Deep Learning Architectures and Final Project

Yale

CPSC/AMTH 452/552

CBB 663

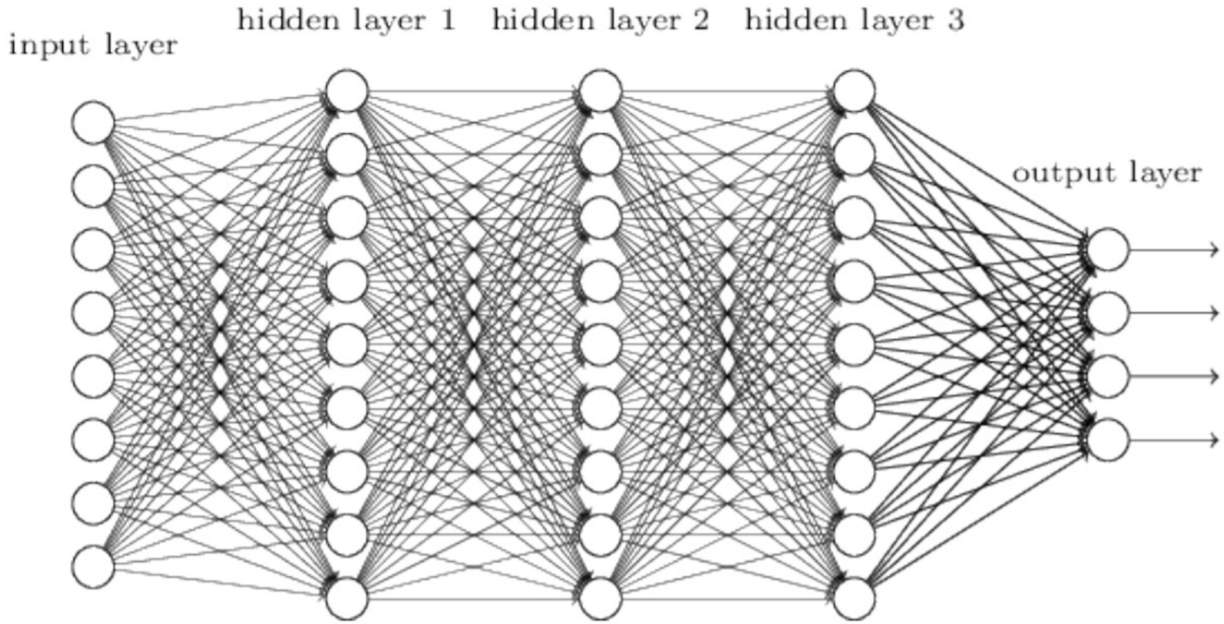
Outline

1. Deep learning architectures
    - CNNs
    - RNNs
    - Autoencoders
    - Word embeddings
    - Transformers
    - Generative models (e.g. GANs)
    - Graph Neural Networks
    - Ultra deep learning (ResNet)
    - Neural ODEs

- Project Specifics
    - Potential Topics
    - Format of proposal and report

Fully connected network

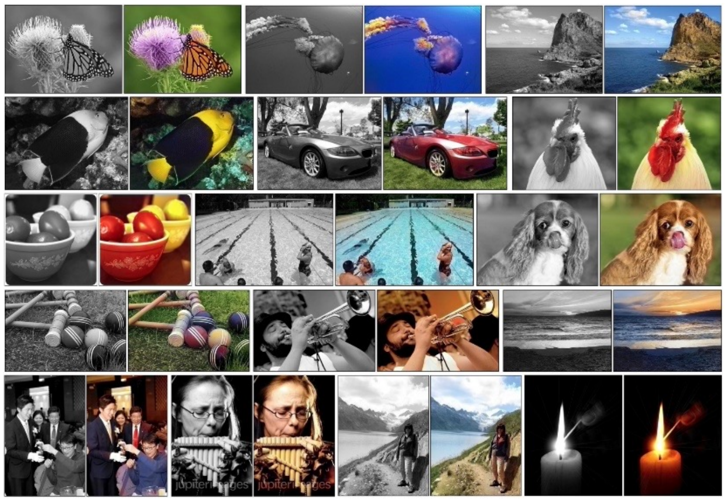
- Every feature interacts with every other feature
- Weight matrix at every level allowed to be dense

