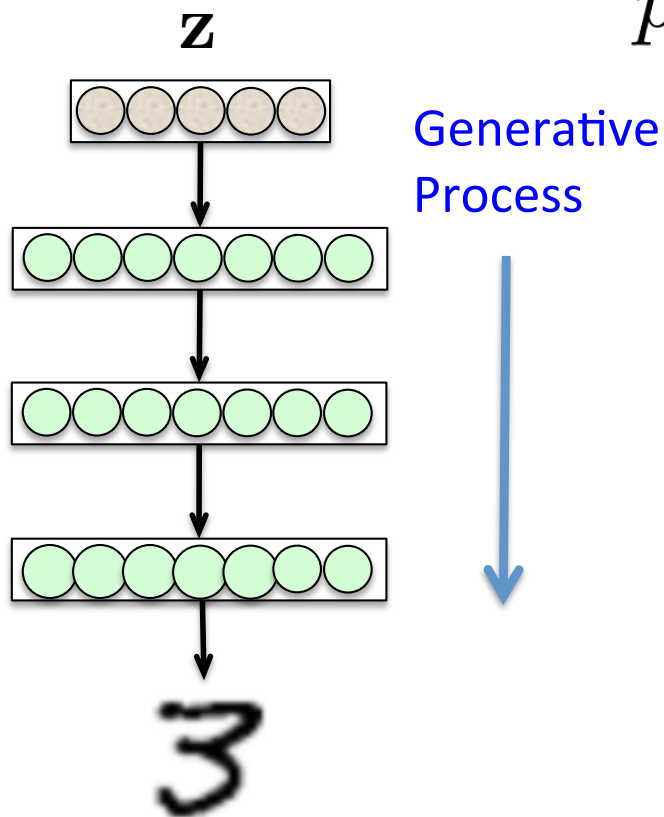
Deep Believe Network



Decoder-Based Models

- **Decoder-Based Models:** Transform samples from some simple distribution (e.g. normal) to the data manifold:

$$p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z})$$



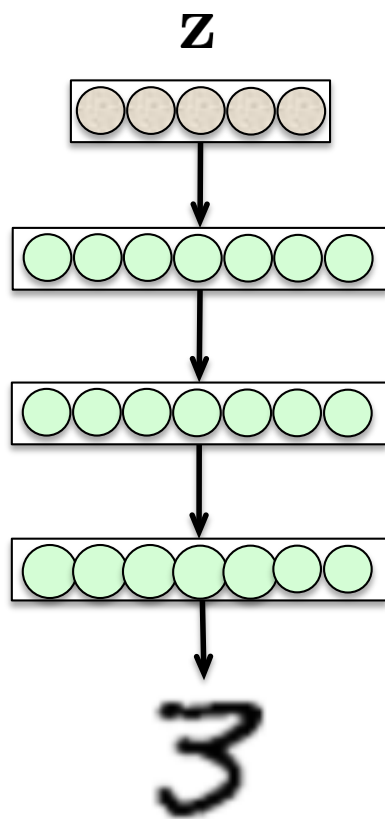
← Deterministic neural network

- Variational Autoencoders (VAEs) (Knigam and Welling, 2014)
- Generative Adversarial Networks (GANs) (Goodfellow et.al., 2014)
- Generative Moment Matching Networks (GMMNs) (Li & Swersky, 2015; Dziugaite et al., 2015)

Decoder-Based Models

- **Decoder-Based Models:** Transform samples from some simple distribution (e.g. normal) to the data manifold:

$$p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z})$$



Generative
Process

Deterministic neural network

- Variational Autoencoders (VAEs) (Knigam and Welling, 2014)
- Generative Adversarial Networks (GANs) (Goodfellow et.al., 2014)

AIS can be used to properly
evaluate decoder-based models
(Wu, Burda, Salakhutdinov, Grosse, 2016)

networks
ziugaite et

Talk Roadmap

Part 1: Supervised Learning: Deep Networks

- Definition of Neural Networks
- Training Neural Networks
- Recent Optimization / Regularization Techniques

