Machine Learning for Clinicians:

Advances for Multi-Modal Health Data

Michael C. Hughes

A Tutorial at MLHC 2018, August 16, 2018

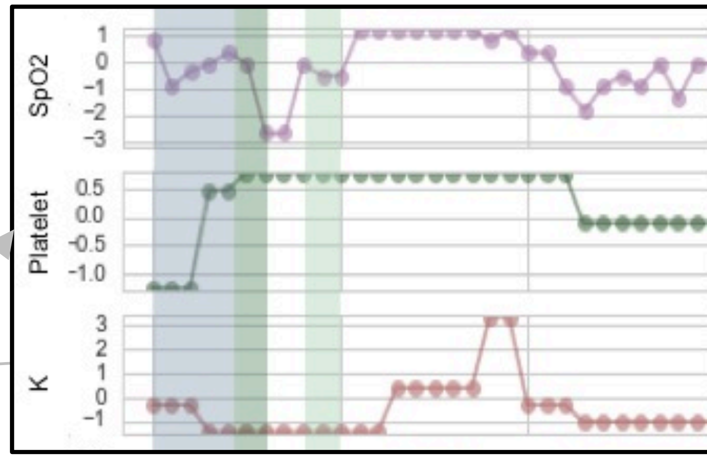
PART 2: Learning Representations for Sequences, Images, EHR, and Text

- Learned representations: topic models, CNNs, RNNs
- Tricks of the trade: Data augmentation, dropout
- Models that generate data: Deep Generative Models, VAE, GAN

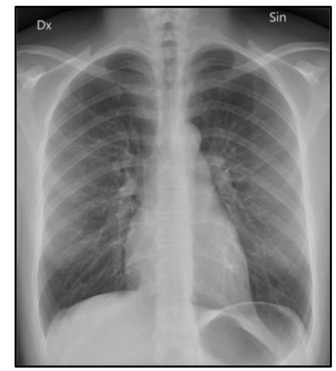
Slides / Resources / Bibliography:
https://michaelchughes.com/mlhc2018_tutorial.html

Part 2: Learning Representations for Sequences, Images, EHR, and Text

Vital Signs over time



Imaging



Text from Clinical Notes

"21yo woman who presented with pain on the left side of her thorax"

How to represent this structured data for prediction/classification?

Part 2 outline

2-stage hand-engineered representations of data

- Bag of words for images, text, EHR codes

Learnable representations of data

- Images
- Time series
- Text

- Tricks of the trade
- Models that generate data

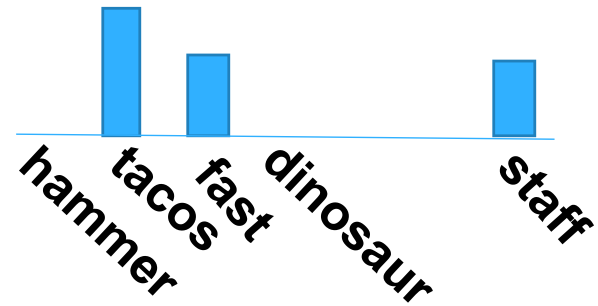
Bag-of-words representation

Popular 2000-2012

original data | unordered "bag" of vocab symbols | **count vector** over large (fixed-size) vocabulary

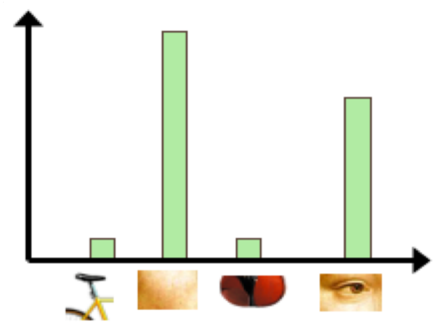
Text

Friendly staff, good tacos, fresh ingredients, and fast service. What more can you look for at taco bell?



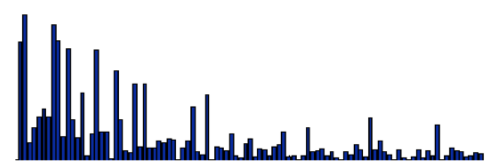
Images

Credit: Fei-Fei Li



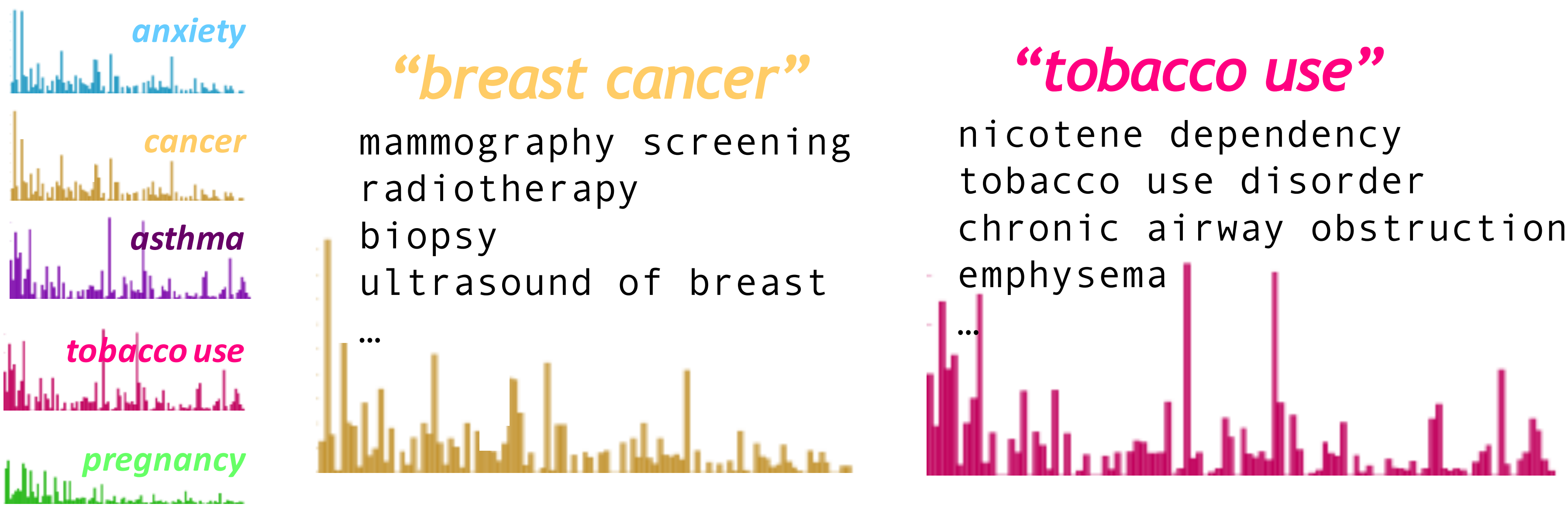
ICD-9/ CPT Codes

09/01 emphysema
09/01 biopsy
09/01 emphysema
…
09/15 radiotherapy



Topic models for clinical bag-of-codes

Explain all data via set of shared clinical "topics"



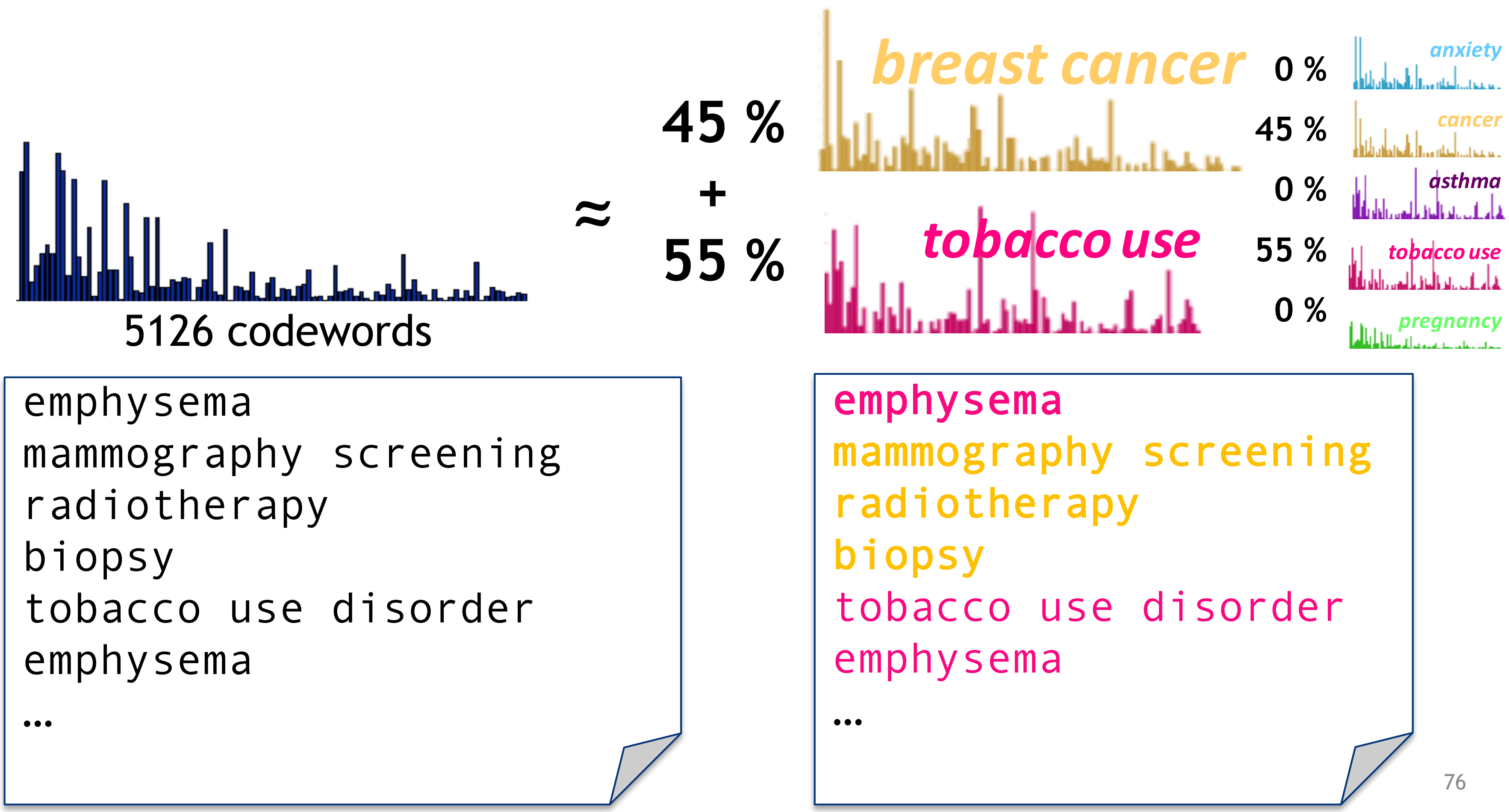
Each "topic" is a distribution over all 5126 possible billing codewords

How to interpret?

- Might be a **subtype** of target disease (bipolar, postpartum, etc.)
- Might be **related conditions**

Flexible Patient Representation

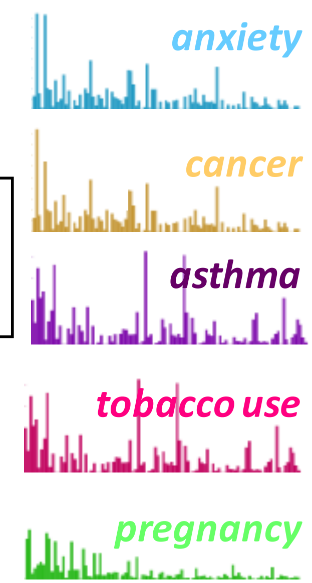
Each patient's history is a "mixture" of topics

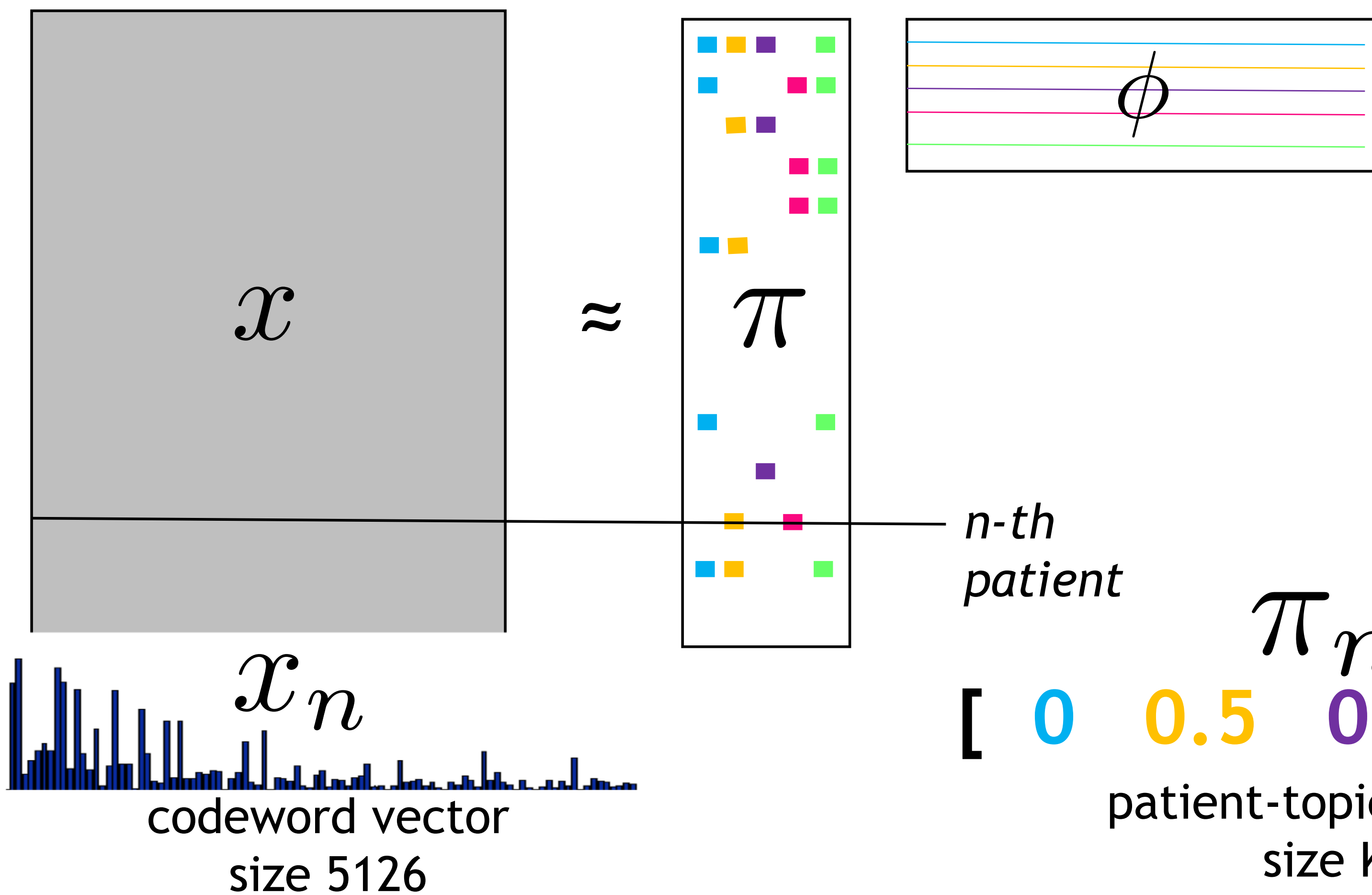


Topic Model

e.g. Latent Dirichlet Allocation
"LDA" (Blei, Ng & Jordan '03)