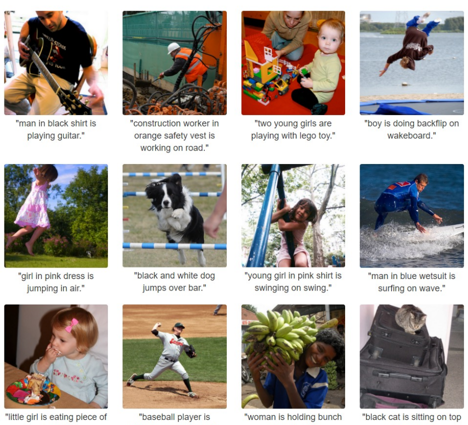


Michael Nielsen

Convolutional Neural Networks (CNNs)

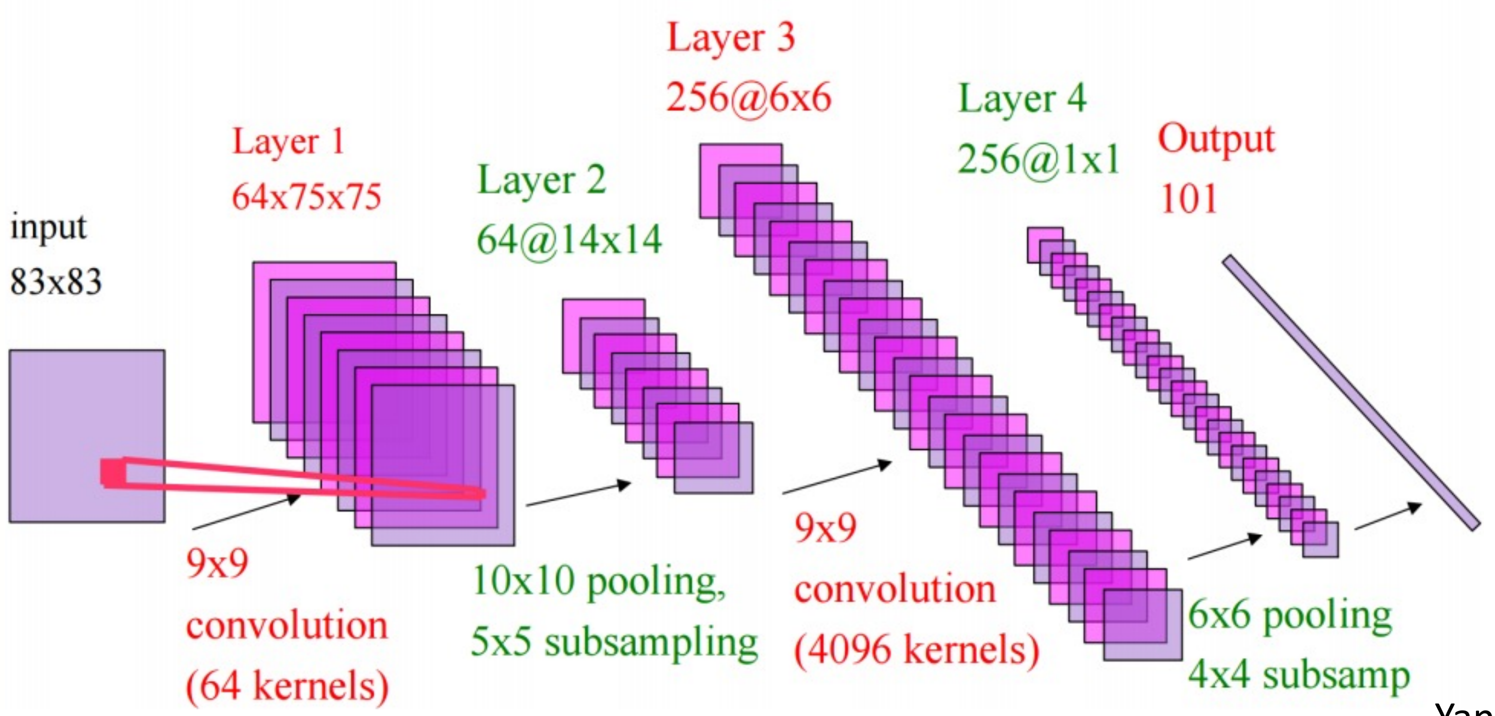
- Very successful in images





Convolutional Neural Networks (CNNs)