Part 2: Unsupervised Learning: Learning Deep Generative Models

Part 3: Open Research Problems

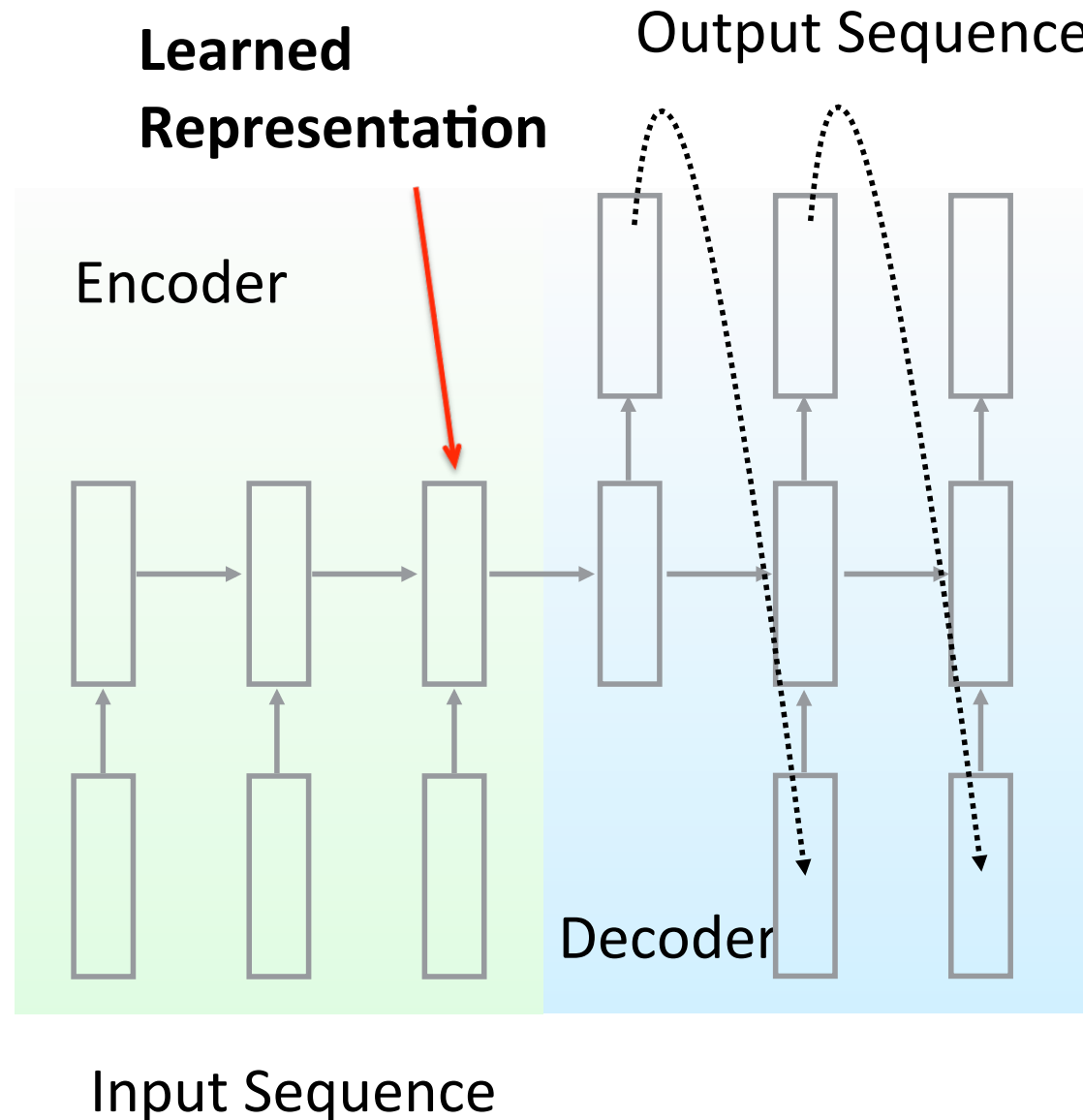
(Some) Open Problems

- Unsupervised Learning / Transfer Learning / One-Shot Learning
- Reasoning, Attention, and Memory
- Natural Language Understanding
- Deep Reinforcement Learning

(Some) Open Problems

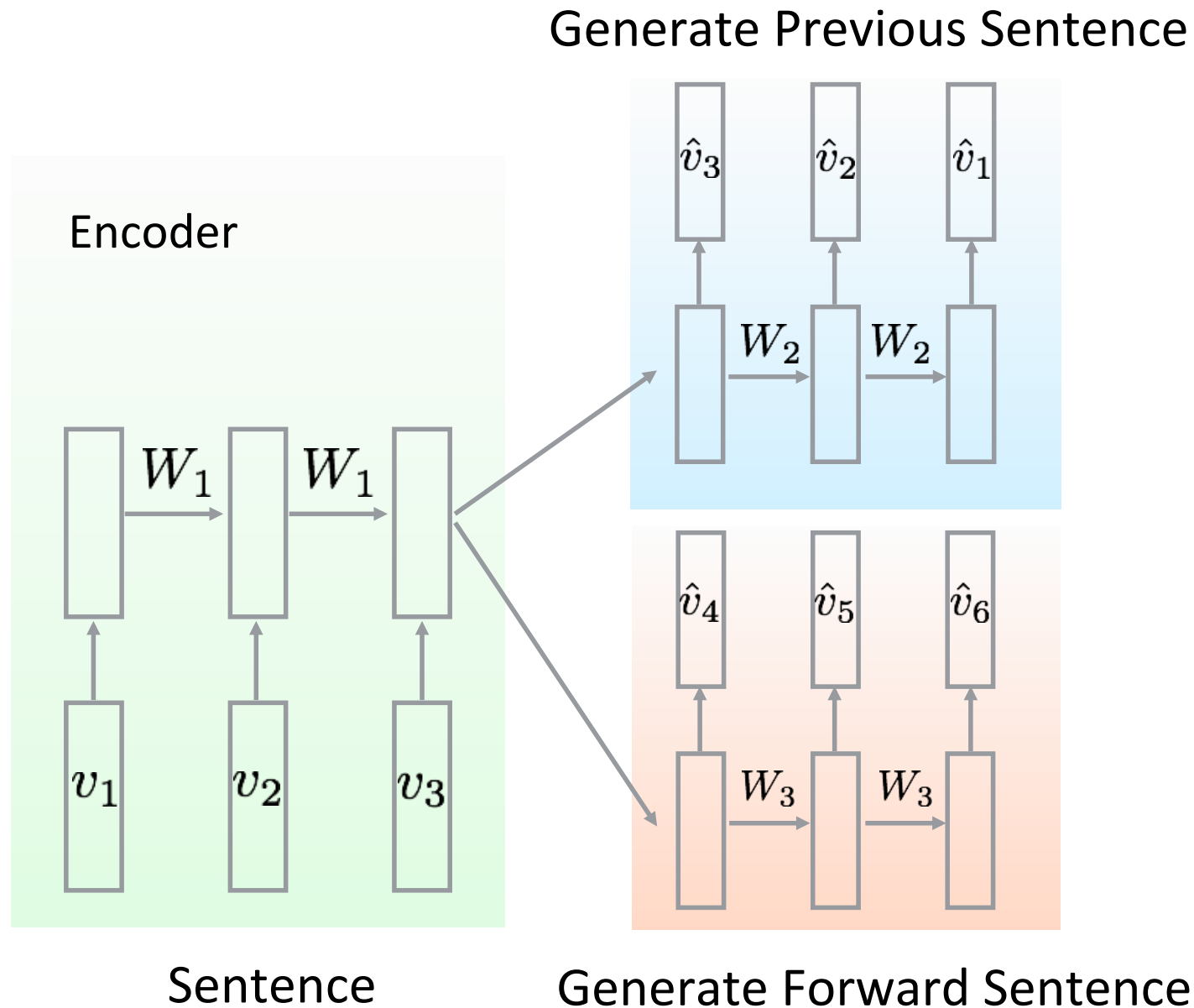
- Unsupervised Learning / Transfer Learning / One-Shot Learning
- Reasoning, Attention, and Memory
- Natural Language Understanding
- Deep Reinforcement Learning