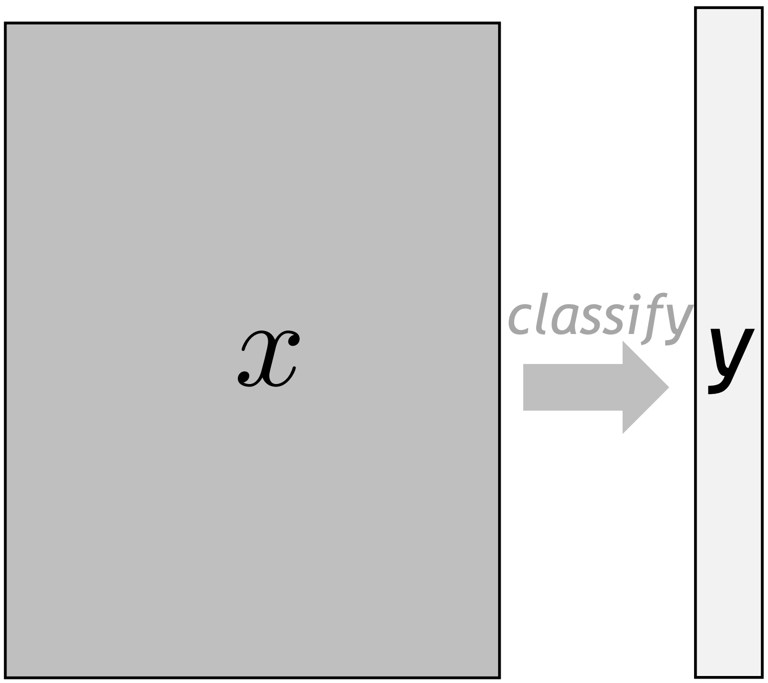
1) Pick number of topics *K*
2) Train to **reconstruct data x**



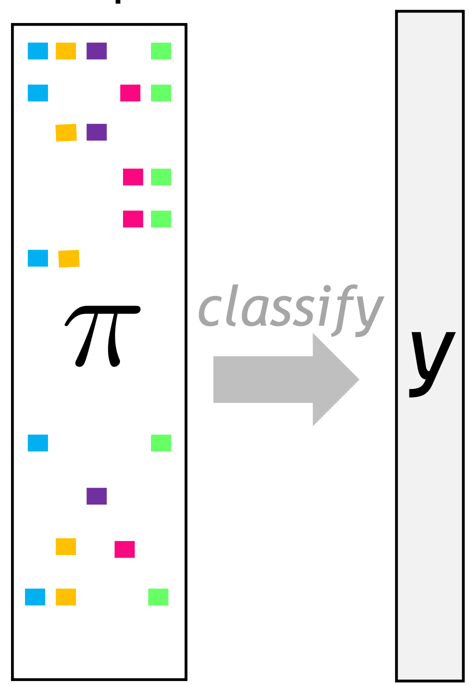


Predictions from bag-of-words

INPUT:
High-dim. codewords



INPUT:
Low-dim.
patient-topic vector



+ more interpretable!

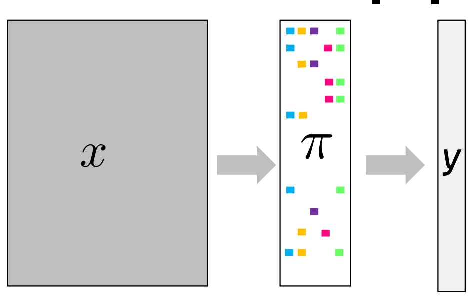
Will discovered topic features make good predictions?

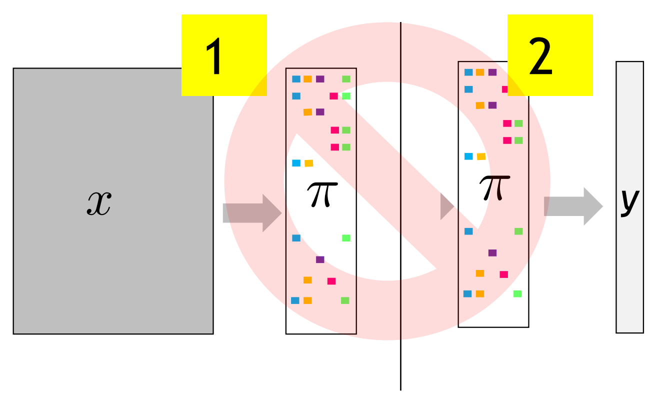
Why not bag-of-words?

- Tractable, but lose information (order matters)
- Cool models for low-dim. representations, but hard to integrate into predictive task

What might be better?

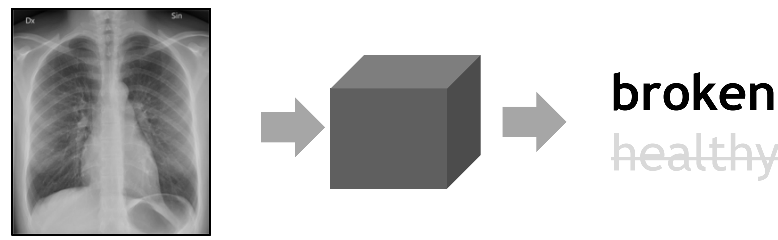
- Avoid two stage represent-then-predict
- Learn representations end-to-end
  - **one trainable pipeline**



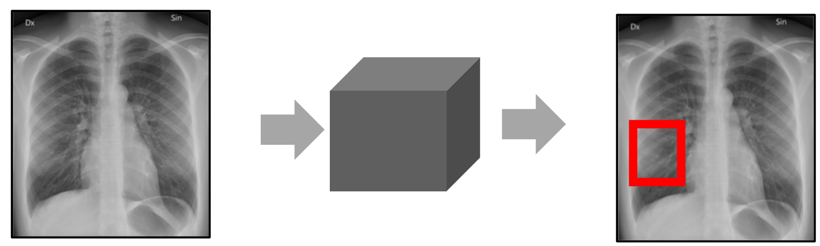


Prediction tasks with IMAGES

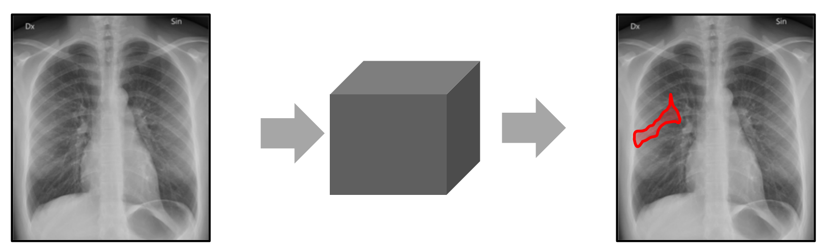
- Image classification



- Object detection

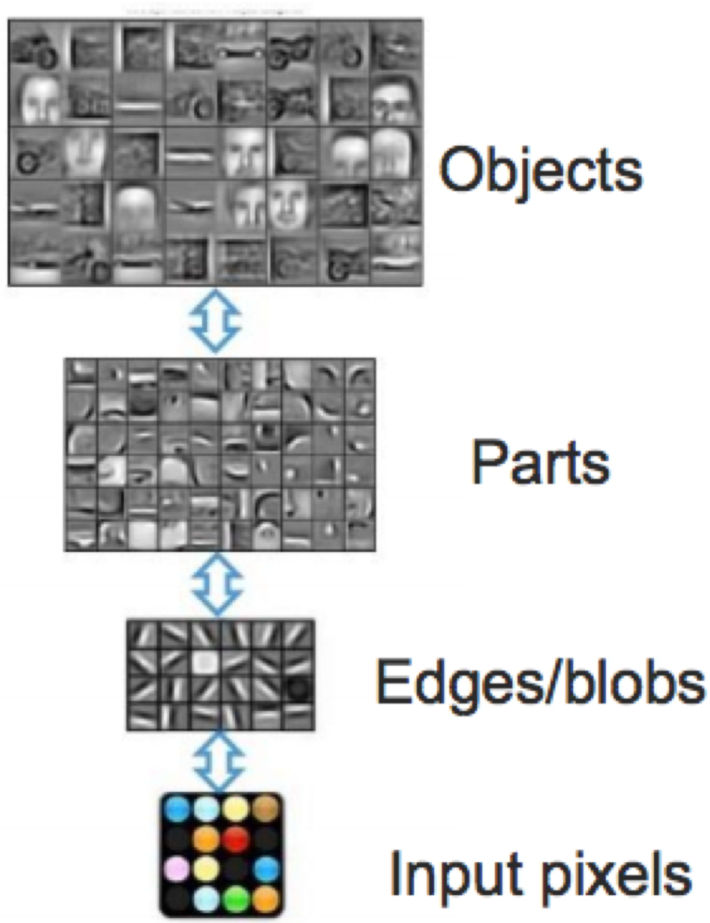


- Object segmentation



For much more, see survey on Deep Learning for Medical Images: Litjens et al. 2017

Convolutional Neural Networks (CNNs): Trainable features for images

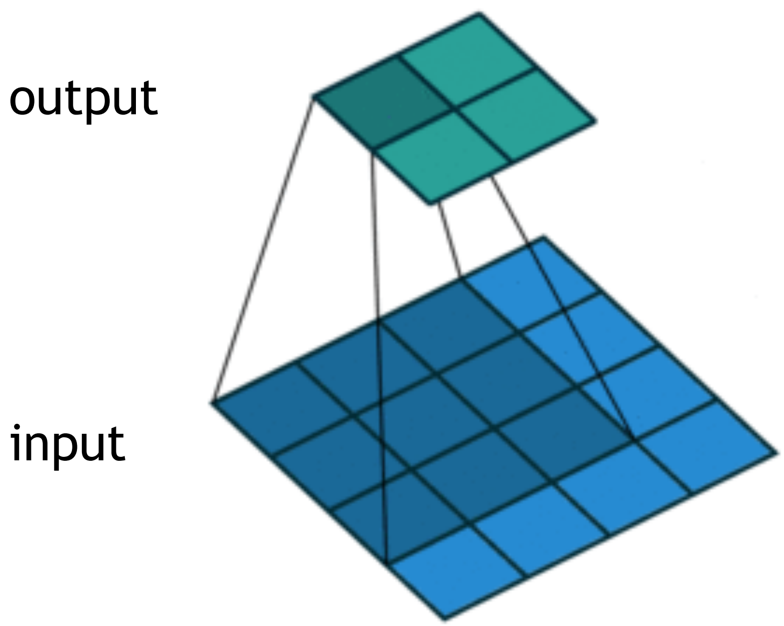


Goal: learn feature representations that:

- Represent high-level information
  - “objects” and “parts”
- Invariant to translation
  - object could appear anywhere

Credit: L.P. Morency & T. Baltrusaitis, ACL 2017 Tutorial
https://www.cs.cmu.edu/~morency/MMML-Tutorial-ACL2017.pdf

Basic 2D Convolution Operation



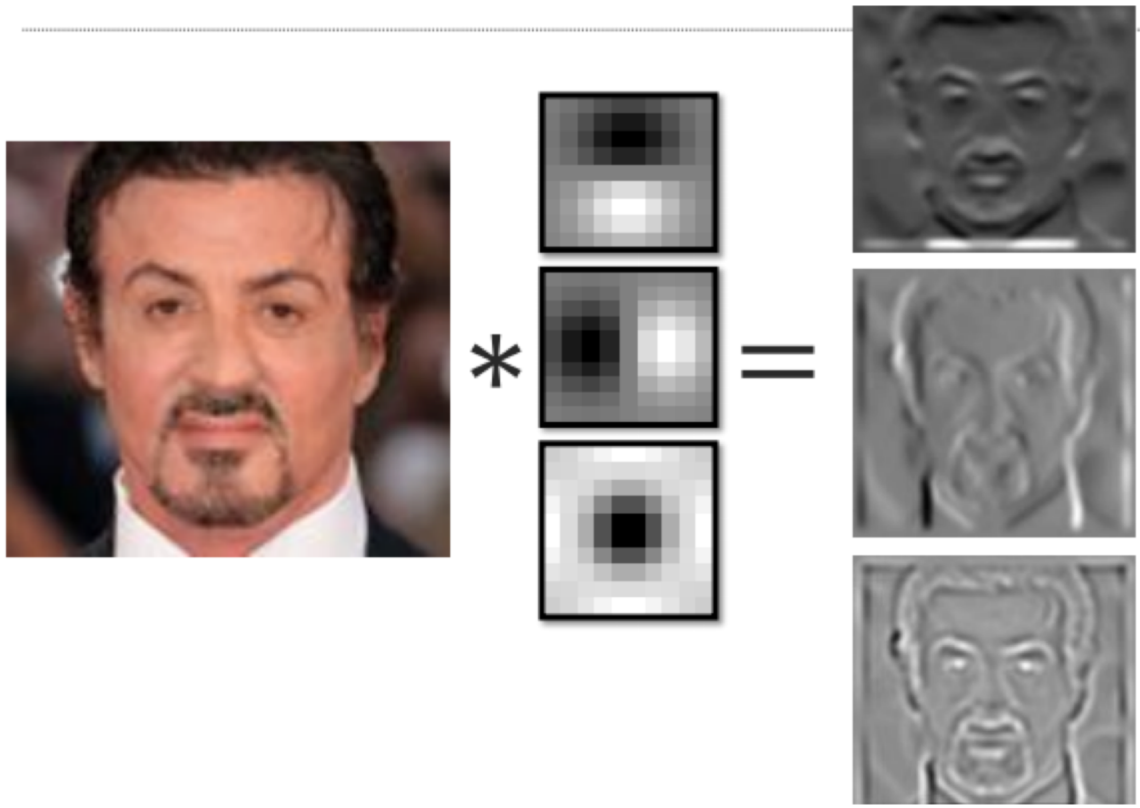
Slide same “small window”
with fixed weights
across entire image