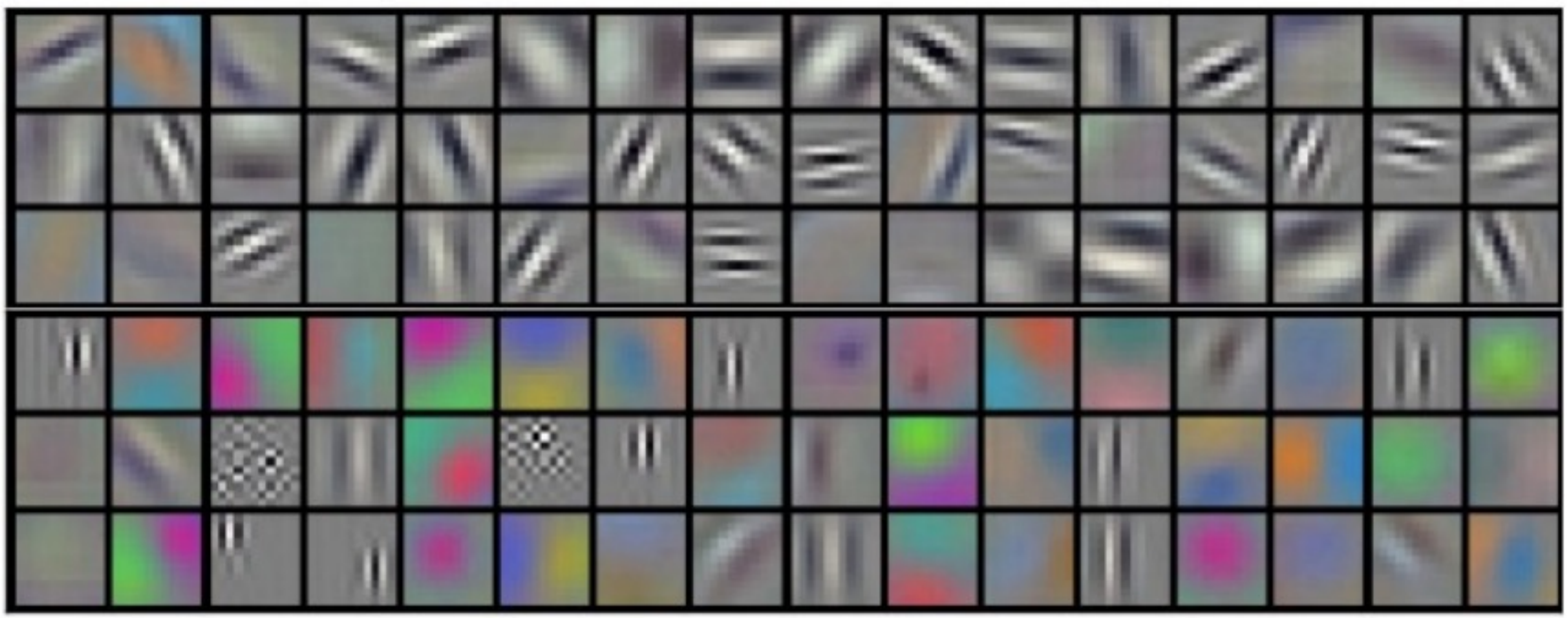
- Only pixels that are close to each other in the image interact with each other (convolution layer)
- Weight matrices are highly structured
- "Pooling" helps to simplify output of convolution layer