Sequence to Sequence Learning



- RNN Encoder-Decoders for Machine Translation (Sutskever et al. 2014; Cho et al. 2014; Kalchbrenner et al. 2013, Srivastava et.al., 2015)

Skip-Thought Model



(Kiros et al., NIPS 2015)

Learning Objective

- **Objective:** The sum of the log-probabilities for the next and previous sentences conditioned on the encoder representation:

$$\sum_t \log P(w_{i+1}^t | w_{i+1}^{<t}, \mathbf{h}_i) + \sum_t \log P(w_{i-1}^t | w_{i-1}^{<t}, \mathbf{h}_i)$$

representation of encoder
↓

Forward sentence Previous sentence

- **Data:** Book-11K corpus:

# of books	# of sentences	# of words	# of unique words
11,038	74,004,228	984,846,357	1,316,420

Semantic Relatedness

		Method	r	ρ	MSE
SemEval 2014 sub- missions	{	Illinois-LH [18]	0.7993	0.7538	0.3692
		UNAL-NLP [19]	0.8070	0.7489	0.3550
		Meaning Factory [20]	0.8268	0.7721	0.3224
		ECNU [21]	0.8414	—	—
Results reported by Tai et.al.	{	Mean vectors [22]	0.7577	0.6738	0.4557
		DT-RNN [23]	0.7923	0.7319	0.3822
		SDT-RNN [23]	0.7900	0.7304	0.3848
		LSTM [22]	0.8528	0.7911	0.2831
		Bidirectional LSTM [22]	0.8567	0.7966	0.2736
		Dependency Tree-LSTM [22]	0.8676	0.8083	0.2532
Ours	{	uni-skip	0.8477	0.7780	0.2872
		bi-skip	0.8405	0.7696	0.2995
		combine-skip	0.8584	0.7916	0.2687
		combine-skip+COCO	0.8655	0.7995	0.2561

- Our models outperform all previous systems from the SemEval 2014 competition.

Semantic Relatedness Recurrent Neural Network

- How similar the two sentences are on the scale 1 to 5?

Ground Truth 5.0

Prediction 4.9

A man is driving a car.

A car is being driven by a man.

Ground Truth 2.9

Prediction 3.5

A girl is looking at a
woman in costume.

A girl in costume looks like
a woman.

Ground Truth 2.6

Prediction 4.4

A person is performing
tricks on a motorcycle

The performer is tricking a
person on a motorcycle

Paraphrase Detection

- Microsoft Research Paraphrase Corpus: For two sentences one must predict whether or not they are paraphrases.

	Method	Acc	F1	
Recursive Auto-encoders	feats [24]	73.2		The training set contains 4076 sentence pairs (2753 are positive)
	RAE+DP [24]	72.6		
	RAE+feats [24]	74.2		
	RAE+DP+feats [24]	76.8	83.6	
Best published results	FHS [25]	75.0	82.7	The test set contains 1725 pairs (1147 are positive).
	PE [26]	76.1	82.7	
	WDDP [27]	75.6	83.0	
	MTMETRICS [28]	77.4	84.1	
Ours	uni-skip	73.0	81.9	
	bi-skip	71.2	81.2	
	combine-skip	73.0	82.0	
	combine-skip + feats	75.8	83.0	

Neural Story Telling



Sample from the Generative Model (recurrent neural network):

She was in love with him for the first time in months, so she had no intention of escaping.

The sun had risen from the ocean, making her feel more alive than normal . She is beautiful, but the truth is that I do not know what to do. The sun was just starting to fade away, leaving people scattered around the Atlantic Ocean.

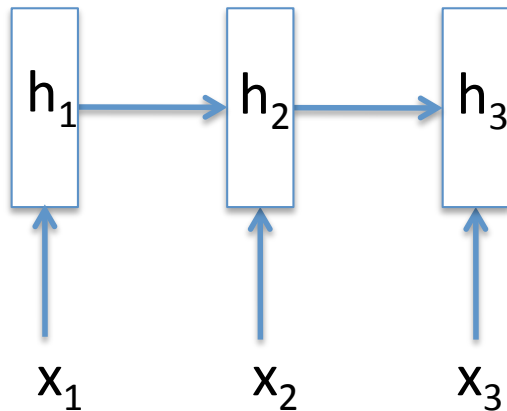
Recurrent Neural Network

$$\mathbf{h}_t = \phi(\mathbf{U}\mathbf{h}_{t-1} + \mathbf{W}\mathbf{x}_t + \mathbf{b})$$