Each output value depends
on **small subset** of input

Advantages

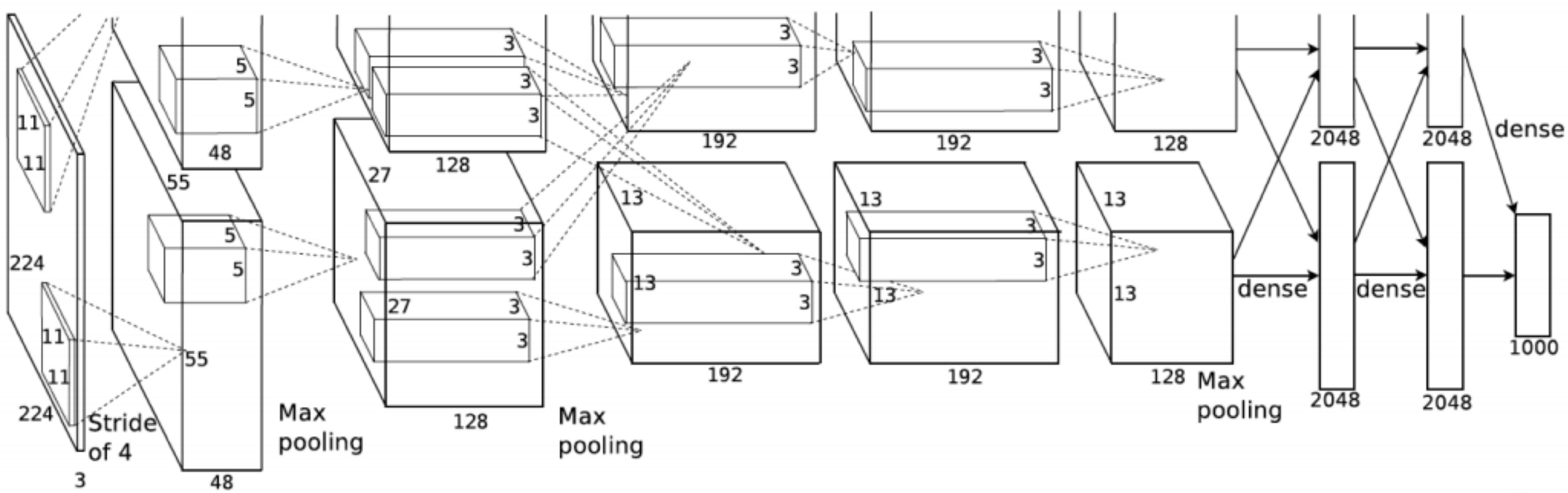
- Fewer parameters to learn
- Can detect same pattern in any position in the image

Example Convolution in 2D



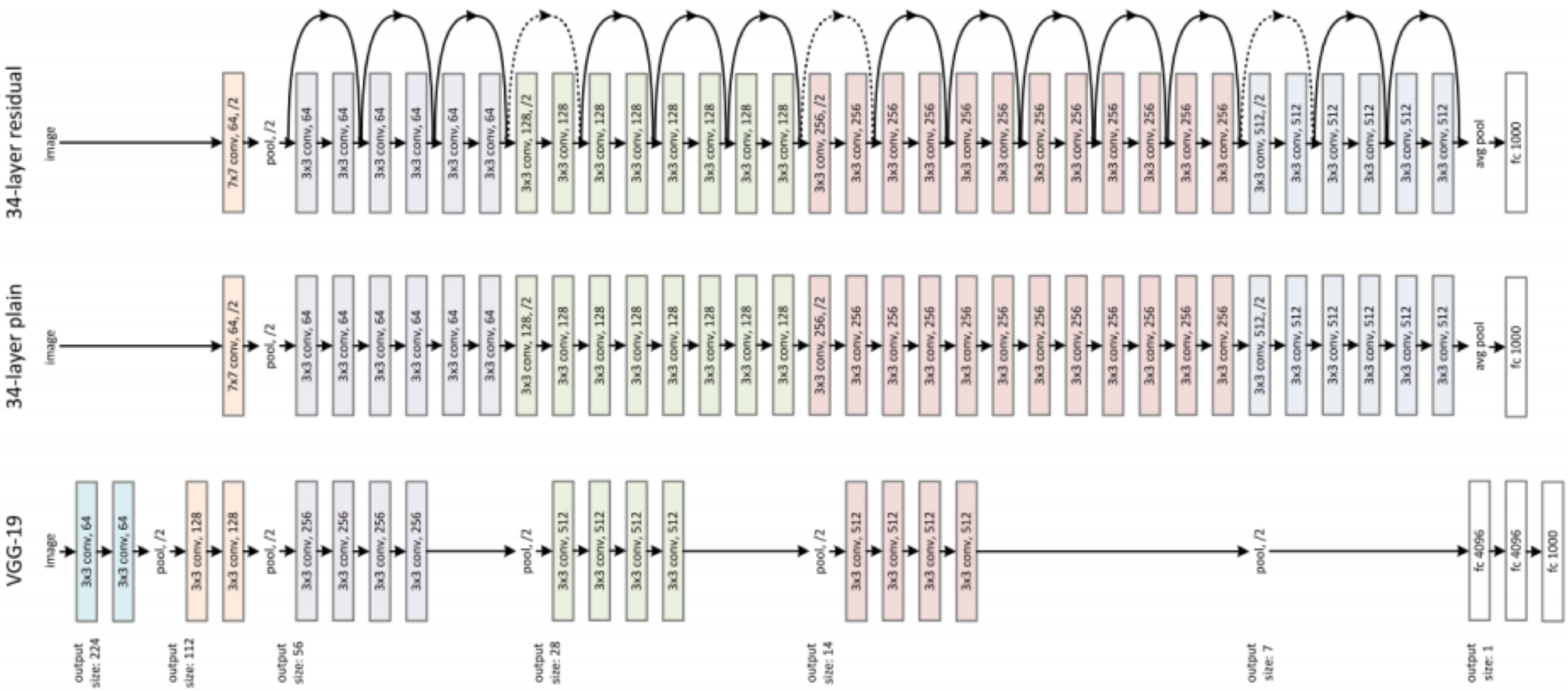
Credit: L.P. Morency & T. Baltrusaitis, ACL 2017 Tutorial
https://www.cs.cmu.edu/~morency/MMML-Tutorial-ACL2017.pdf

Deep CNN Example: AlexNet



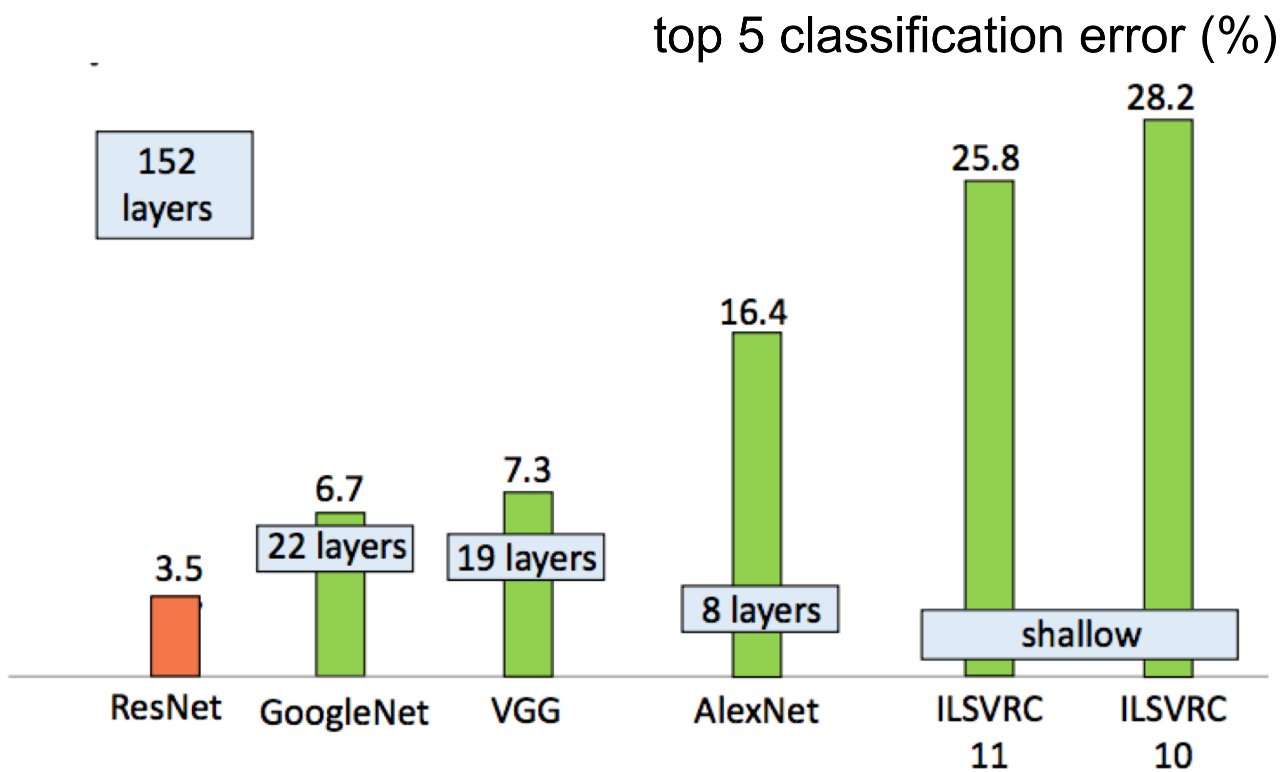
Credit: Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." NIPS 2012

Deep CNN Example: ResNet



Credit: Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun.
"Deep residual learning for image recognition”,
CVPR 2016

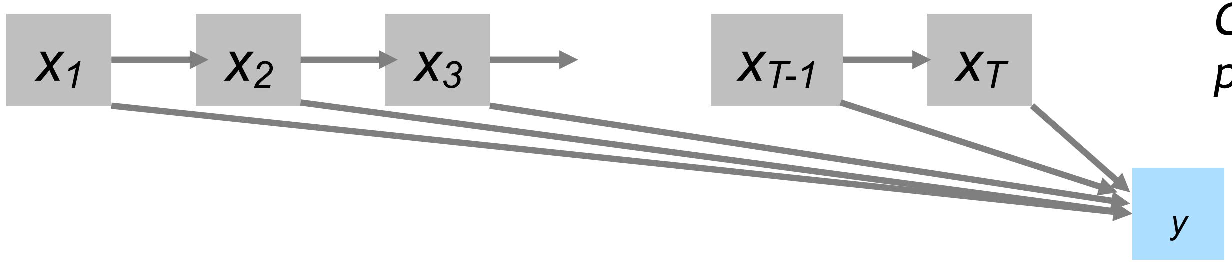
Error vs. Depth on the ImageNet benchmark



Credit: KDD Tutorial by Sun, Xiao, & Choi: http://dl4health.org/
Figure idea originally from He et. al., CVPR 2016

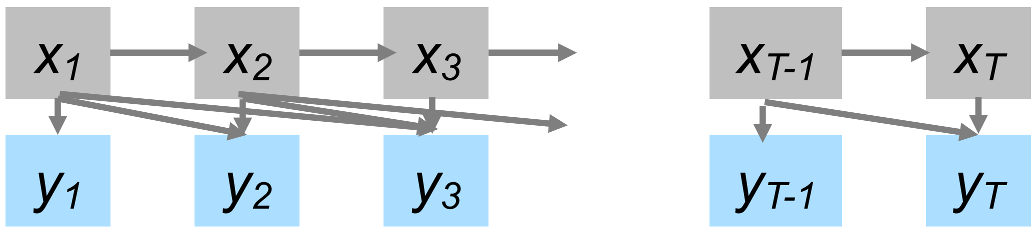
Prediction tasks for TIME SERIES

Predict one label per sequence



Given vital sign history, predict mortality risk

Predict one label per timestep



Given vital sign history, predict need for ventilator at each hour

Other tasks: “seq2seq”, where x has length T and y has length U

Assumptions for Time Series ML

1) Regular time intervals between observations

Generative Paradigm

- Hidden Markov Models
  - *Rabiner '89*
- State Space Models
  - *Kalman '60*

Deep Learning Paradigm

- Recurrent Neural Nets (RNNs)
  - LSTM
  - GRU

2) Irregular intervals?

- EITHER Deliberately model irregularity

Generative Paradigm

- Continuous Time HMMs
  - *Leiva-Murillo et al. NIPS '11*
  - *Liu et al. NIPS '15*

Deep Learning Paradigm

- Extensions of RNNs

- OR Align to a regular grid, then goto (1)

Assumptions for Time Series ML

1) Regular time intervals between observations

Generative Paradigm
- Hidden Markov Models
  - *Rabiner '89*
- State Space Models
  - *Kalman '60*

Deep Learning Paradigm
- Recurrent Neural Nets (RNNs)
  - LSTM
  - GRU

Focus here on this tutorial
- Easier to integrate with prediction task
- Easier to train end-to-end

2) Irregular intervals?

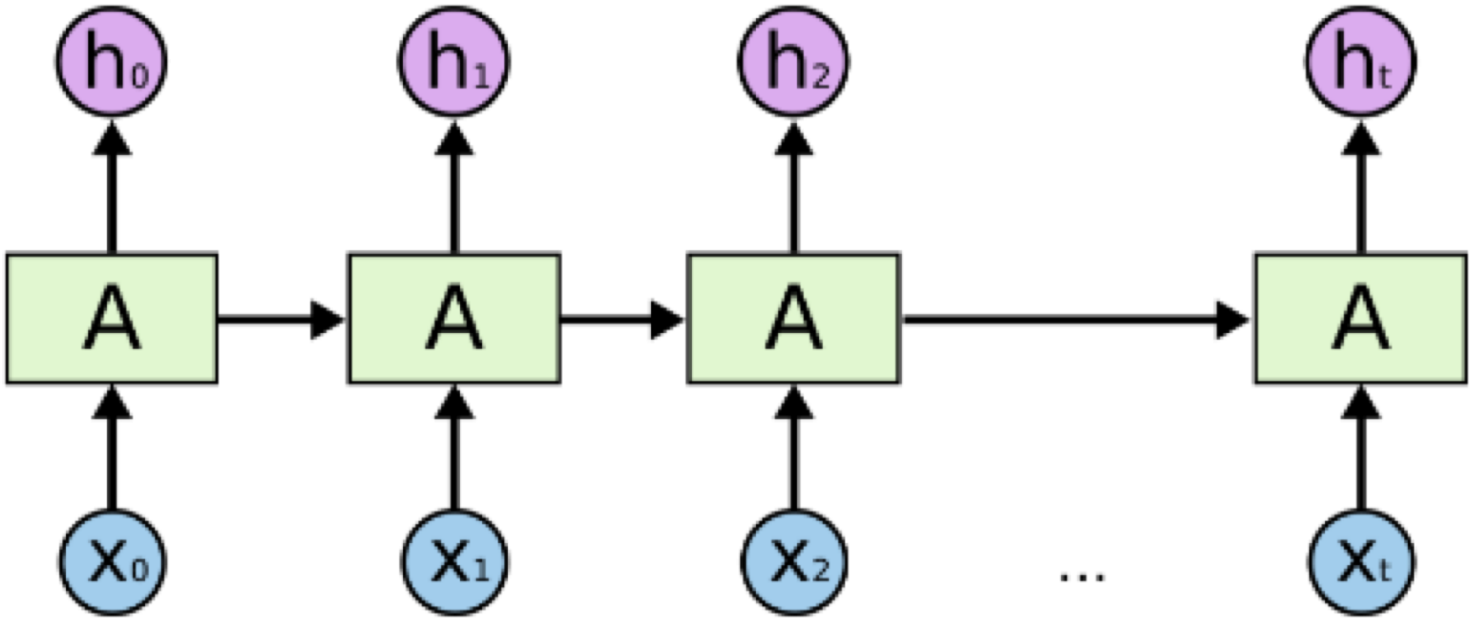
- EITHER Deliberately model i

Generative Paradigm
- Continuous Time HMMs
  - *Leiva-Murillo et al. NIPS '11*
  - *Liu et al. NIPS '15*