Yann LeCun

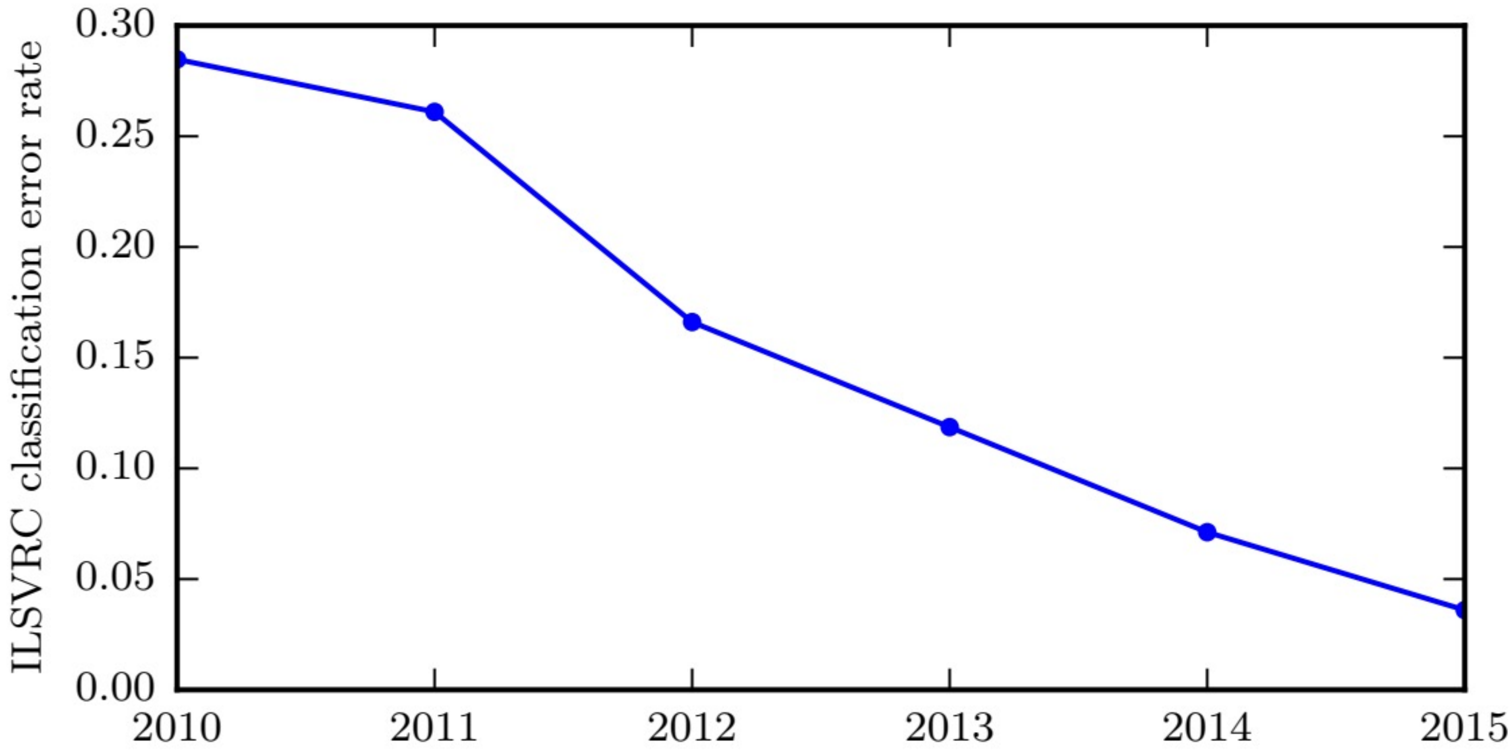
Convolutional Neural Networks (CNNs)

- Weights from the first layer tend to look like directional filters after training
  - Detects edges, color change, etc.