Nonlinearity

Hidden State at
previous time step

Input at time
step t



Multiplicative Integration

- Replace

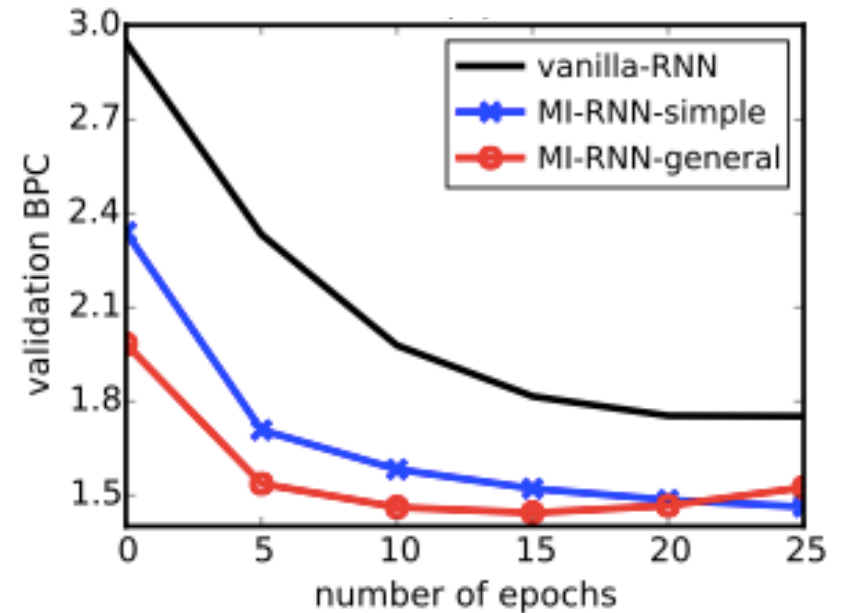
$$\phi(\mathbf{U}\mathbf{h} + \mathbf{W}\mathbf{x} + \mathbf{b})$$

- With

$$\phi(\mathbf{U}\mathbf{h} \odot \mathbf{W}\mathbf{x} + \mathbf{b})$$

- Or more generally

$$\phi(\alpha \odot \mathbf{U}\mathbf{h} \odot \mathbf{W}\mathbf{x} + \beta_1 \odot \mathbf{U}\mathbf{h} + \beta_2 \odot \mathbf{W}\mathbf{x} + \mathbf{b})$$

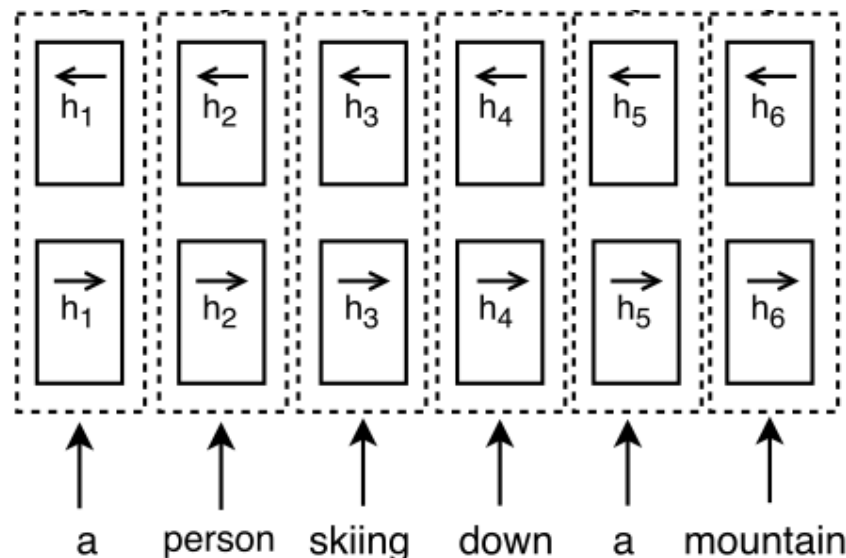


“Who Did What” Dataset

- **Document:** Japanese prime minister Taro Aso said on Friday he would call for stronger monitoring of international finance at the G20 summit next week..... US treasury secretary Timothy Geithner has said president Barack Obama would discuss new global financial regulatory standards at the London summit.
- **Query:** US president Barack will push higher financial regulatory standards for across the globe at the upcoming G20 summit in London XXX said on Thursday
- **Answer:** Timothy Geithner

Onishi, Wang, Bansal, Gimpel, McAllester.
Who did what: A large-scale person-centered
cloze dataset. EMNLP, 2016.

Representing Document/Query



- **Forward RNN** reads sentences from left to right:

$$[\vec{h}_1, \vec{h}_2, \dots, \vec{h}_{|D|}]$$

- **Backward RNN** reads sentences from right to left:

$$[\overleftarrow{h}_1, \overleftarrow{h}_2, \dots, \overleftarrow{h}_{|D|}]$$

- The hidden states are then concatenated:

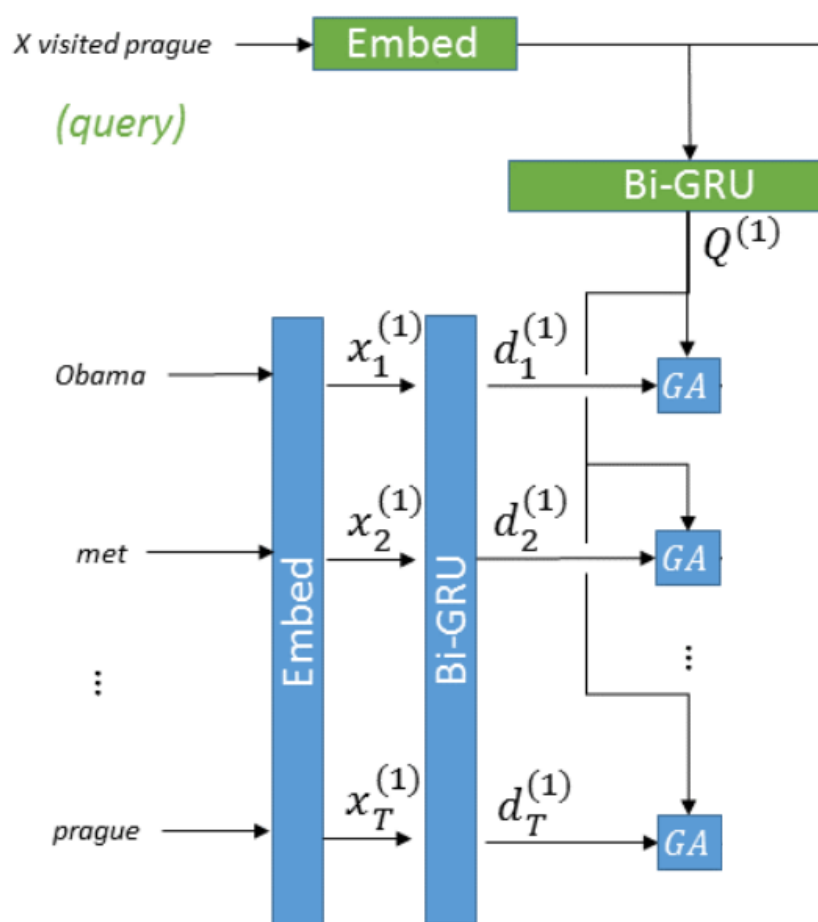
$$\overleftrightarrow{\text{GRU}} = [h_1, h_2, \dots, h_{|D|}], \quad h_i = [\vec{h}_i, \overleftarrow{h}_i]$$

- Use GRUs to encode a document and a query:

$$D = \overleftrightarrow{\text{GRU}}_D(X) \quad Q = \overleftrightarrow{\text{GRU}}_Q(Y)$$

Gated Attention (GA) Mechanism

- For each word in document D, we form a **token-specific representation of the query Q**:



$$\alpha_i = \text{softmax}(Q^\top d_i)$$

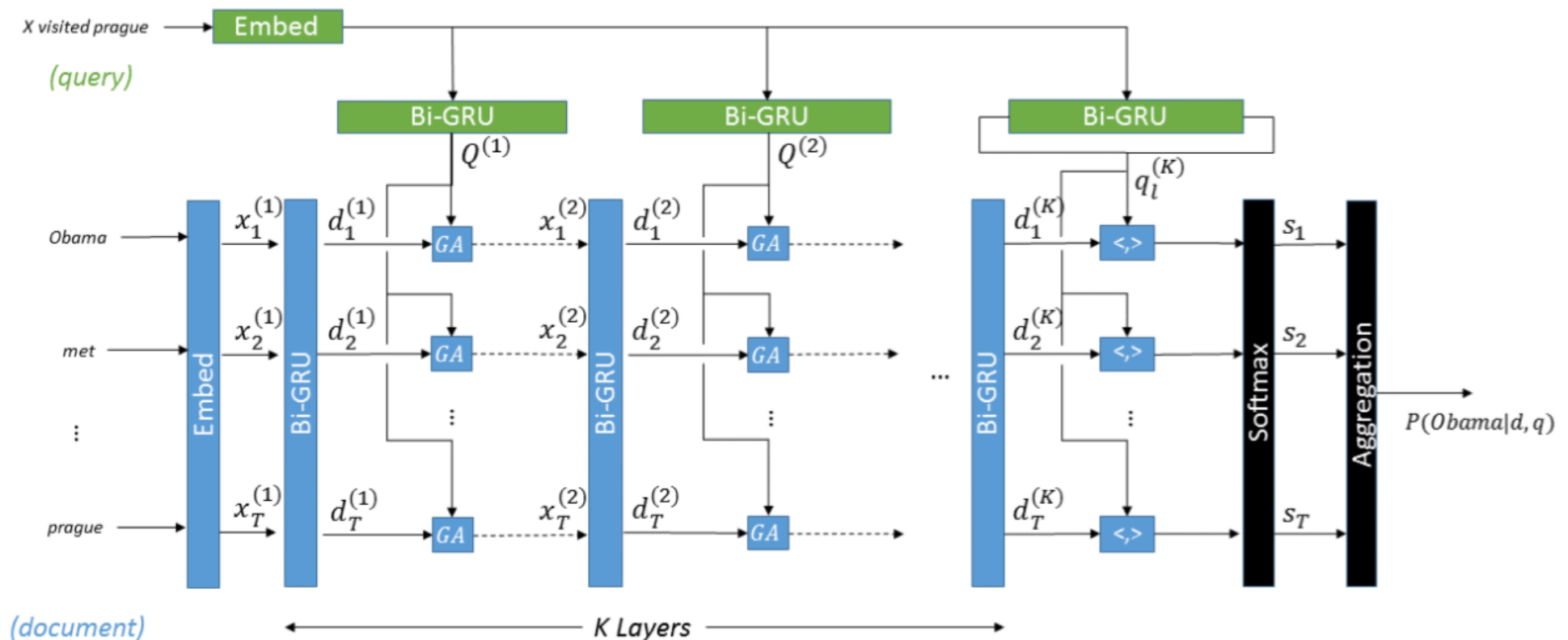
$$\tilde{q}_i = Q \alpha_i$$

$$x_i = d_i \odot \tilde{q}_i$$

- use the element-wise multiplication operator to model the interactions between d_i and \tilde{q}_i

Multi-hop Architecture

- Many QA tasks require reasoning over multiple sentences.
- Need to performs several passes over the context.