Deep Learning Paradigm
- Extensions of RNNs

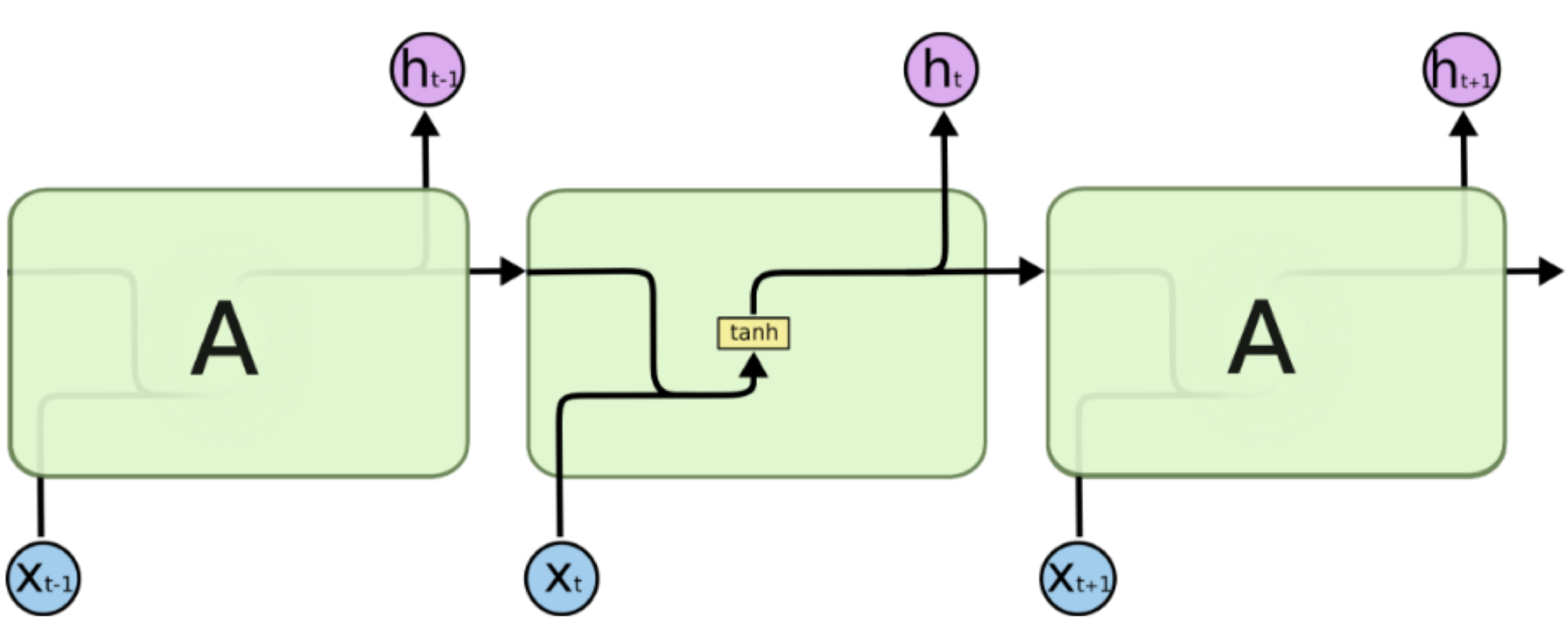
- OR Align to a regular grid, then goto (1)

Recurrent Neural Networks (RNNs)



Credit: Chris Olah http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Simple RNN unit



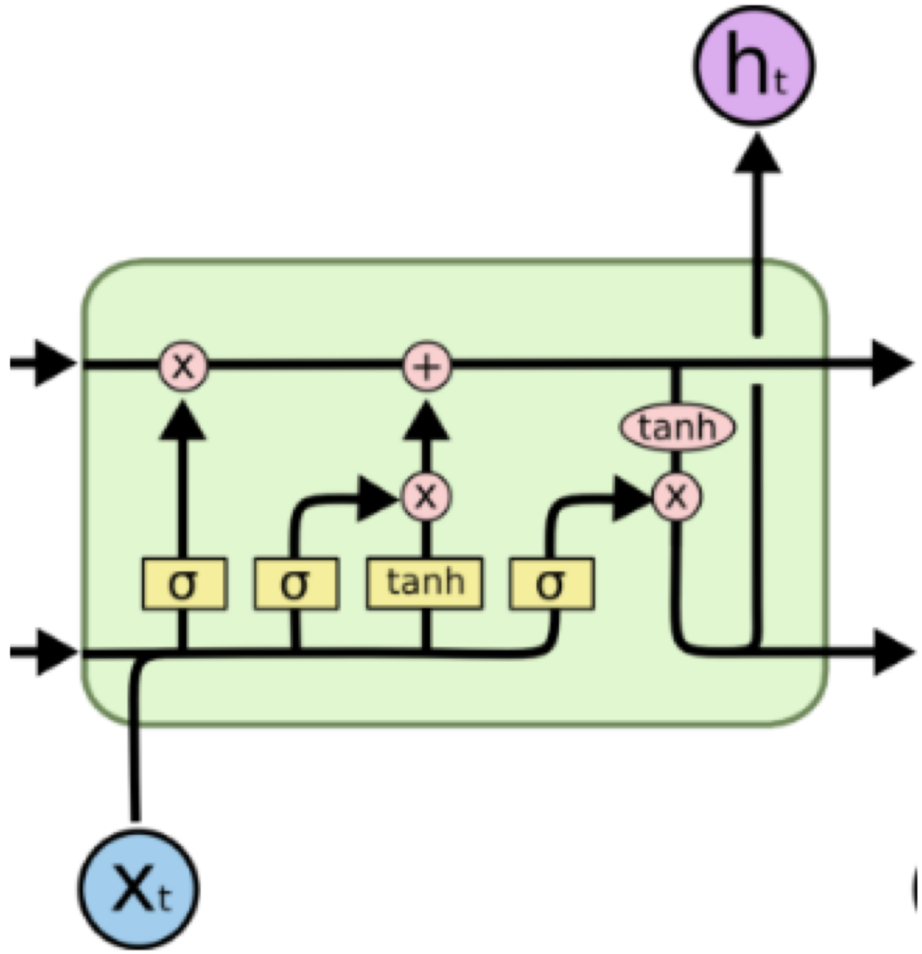
Each “A” cell shares *same* weight parameters

Credit: Chris Olah http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Long Short Term Memory unit (LSTM)

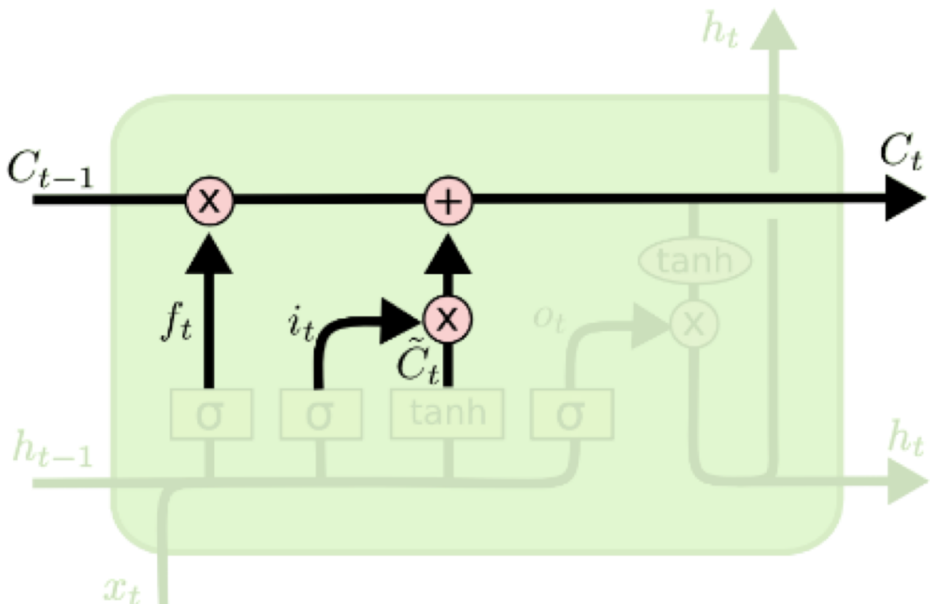
"cell state"
vector c

"hidden state"
vector h



Credit: Chris Olah http://colah.github.io/posts/2015-08-Understanding-LSTMs/

LSTM: Captures long-range info.



Settings of f and i exist that could maintain same c for any number of steps t

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

f between 0 and 1

- 0 means "forget old cell value"
- 1 means "keep old cell value"

i between 0 and 1

- 0 means "discard new cell value"
- 1 means "keep new cell value"

Credit: Chris Olah http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Learning to Diagnose with LSTMs (Lipton et al ICLR 2016)

- 10k sequences from the Pediatric ICU
- Durations of 12 hrs to several months
- 13 vital signs (blood pressure, heart rate, etc.)
- Prediction task: label each sequence with 128 separate ICD diagnoses

Each example consists of irregularly sampled multivariate time series with both missing values and, occasionally, missing variables. We resample all time series to an hourly rate, taking the mean measurement within each one hour window.

Performance: LSTM vs baseline

Model	Micro AUC	Macro AUC
Base Rate	0.7128	0.5
Log. Reg., First 6 + Last 6	0.8122	0.7404
Log. Reg., Expert features	0.8285	0.7644
MLP, First 6 + Last 6	0.8375	0.7770
MLP, Expert features	**0.8551**	**0.8030**
LSTM-DO-TR	**0.8560**	**0.8075**
Max of LSTM-DO-TR & MLP	**0.8643**	**0.8194**

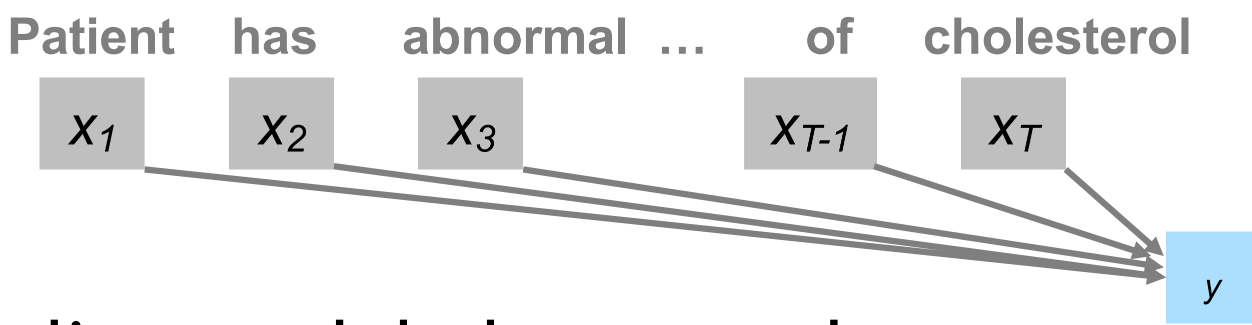
- MLP has 3 layers, layer size = 300
- LSTM has 2 layers, layer size = 128

143 "Expert features": 11 stats for each of 13 vital signs

- mean, min, max, median, slope, etc.

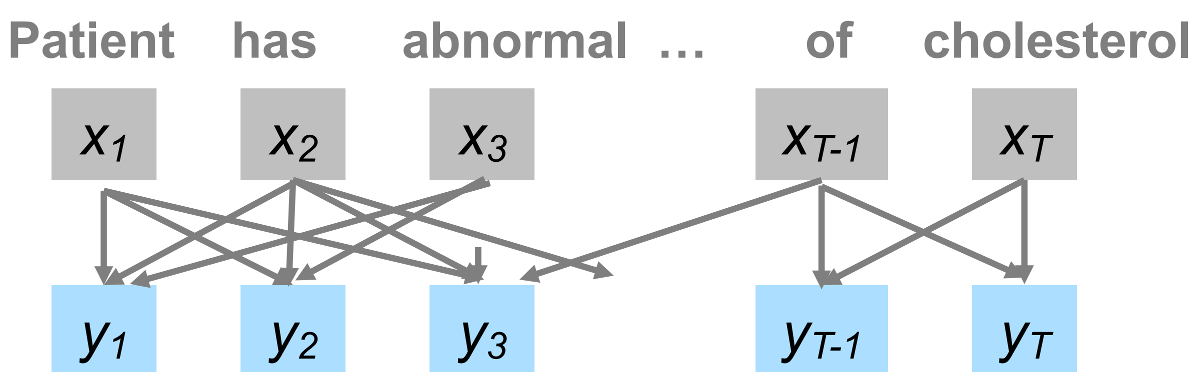
Prediction tasks with TEXT

Predict one label per sentence



Given sentence, predict mortality risk

Predict one label per word



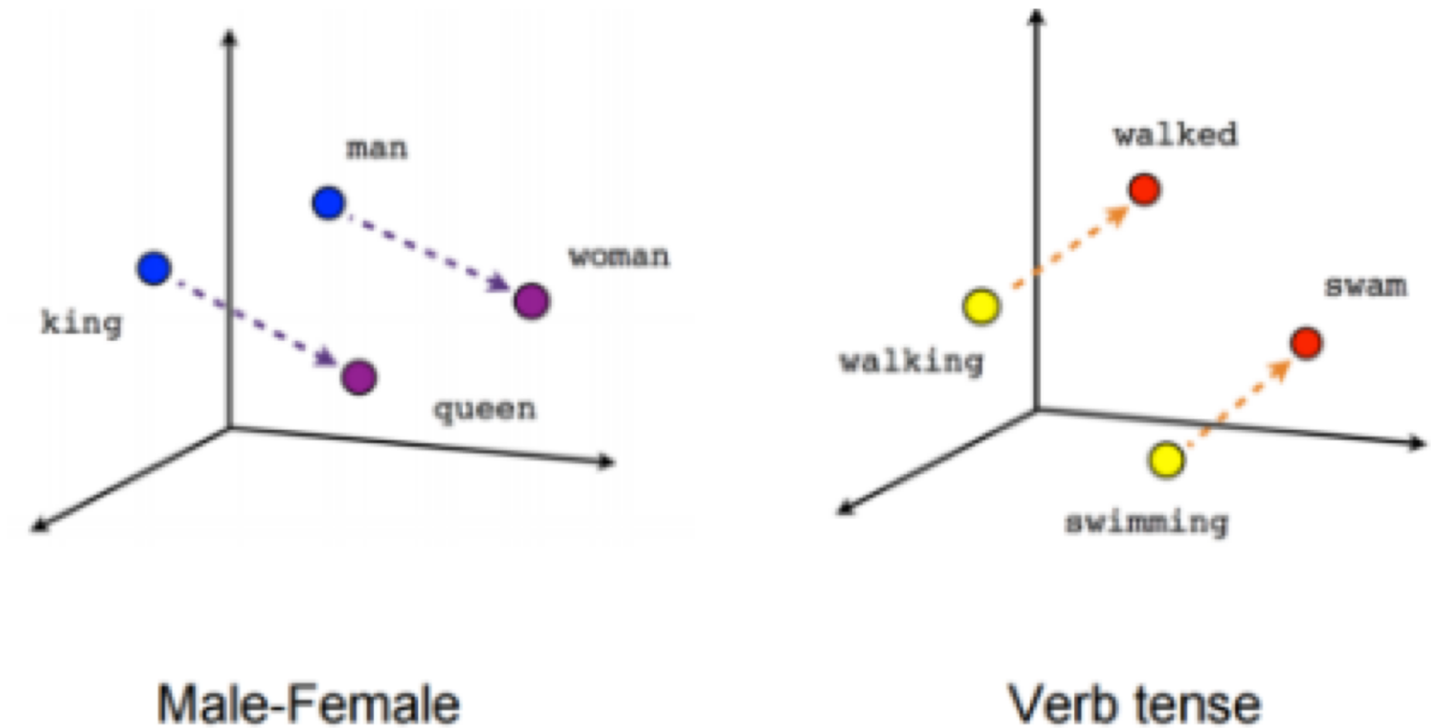
Given sentence, predict which words are relevant for a sepsis diagnosis