CS 231n, Karpathy

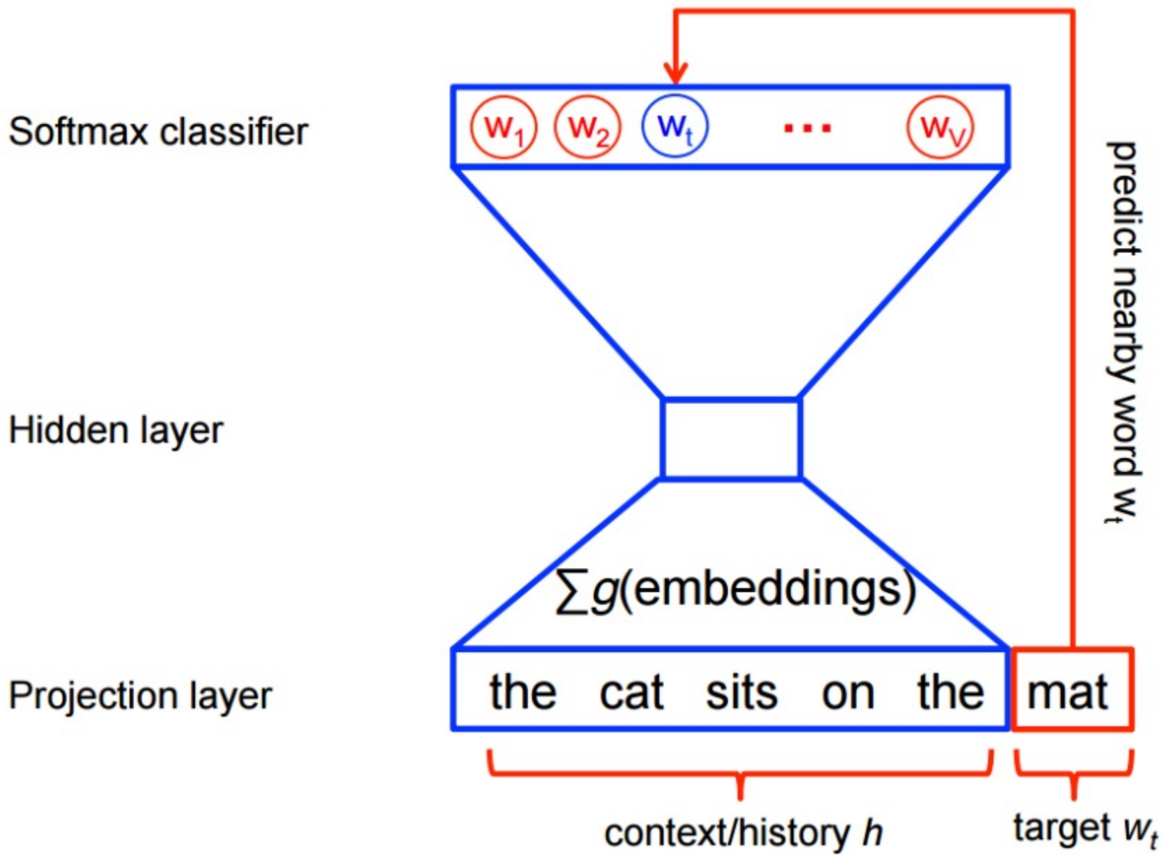
Convolutional Neural Networks (CNNs)



Goodfellow et al., 2016

Word2Vec

- Organization of words via neural networks
- Next word in a sentence can be predicted based on organization



Recurrent Neural Networks (RNNs)

- Useful when time is important

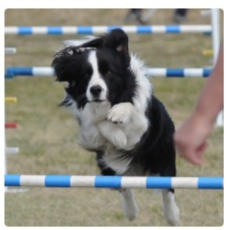

"man in black shirt is playing guitar."


"construction worker in orange safety vest is working on road."


"two young girls are playing with lego toy."

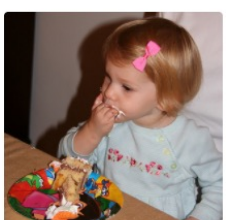

"boy is doing backflip on wakeboard."


"girl in pink dress is jumping in air."


"black and white dog jumps over bar."

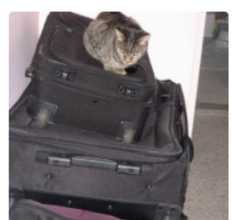

"young girl in pink shirt is swinging on swing."


"man in blue wetsuit is surfing on wave."


"little girl is eating piece of cake."


"baseball player is throwing ball in game."