Affect of Multiplicative Gating

- Performance of different gating functions on “Who did What” (WDW) dataset.

Gating Function	Accuracy	
	Val	Test
Sum	62.9	62.1
Concatenate	63.1	61.1
Multiply	67.8	67.0

Model	Strict		Relaxed	
	Val	Test	Val	Test
Human †	–	84.0	–	–
Attentive Reader †	–	53.0	–	55.0
AS Reader †	–	57.0	–	59.0
Stanford AR †	–	64.0	–	65.0
NSE †	66.5	66.2	67.0	66.7
GA Reader-- †	–	57.0	–	60.0
GA Reader	67.8	67.0	66.4	66.3
GA Reader (+feature)	70.1	69.5	70.9	70.6

Model	CNN		Daily Mail		CBT-NE		CBT-CN	
	Val	Test	Val	Test	Val	Test	Val	Test
Humans (query) †	–	–	–	–	–	52.0	–	64.4
Humans (context + query) †	–	–	–	–	–	81.6	–	81.6
LSTMs (context + query) †	–	–	–	–	51.2	41.8	62.6	56.0
Deep LSTM Reader †	55.0	57.0	63.3	62.2	–	–	–	–
Attentive Reader †	61.6	63.0	70.5	69.0	–	–	–	–
Impatient Reader †	61.8	63.8	69.0	68.0	–	–	–	–
MemNets †	63.4	66.8	–	–	70.4	66.6	64.2	63.0
AS Reader †	68.6	69.5	75.0	73.9	73.8	68.6	68.8	63.4
DER Network †	71.3	72.9	–	–	–	–	–	–
Stanford AR (relabeling) †	73.8	73.6	77.6	76.6	–	–	–	–
Iterative Attentive Reader †	72.6	73.3	–	–	75.2	68.6	72.1	69.2
EpiReader †	73.4	74.0	–	–	75.3	69.7	71.5	67.4
AoA Reader †	73.1	74.4	–	–	77.8	72.0	72.2	69.4
ReasoNet †	72.9	74.7	77.6	76.6	–	–	–	–
NSE †	–	–	–	–	78.2	73.2	74.3	71.9
MemNets (ensemble) †	66.2	69.4	–	–	–	–	–	–
AS Reader (ensemble) †	73.9	75.4	78.7	77.7	76.2	71.0	71.1	68.9
Stanford AR (relabeling,ensemble) †	77.2	77.6	80.2	79.2	–	–	–	–
Iterative Attentive Reader (ensemble) †	75.2	76.1	–	–	76.9	72.0	74.1	71.0
EpiReader (ensemble) †	–	–	–	–	76.6	71.8	73.6	70.6
AS Reader (+BookTest) † ‡	–	–	–	–	80.5	76.2	83.2	80.8
AS Reader (+BookTest,ensemble) † ‡	–	–	–	–	82.3	78.4	85.7	83.7
GA Reader--	73.0	73.8	76.7	75.7	74.9	69.0	69.0	63.9
GA Reader	77.9	77.9	81.5	80.9	74.9	70.8	71.8	69.0
GA Reader (+feature)	77.3	76.9	80.7	80.0	76.8	72.5	73.1	69.6

(Some) Open Problems

- Unsupervised Learning / Transfer Learning / One-Shot Learning
- Reasoning and Natural Language Understanding
- Deep Reinforcement Learning

One-Shot Learning

“zarc”