Unlike time-series forecasting prediction tasks, can use **bidirectional representations**

Word Embeddings (word2vec)

Goal: map each word in vocabulary to high-dimensional vector

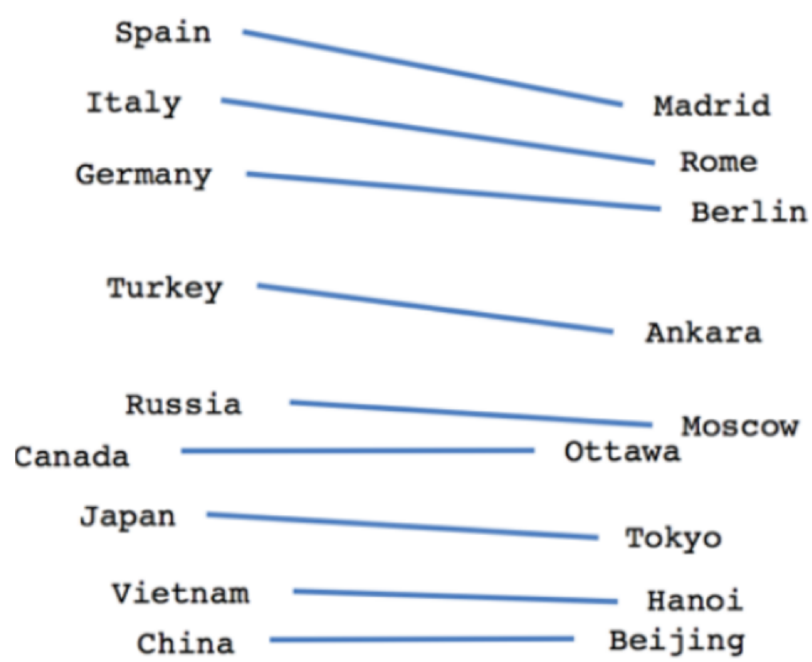
- Preserve semantic meaning in this new vector space



vec(swimming) – vec(swim) + vec(walk) = vec(walking)

Word Embeddings (word2vec)

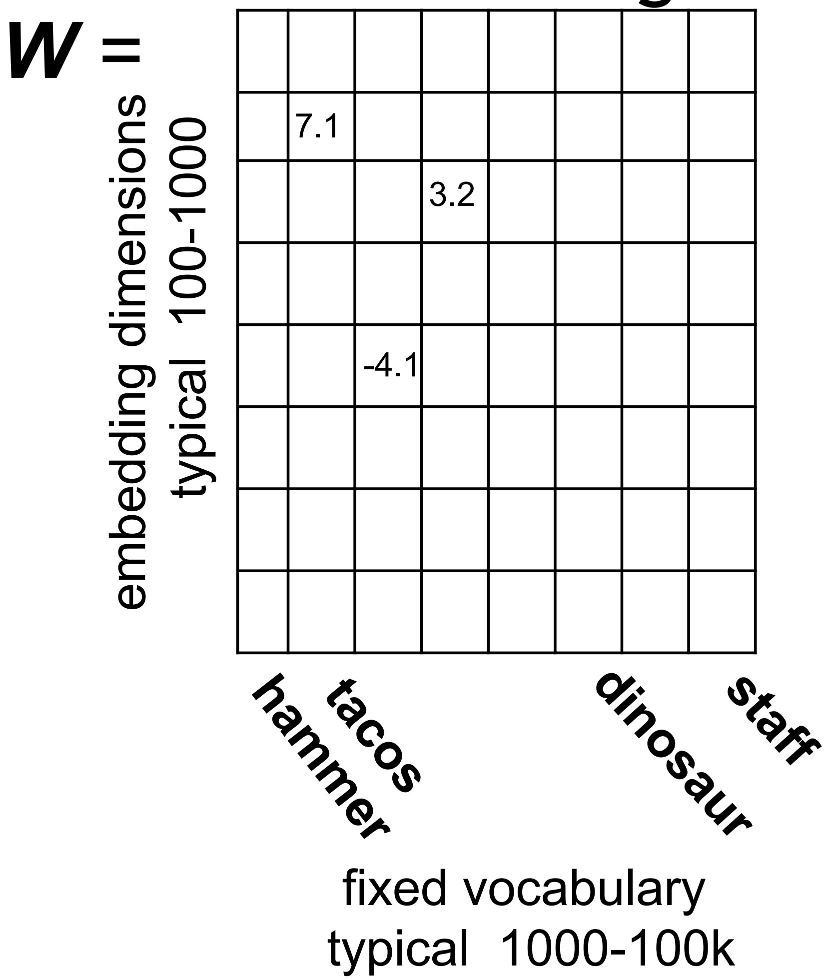
Goal: map each word in vocabulary to high-dimensional vector
- Preserve semantic meaning in this new vector space



Country-Capital

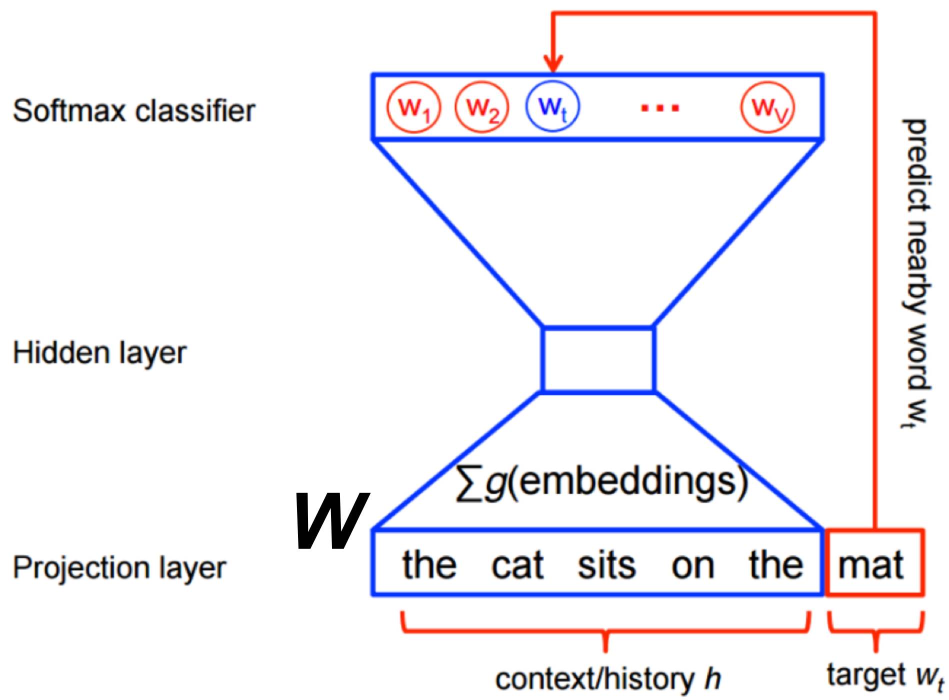
How to embed?

Goal: learn weights



Training

Reward embeddings that predict nearby words in the sentence.



Credit:
https://www.tensorflow.org/tutorials/representation/word2vec

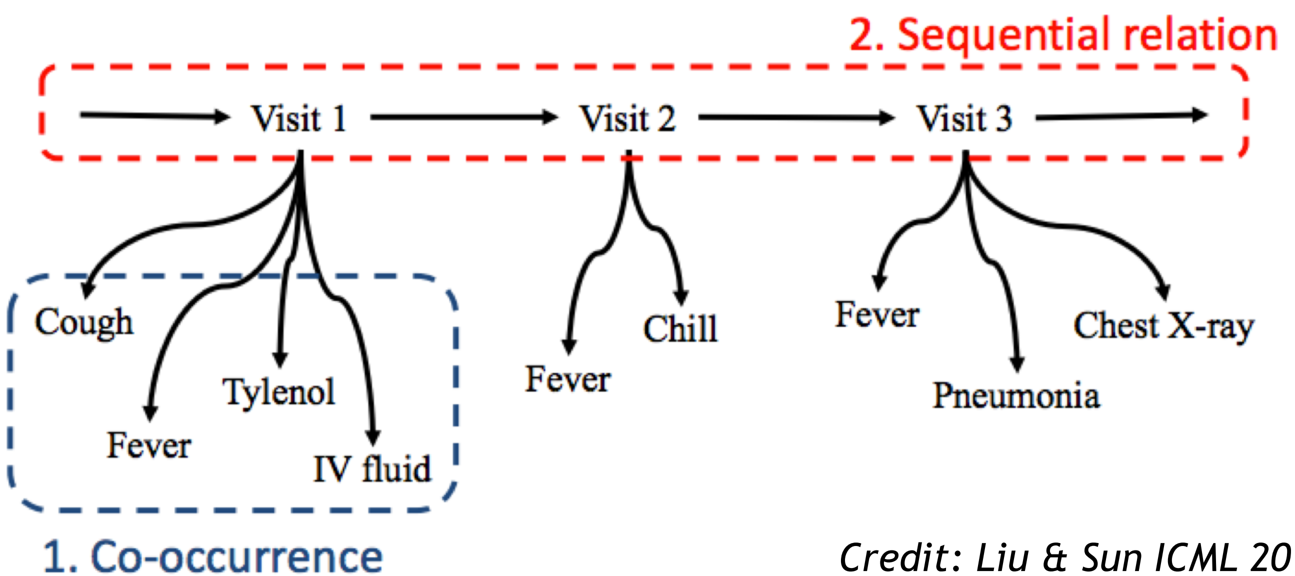
Embeddings for EHR: "med2vec"

Multi-layer Representation Learning for Medical Concepts

KDD 2016

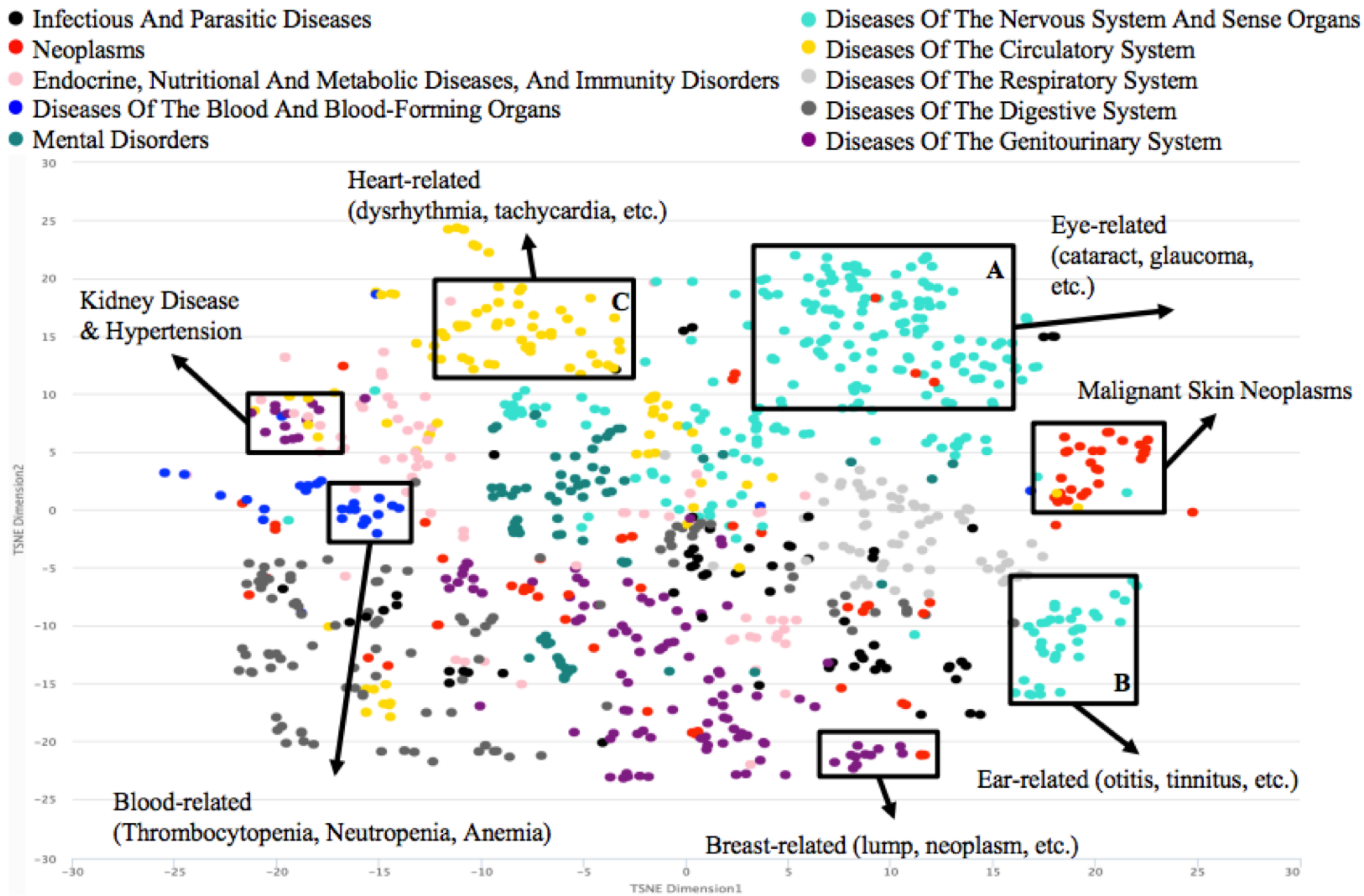
Edward Choi[1], Mohammad Taha Bahadori[1], Elizabeth Searles[2], Catherine Coffey[2], Michael Thompson[2], James Bost[2], Javier Tejedor-Sojo[2], Jimeng Sun[1]
[1]Georgia Institute of Technology [2]Children's Healthcare of Atlanta

Goal: Embed patient visits in way that predicts neighboring visits



Credit: Liu & Sun ICML 2017 Tutorial

Embeddings for EHR: “med2vec”



Credit: Liu & Sun ICML 2017 Tutorial

Embeddings for clinical notes

Radiology report annotation using intelligent word embeddings: Applied to multi-institutional chest CT cohort