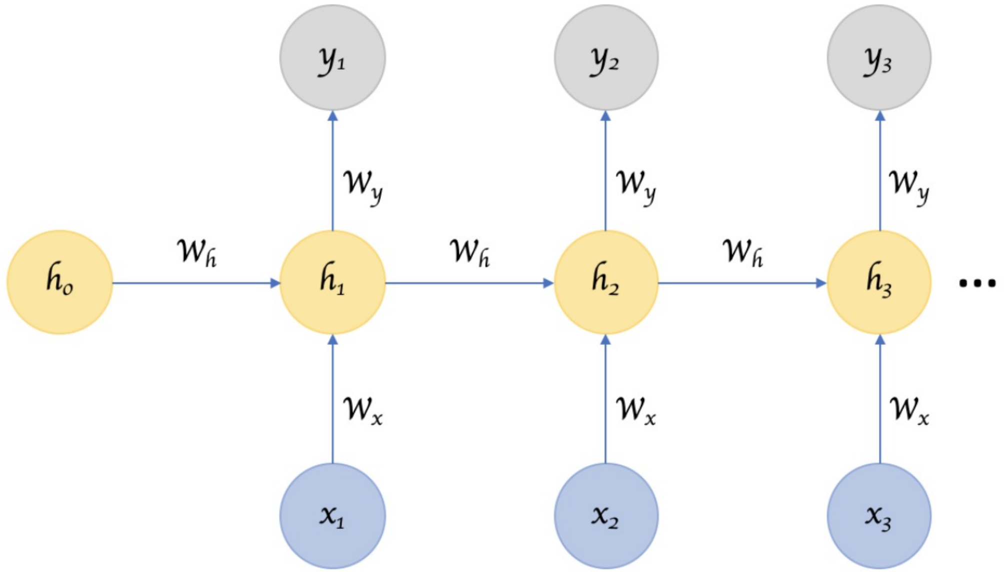

"woman is holding bunch of bananas."


"black cat is sitting on top of suitcase."