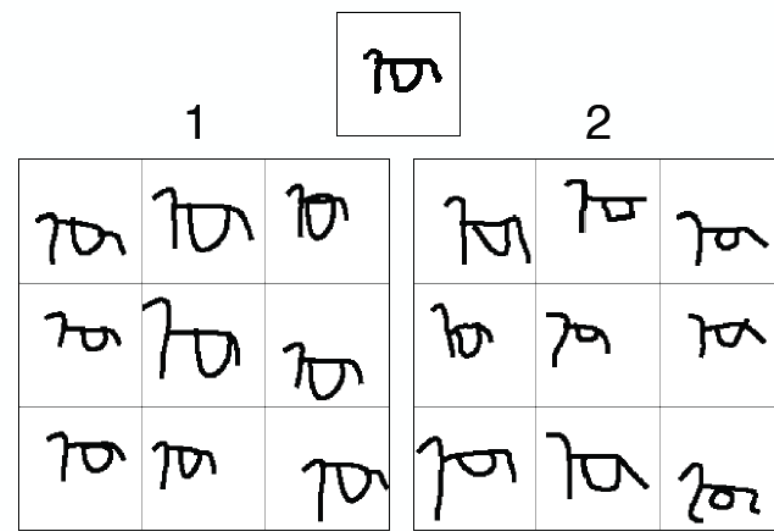
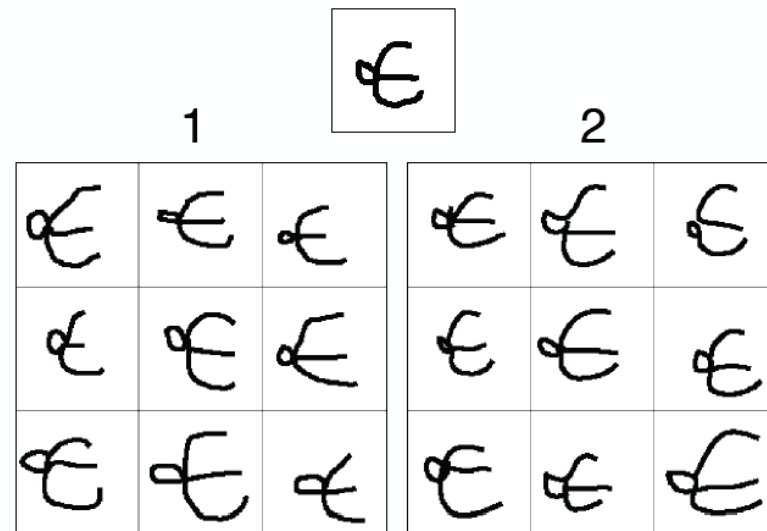
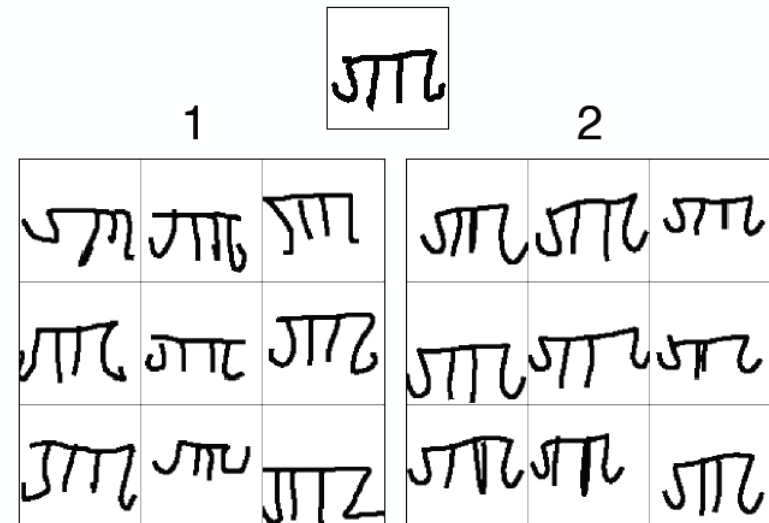
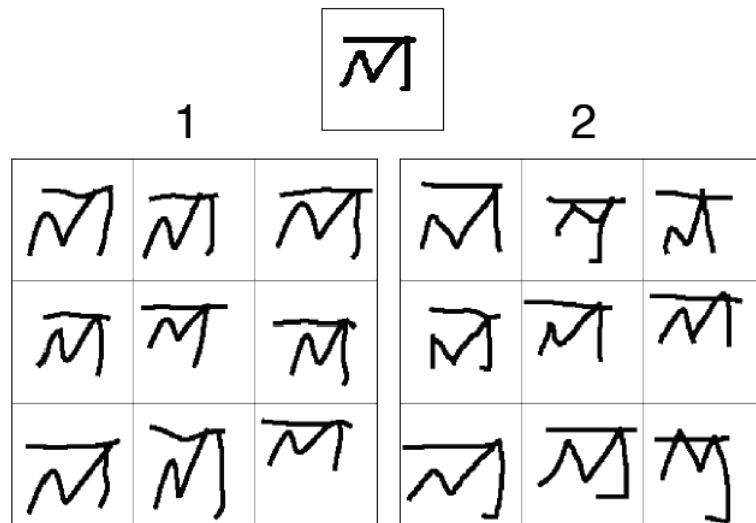
৮	৮	৮	৮
৮	৮	৮	৮
৮	৮	৮	৮
৮	৮	৮	৮

One-Shot Learning



How can we learn a novel concept – a high dimensional statistical object – from few examples.

One-Shot Learning: Humans vs. Machines



Reinforcement Learning

- Can a single network play many games at once?
- Can we learn new games faster by using knowledge about the previous games?