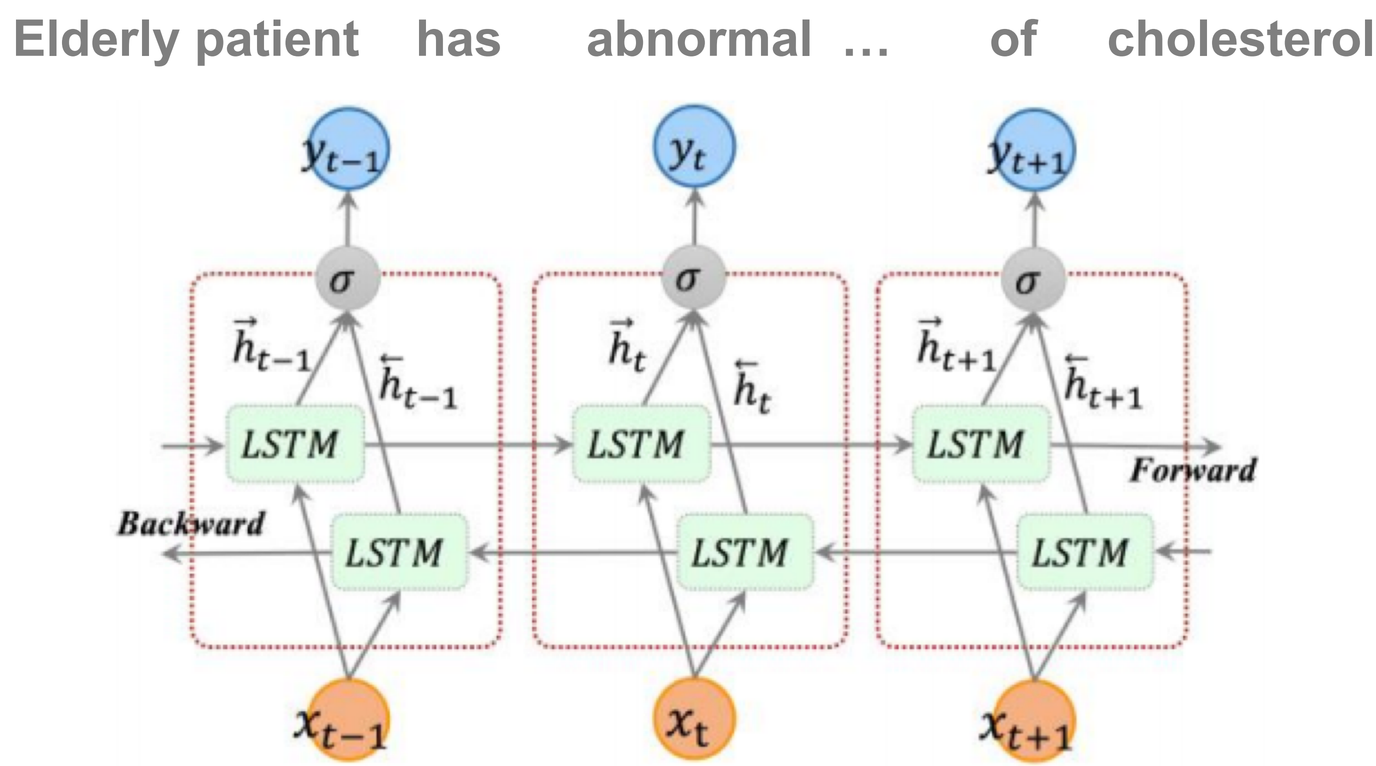
Imon Banerjee[a,*], Matthew C. Chen[b], Matthew P. Lungren[b,*,1], Daniel L. Rubin[a,b,*,1]

[a] Department of Biomedical Data Science, Stanford University, Stanford, CA, United States
[b] Department of Radiology, Stanford University, Stanford, CA, United States

Table 2
Clustered explored from IWE space using K-means++.

Clusters	Words
Cluster 1: Cancer	'carcinoma', 'metastas', 'metastasi', 'mass', 'malign', 'adenocarcinoma', 'lymphoma', 'tumor', 'lymphadenopathi', 'carcinomatosi', 'adenopathi', 'neoplasm', 'cancer', 'lymphomat', 'metastat', 'metastat_diseas',
Cluster 2: Cardiac	'ventricl', 'heart', 'pulmonari_arteri', 'atrium', 'ventricular', 'atrial'
Cluster 3: Skeletal	'boni', 'lytic', 'vertebr_bodi', 'sclerot', 'skeleton', 'bone', 'lucent', 'spine', 'sclerosi', 'osseous'
Cluster 4: Location	'right_lower', 'left_lower', 'left_upper', 'right_upper', 'upper', 'lower'
Cluster 5: Effusion	'pleural_effus', 'bilater_pleural_effus', 'left_pleural_effus', 'effus', 'right_pleura
Cluster 6: Hemorrhage/infection in lungs	'hemorrhag', 'layer', 'air', 'pneumoperitoneum', 'space', 'wound', 'hemoperitoneum', 'empyema', 'pneumothorac', 'pneumomediastinum', 'her 'blood', 'abscess', 'hydropneumothorax', 'pneumothorax', 'hemithorax', 'bronchopleur', 'pigtail', 'fluid', 'intraperiton', 'bleed', 'hematoma', 'pocket'
Cluster 7: Suspicious	'concern', 'suspici','worrisom'
....	

Bidirectional LSTM



Hidden representation at position t uses information from BOTH left and right contexts

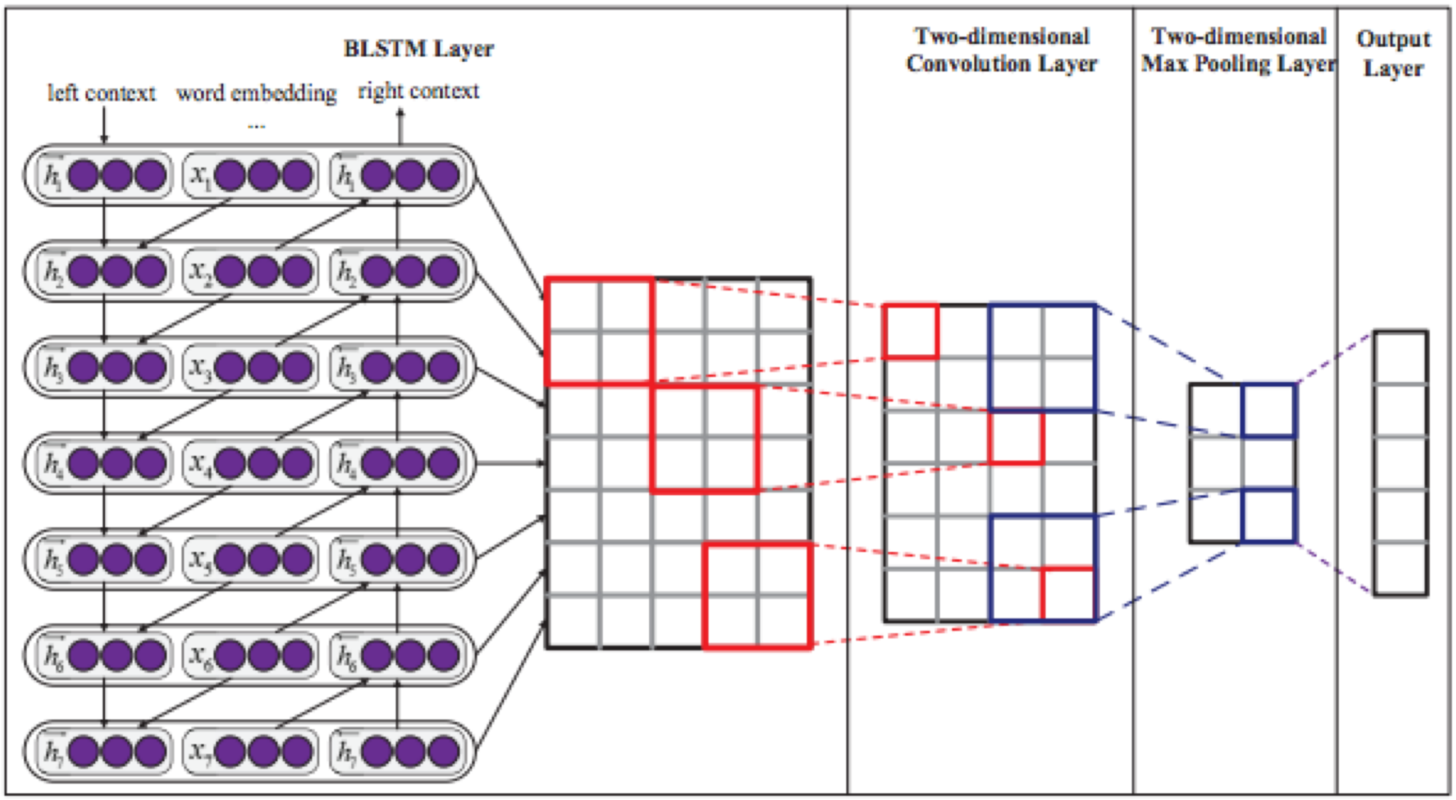
Image Credit: Cui, Ke, and Wang 2018
https://arxiv.org/abs/1801.02143

Bidir LSTM refs:
- Schuster and Paliwal 1997
- Graves & Schmidhuber 2005

Neural Net Parts are Composable
Bidirectional LSTM + Convolutions?

Text Classification Improved by Integrating Bidirectional LSTM with Two-dimensional Max Pooling

Peng Zhou[1], Zhenyu Qi[1]; Suncong Zheng[1], Jiaming Xu[1], Hongyun Bao[1], Bo Xu[1,2]

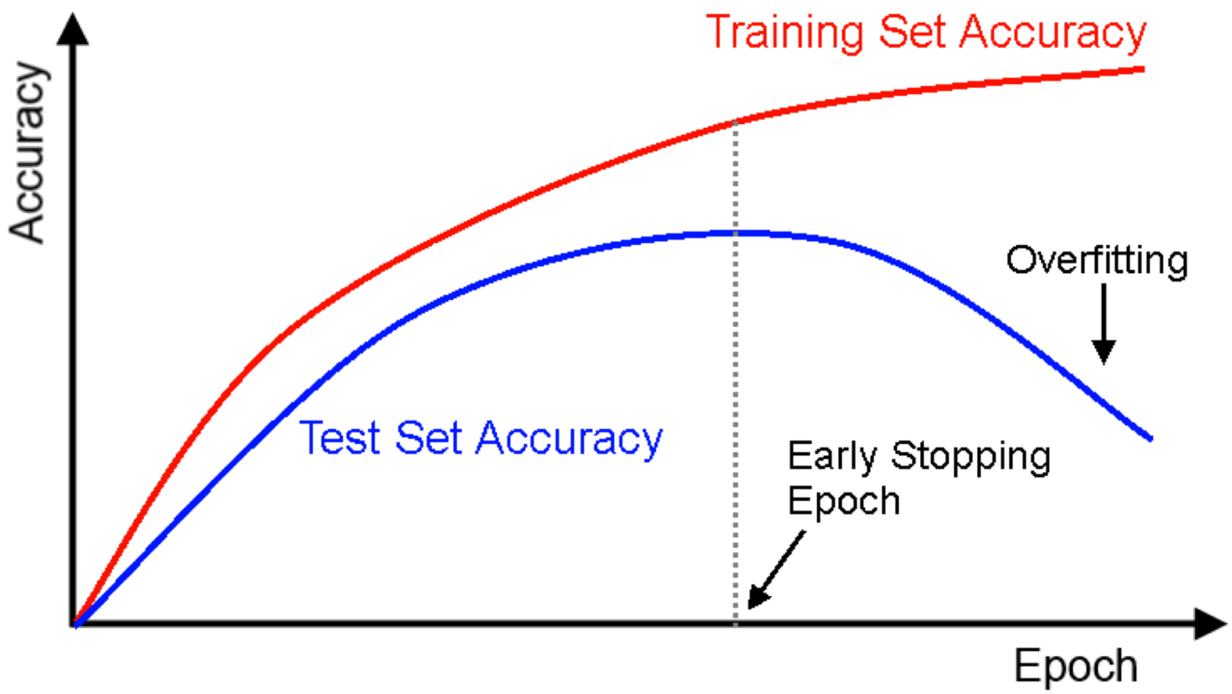


Part 2 outline

- 2-stage hand-engineered representations of data
  - Bag of words for images, text, EHR codes
- Learnable representations of data
  - Images
  - Time series
  - Text
- **Tricks of the trade**
- Models that generate data

Trick: Early Stopping

What clinically relevant signal should we be using?



Big idea: stop training after your heldout set stops improving
- Avoid overfitting
- Save time / compute resources

Credit: https://deeplearning4j.org/docs/latest/deeplearning4j-nn-early-stopping

Tricks: Data Augmentation

Data Augmentation: Increase effective size of dataset by applying small, random perturbations to features during training.

Choose perturbations which do not change label.

Images
- Flip left-to-right
- Slight rotations or crops
- Recolor or brighten

Text
- Add slight misspellings
- Replace word with similar word

This scheme approximately captures an important property of natural images, namely, that object identity is invariant to changes in the intensity and color of the illumination. This scheme reduces the top-1 error rate by over 1%.

from AlexNet paper (Krizhevsky et al. NIPS 2012)

Data Augmentation for Melanoma Classification

What clinically relevant process should we be using?

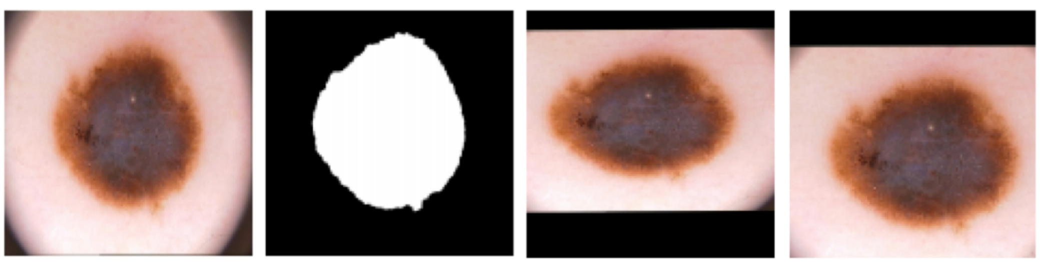
INCREASING DEEP LEARNING MELANOMA CLASSIFICATION BY CLASSICAL AND EXPERT KNOWLEDGE BASED IMAGE TRANSFORMS

*Cristina Nader Vasconcelos**

Departamento de Ciência da Computação
Instituto de Computação
Universidade Federal Fluminense, Brazil

Bárbara Nader Vasconcelos

Serviço de Dermatologia
Hospital Universitário Pedro Ernesto (Hupe)
Universidade Estadual do Rio de Janeiro, Brazil



(a) Original (b) Segmentation (c) Artificial 1 (d) Artificial 2

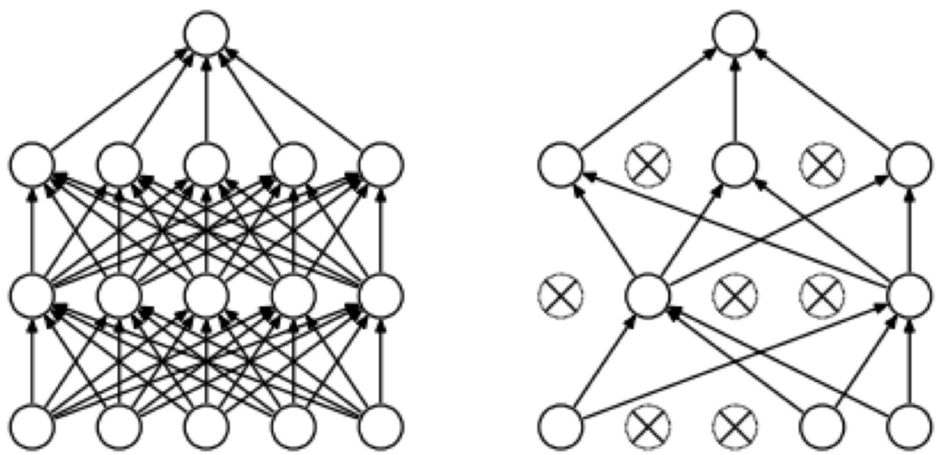
Fig. 2. Distortion of the original image by lesion axis analysis

Tricks: Dropout

Journal of Machine Learning Research 15 (2014) 1929-1958 Submitted 11/13; Published 6/14

Dropout: A Simple Way to Prevent Neural Networks from Overfitting

Nitish Srivastava NITISH@CS.TORONTO.EDU
Geoffrey Hinton HINTON@CS.TORONTO.EDU
Alex Krizhevsky KRIZ@CS.TORONTO.EDU
Ilya Sutskever ILYA@CS.TORONTO.EDU
Ruslan Salakhutdinov RSALAKHU@CS.TORONTO.EDU



(a) Standard Neural Net (b) After applying dropout.