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
```

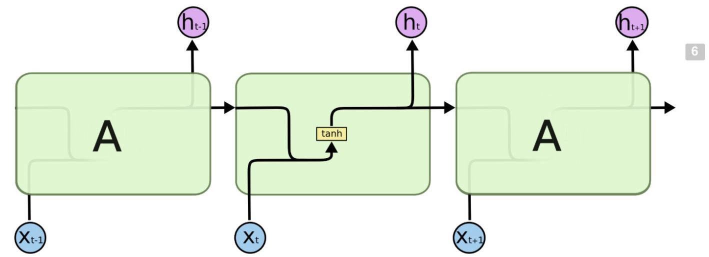
Mörk → Dark

Recurrent Neural Networks

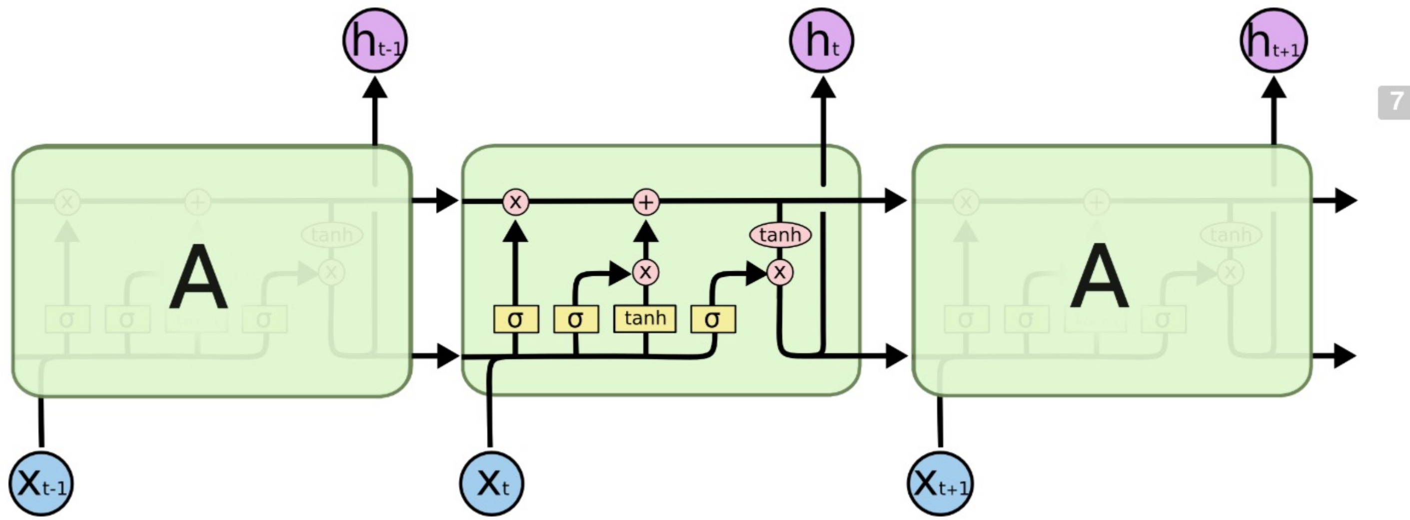


Takes arbitrarily sized inputs and "remember" a hidden state of information

LSTMs



ht-1
ht
ht+1
A
tanh
A
Xt-1
Xt
Xt+1
6

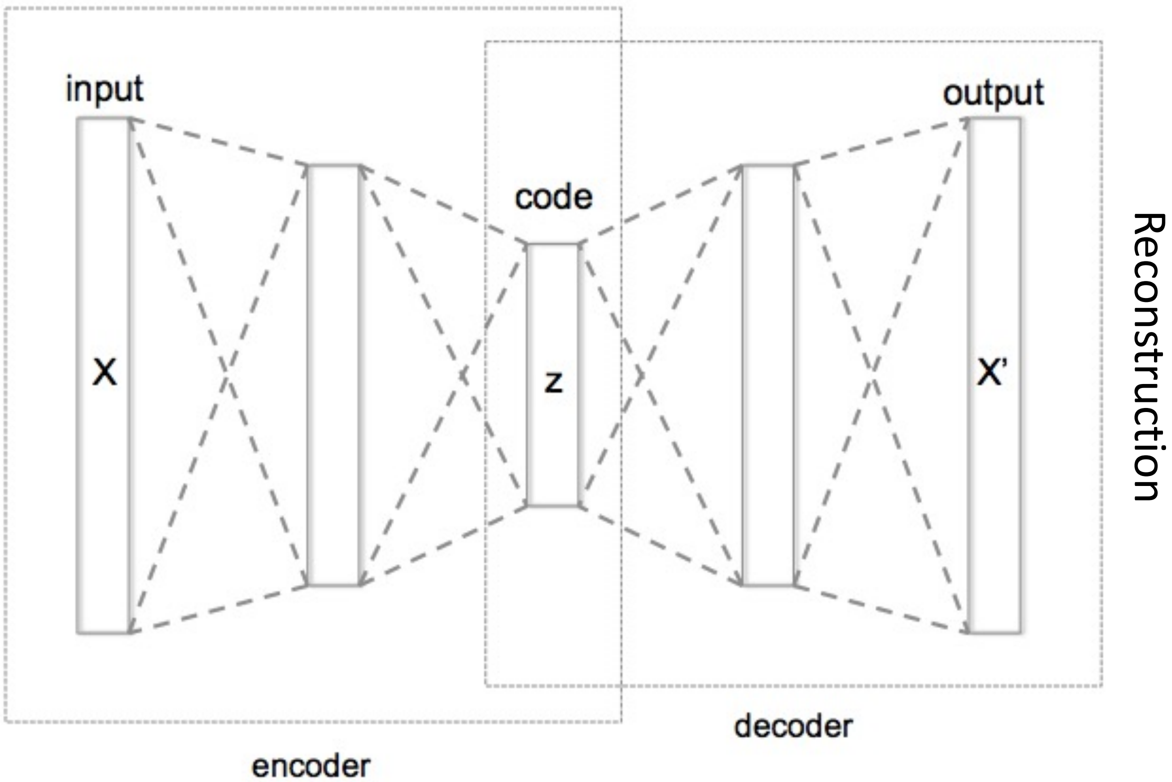


ht-1
ht
ht+1
7
A
σ
σ
tanh