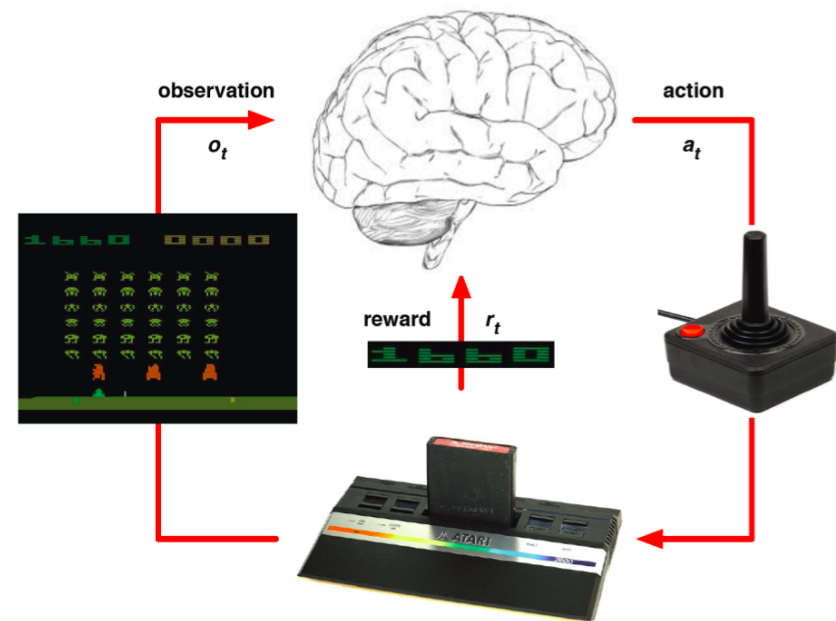


Figure credit: Nando de Freitas

Actor-Mimic Net in Action

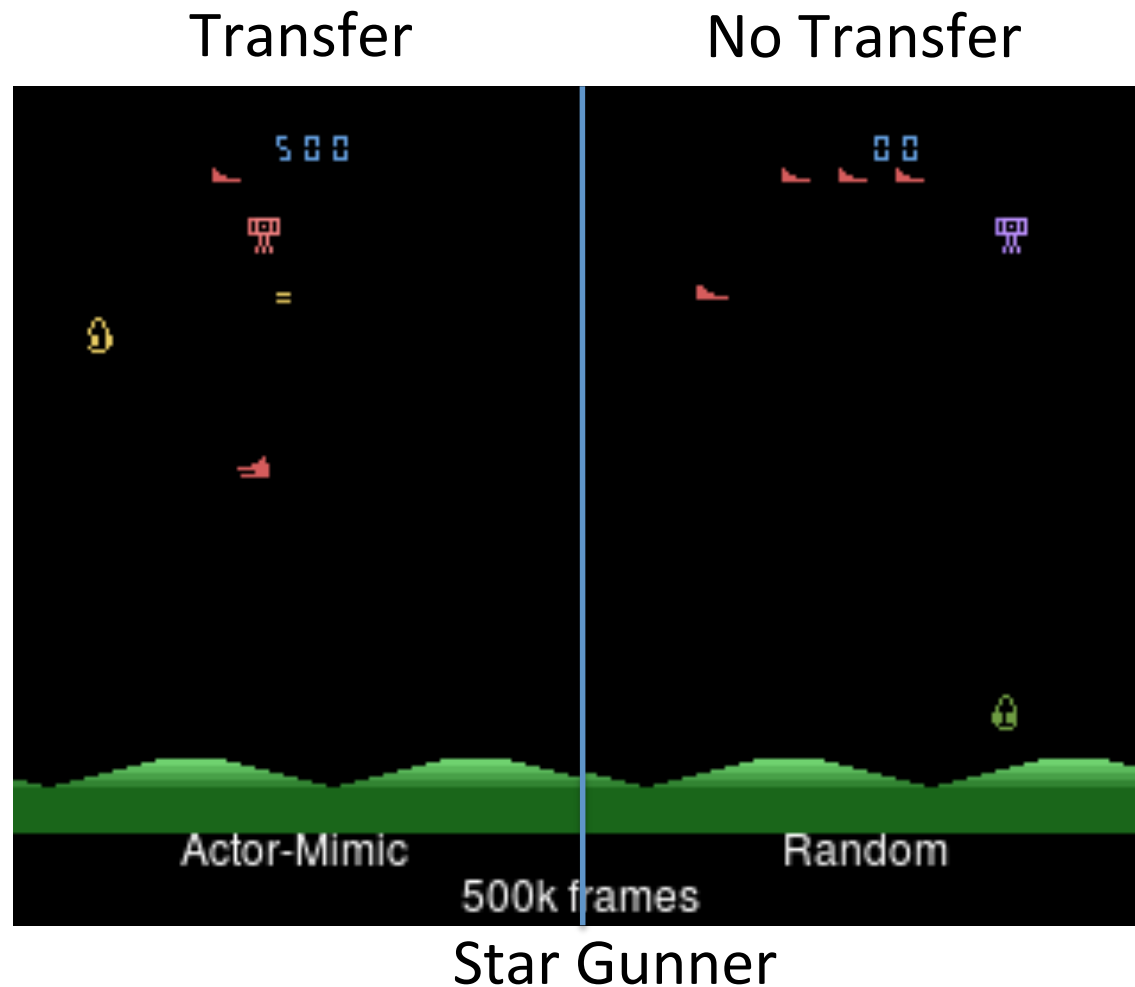
- The multitask network can match expert performance on 8 games (we are extending this to more games).



(Parisotto, Ba, Salakhutdinov, ICLR 2016)

Transfer Learning

- Can the network learn new games faster by leveraging knowledge about the previous games it learned.



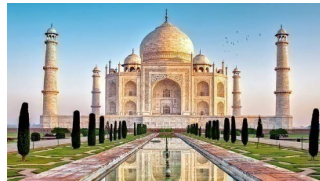
Summary

- Efficient learning algorithms for Deep Unsupervised Models

Text & image retrieval /
Object recognition

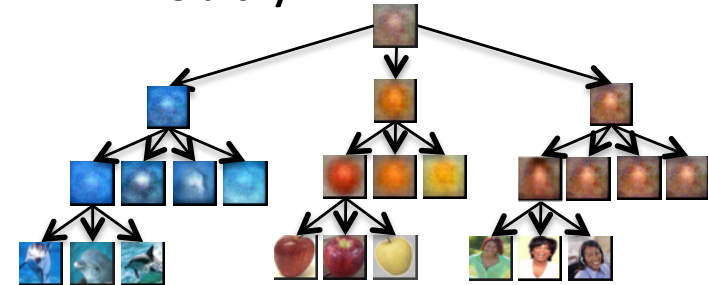


Image Tagging



mosque, tower,
building, cathedral,
dome, castle

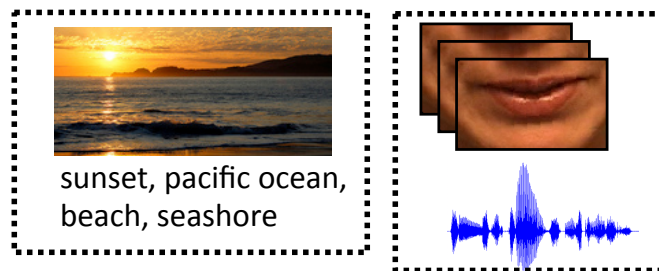
Learning a Category
Hierarchy



Object Detection



Multimodal Data



- Deep models improve the current state-of-the art in many application domains:
 - Object recognition and detection, text and image retrieval, handwritten character and speech recognition, and others.

Thank you