Credit: Srivastava et al. JMLR 2014

Sample at train, downweight at test

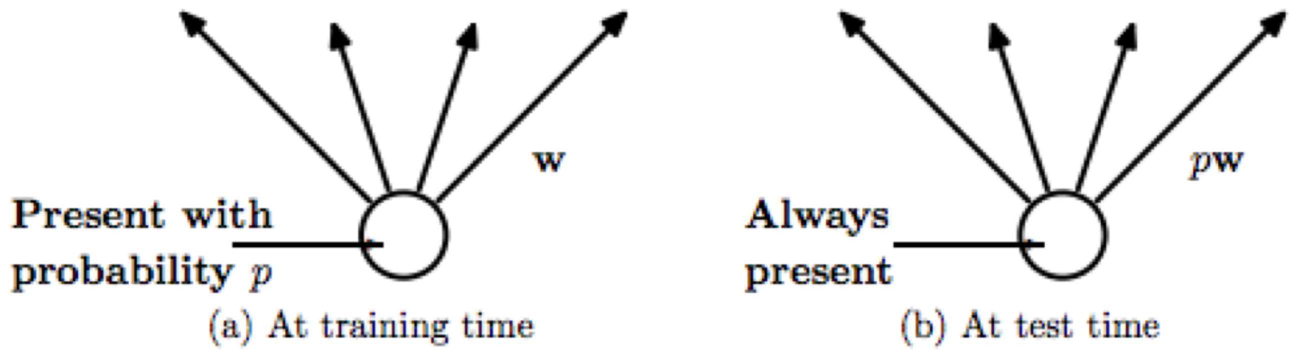


Figure 2: **Left**: A unit at training time that is present with probability p and is connected to units in the next layer with weights $\mathbf{w}$. **Right**: At test time, the unit is always present and the weights are multiplied by p. The output at test time is same as the expected output at training time.

In practice, often set dropout probabilities to 50% for hidden units
20% for input units

Credit: Srivastava et al. JMLR 2014

Dropout Benefits

Decent gains on many tasks (images, genes, sequences)

- *over other regularization (L1/L2) and other models*

- MNIST images

Method	Test Classification error %
L2	1.62
L2 + L1 applied towards the end of training	1.60
L2 + KL-sparsity	1.55
Max-norm	1.35
Dropout + L2	1.25
Dropout + Max-norm	**1.05**

Table 9: Comparison of different regularization methods on MNIST.

lower
is better

- Genetics

Method	Code Quality (bits)
Neural Network (early stopping) (Xiong et al., 2011)	440
Regression, PCA (Xiong et al., 2011)	463
SVM, PCA (Xiong et al., 2011)	487
Neural Network with dropout	567
Bayesian Neural Network (Xiong et al., 2011)	**623**

Table 8: Results on the Alternative Splicing Data Set.

higher
is better

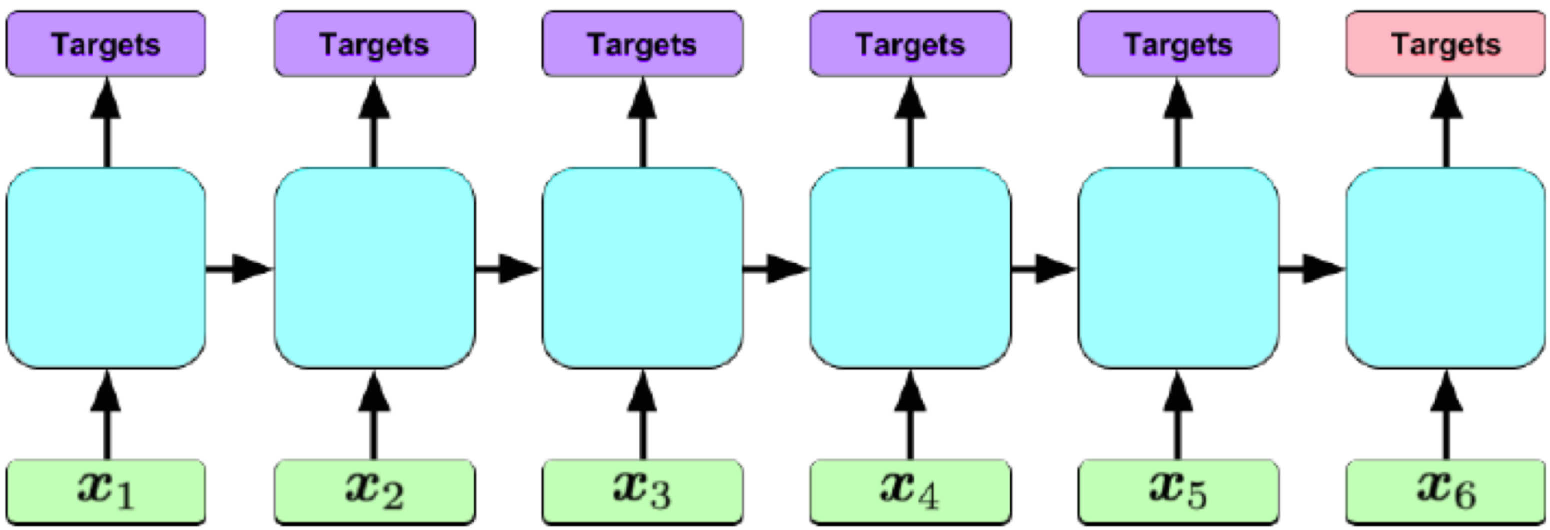
Credit: Srivastava et al. JMLR 2014

Tricks: Label/Target Replication

Lipton et al. 2016: "Learning to Diagnose with LSTMs"

At test time, make one prediction per sequence

At train time, **duplicate target label at every timestep**



$$\alpha \cdot \frac{1}{T} \sum_{t=1}^{T} \text{loss}(\hat{\boldsymbol{y}}^{(t)}, \boldsymbol{y}^{(t)}) + (1-\alpha) \cdot \text{loss}(\hat{\boldsymbol{y}}^{(T)}, \boldsymbol{y}^{(T)})$$

Tricks: Label/Target Replication

Lipton et al. 2016: "Learning to Diagnose with LSTMs"

AUC scores for 128 separate binary diagnostic predictions

	Micro AUC	Macro AUC
LSTM Models with Dropout (probability 0.5)		
LSTM-DO	0.8377	0.7741
LSTM-DO-TR	**0.8560**	**0.8075**

Adding TR leads to modest improvements

Part 2 outline

2-stage hand-engineered representations of data

- Bag of words for images, text, EHR codes

Learnable representations of data

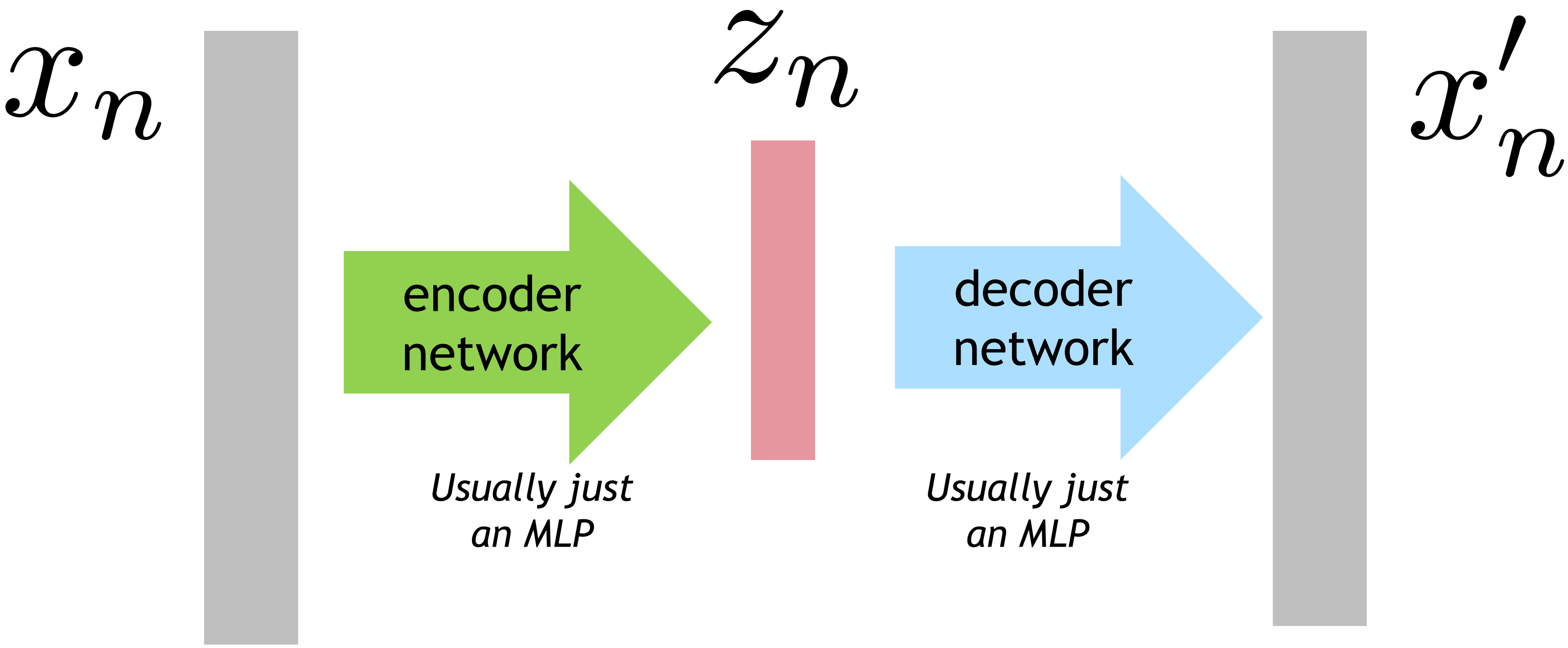
- Images
- Time series
- Text

- Tricks of the trade
- Models that generate data

Autoencoder Neural Networks

Goal: *Compress but retain information!*

- Encode each input feature vector into low-dim. vector
- Decode back into feature vector with little information loss



Use cases: images, text, EHR, etc.

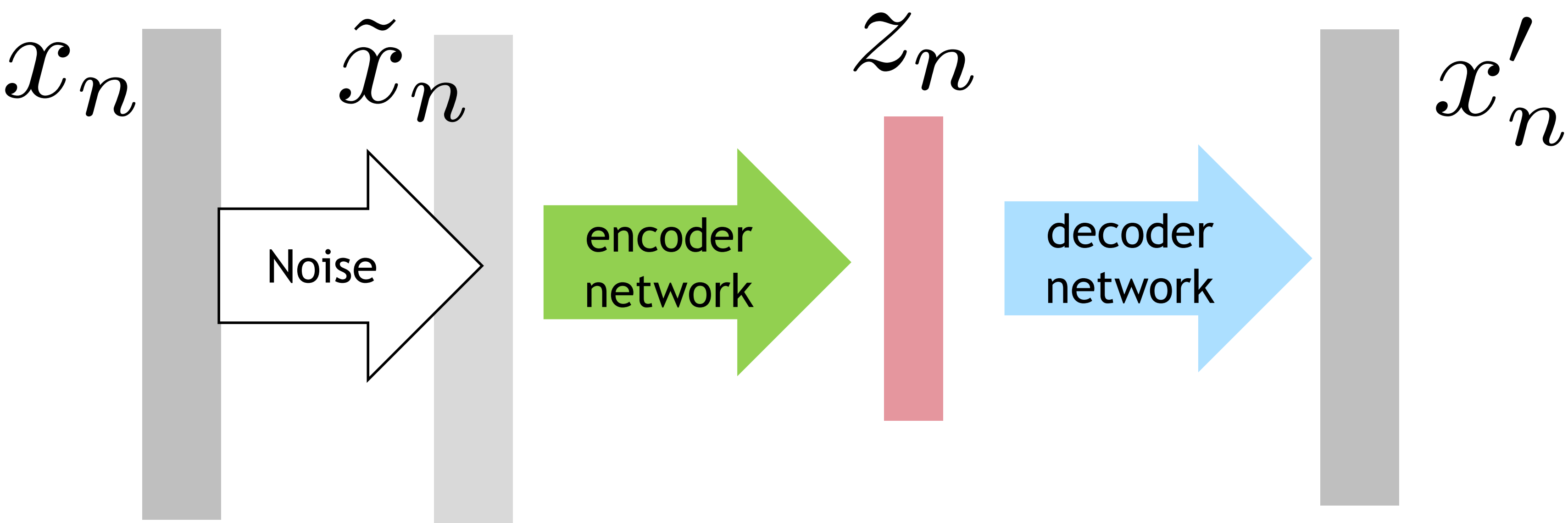
- Use low-dim features for prediction
- Inspect low-dim features for patterns
- Easy storage / fast processing

Training: Optimize encoder & decoder weights to minimize reconstruction error

Denoising Autoencoders

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders, ICML' 08

Goal: Improve robustness by adding noise to data when training



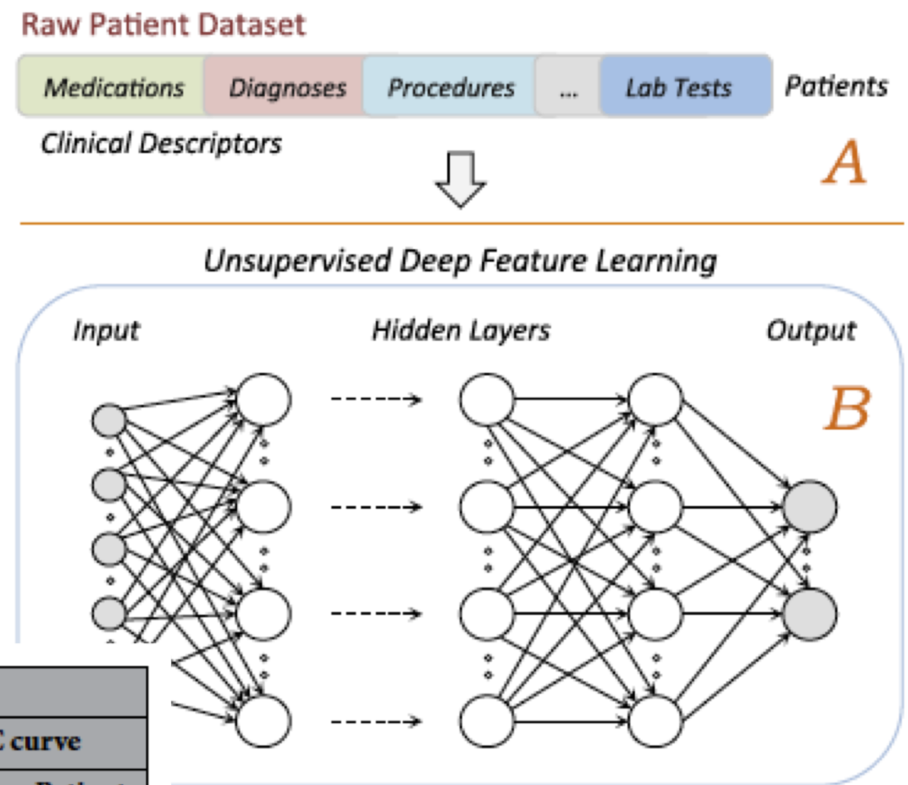
Important: Noise process should match input domain

Training: Optimize encoder & decoder weights to minimize reconstruction error of clean input

Autoencoders for EHR

Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records

Riccardo Miotto[1,2,3], Li Li[1,2,3], Brian A. Kidd[1,2,3], Joel T. Dudley[1,2,3]



Noise → "Masking"
Set 5% random entries to 0

Two stage training:
- Learn autoencoder
- Learn predictor from autoencoder

Time Interval = 1 year (76,214 patients)			
	Area under the ROC curve		
Disease	**RawFeat**	**PCA**	**DeepPatient**
Diabetes mellitus with complications	0.794	0.861	**0.907**
Cancer of rectum and anus	0.863	0.821	**0.887**
Cancer of liver and intrahepatic bile duct	0.830	0.867	**0.886**
Regional enteritis and ulcerative colitis	0.814	0.843	**0.870**
Congestive heart failure (non-hypertensive)	0.808	0.808	**0.865**

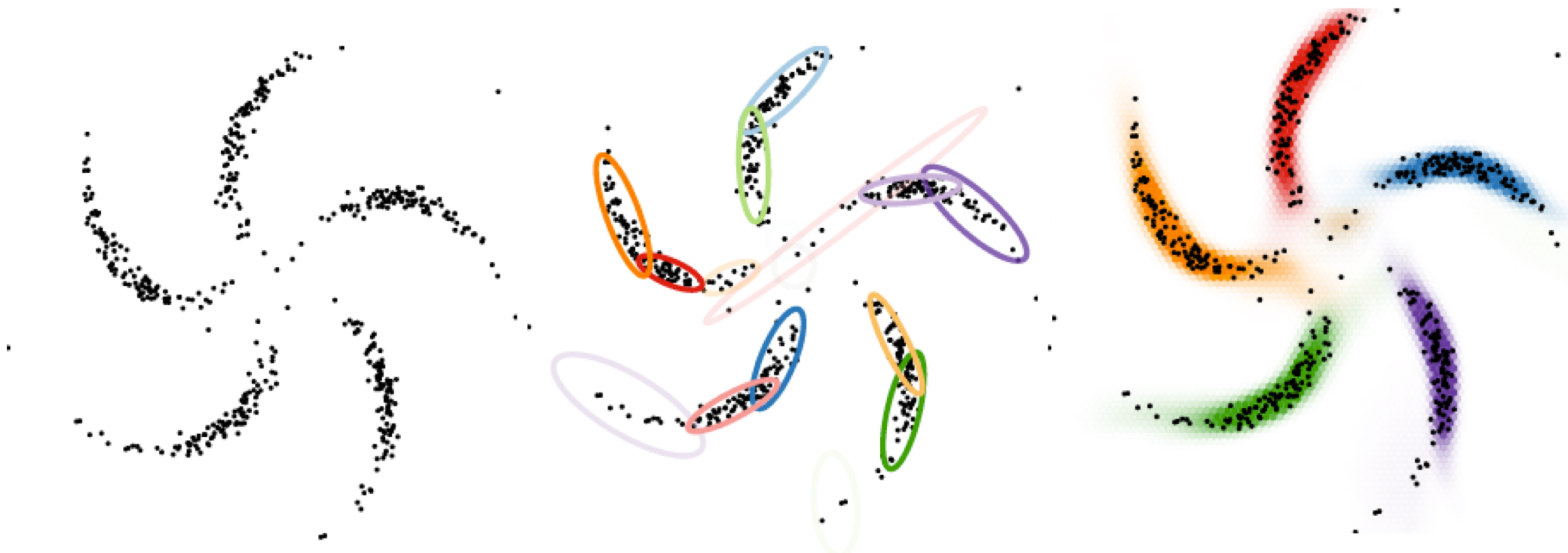
```
Raw         : size 41072
PCA         : size 100*
DeepPatient : size 500

* Best size on validation set(?)
```

Generative Models

Raw Data

Gaussian Mixture

Gaussian Mixture
+ Neural Net likelihood



Credit: Johnson et al. NIPS 2016

Generative Models

Classic Generative Models

e.g. Gaussian mixtures & extensions

PRO

- Identify outliers/anomalies
- Estimate uncertainty
- Can use data with missing values

CON

- Bespoke inference (1+ months for algo. for each new model)
- Limited expressivity: using classic distribution building blocks like Gaussians

Deep Generative Models

e.g. “deep” Gaussian mixtures

PRO

- Benefits of classic framework, plus
- Flexible data generation
- Black-box inference

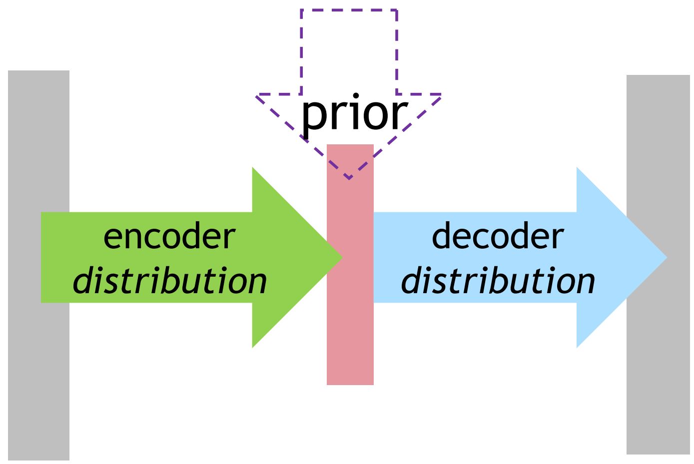
CON

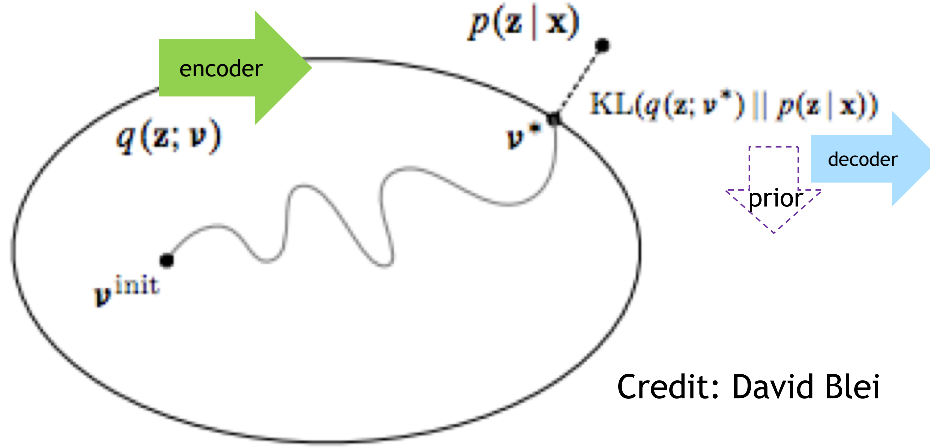
- **Is inference good enough?**
- Interpretation?

Variational Autoencoders (VAEs)

Goal: train deep generative models

- "AE": Each data example ***sampled*** from latent space
- "V": Use variational inference to approximate posterior



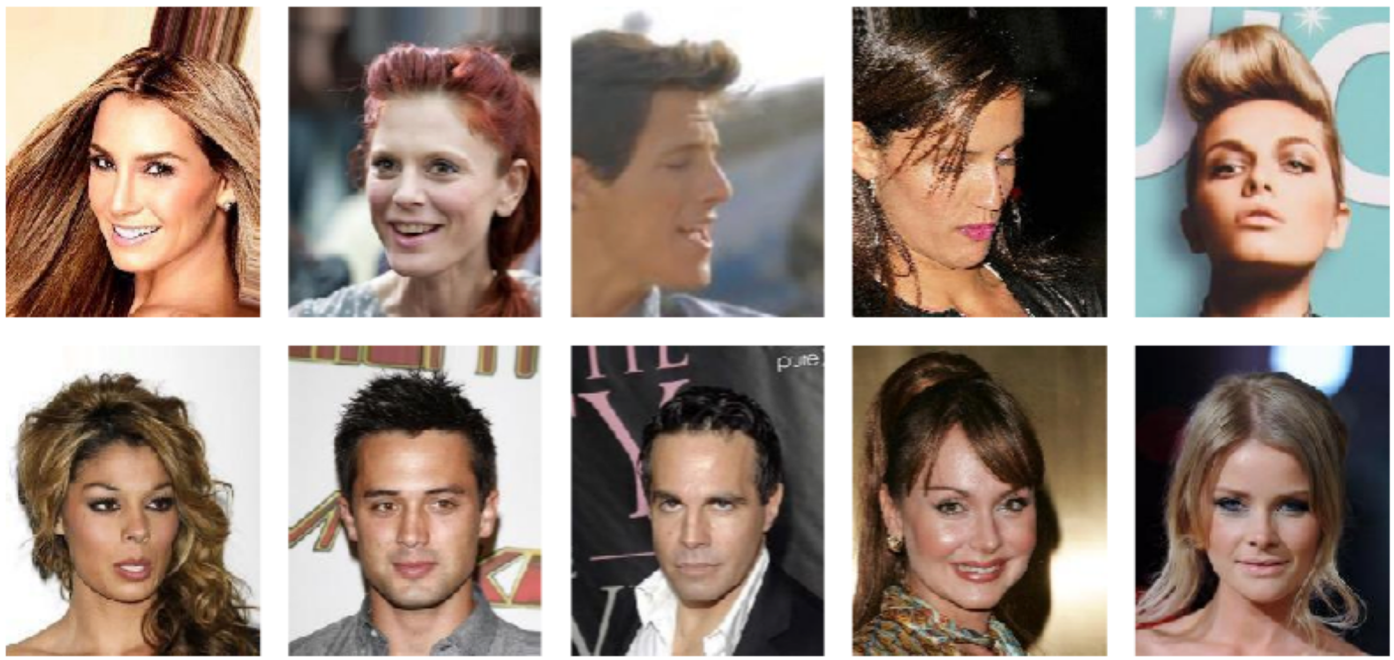


Credit: David Blei

Big idea: We can learn *distributions* over possible embeddings

- Patient with long history has more certain embedding
- Patient with little history could take many possible values

How to build high-quality generative models?



Training Data
(CelebA)

Generative Adversarial Net (GAN)

3.5 Years of Progress on Faces

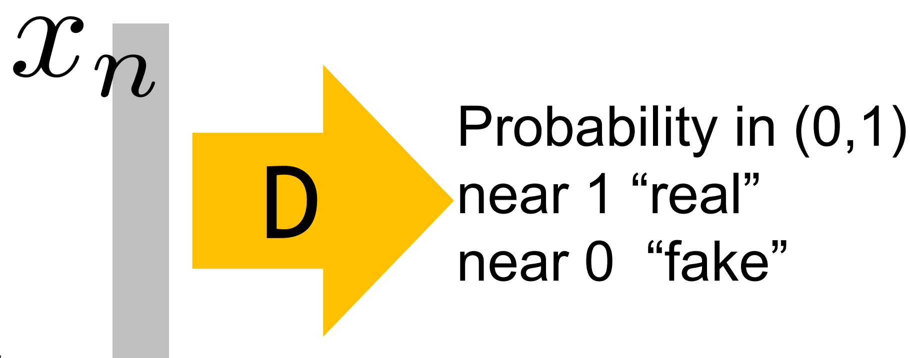
(Brundage et al, 2018)

How do GANs work?

Two player "game"

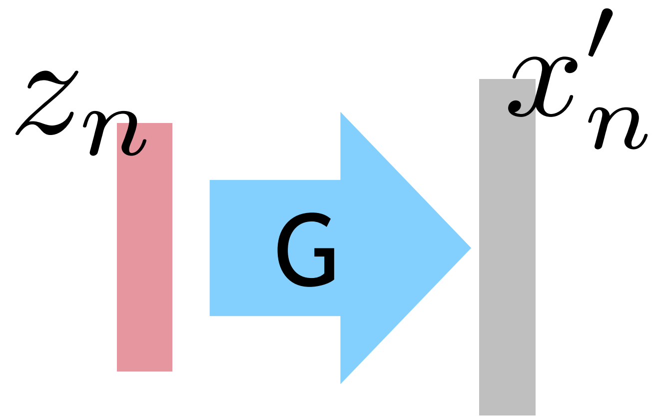
Discriminator:

Identify if feature vector
comes from training data or not



Generator:

Turn low-dim random noise
Into "plausible" data vectors



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

medGAN for Health Data

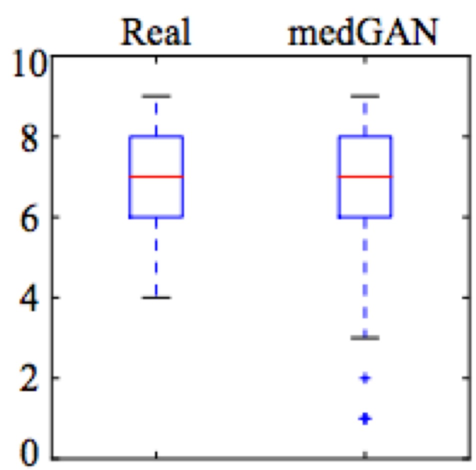
Generating Multi-label Discrete Patient Records using Generative Adversarial Networks

Edward Choi[1] MP2893@GATECH.EDU
Siddharth Biswal[1] SBISWAL7@GATECH.EDU
Bradley Malin[2] BRADLEY.MALIN@VANDERBILT.EDU
Jon Duke[1] JON.DUKE@GATECH.EDU
Walter F. Stewart[3] STEWARWF@SUTTERHEALTH.ORG
Jimeng Sun[1] JSUN@CC.GATECH.EDU

[1] GEORGIA INSTITUTE OF TECHNOLOGY [2] VANDERBILT UNIVERSITY [3] SUTTER HEALTH

Ask doc:
How realistic is the record?

Scale 1-10



Outliers:
* medGAN sometimes generates records with both male and female gender-specific codes

End of Part 2: Best Practice Summary

Do: End-to-end training of representations if your goal is prediction quality

Do: Use clinical knowledge to improve tricks like dropout, early stopping, data augmentation

Do NOT: Blindly trust reproducibility of published methods