σ
tanh
A
Xt-1
Xt
Xt+1

Autoencoders

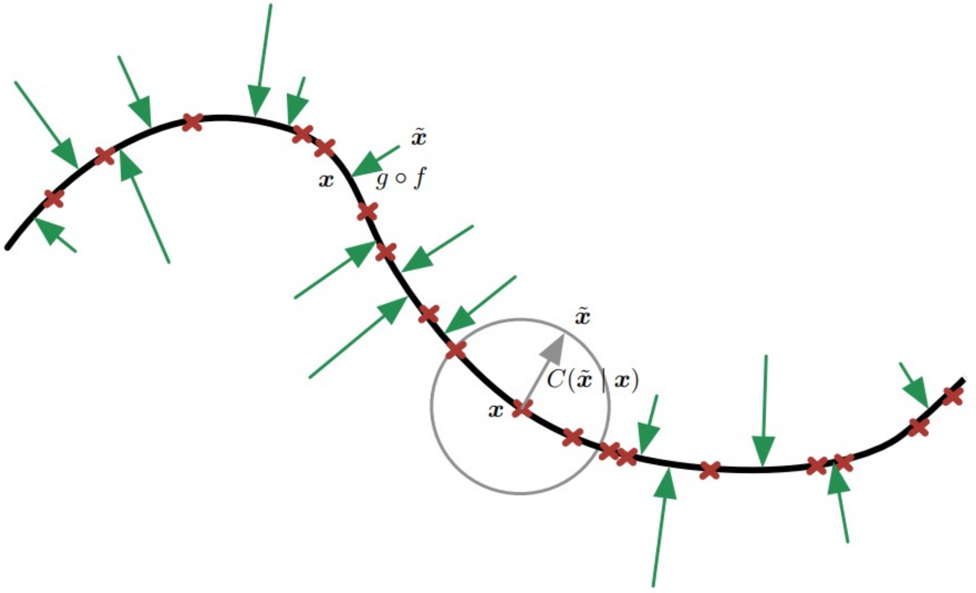
- Attempts to compress the data and then reconstruct the input

“Bottleneck” layer



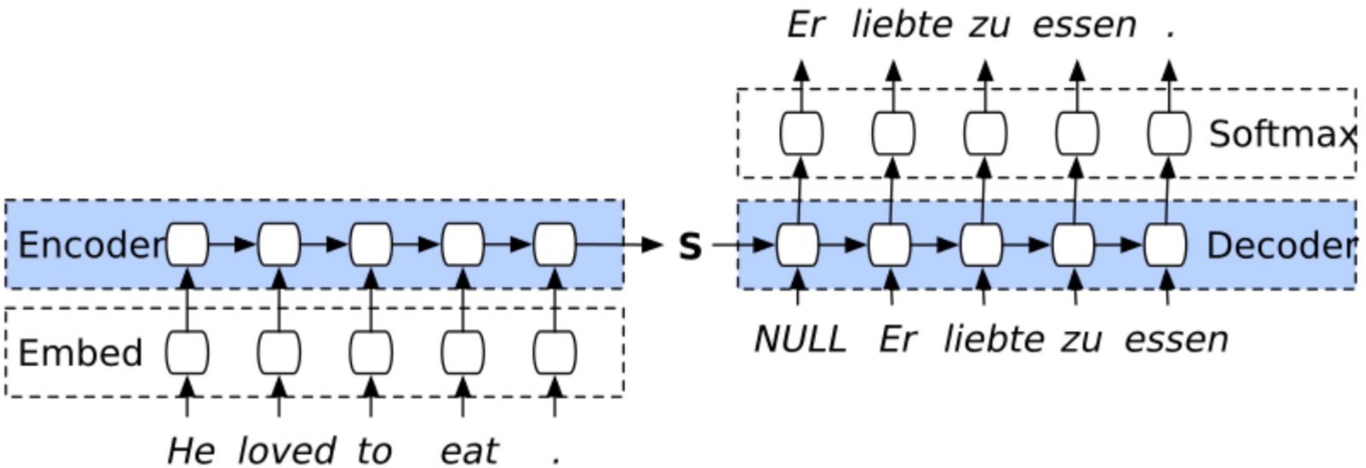
Autoencoder Applications

- Pretraining
- Dimensionality reduction
  - Information retrieval
  - Denoising
  - Data compression
- Generative modeling
- Batch correction

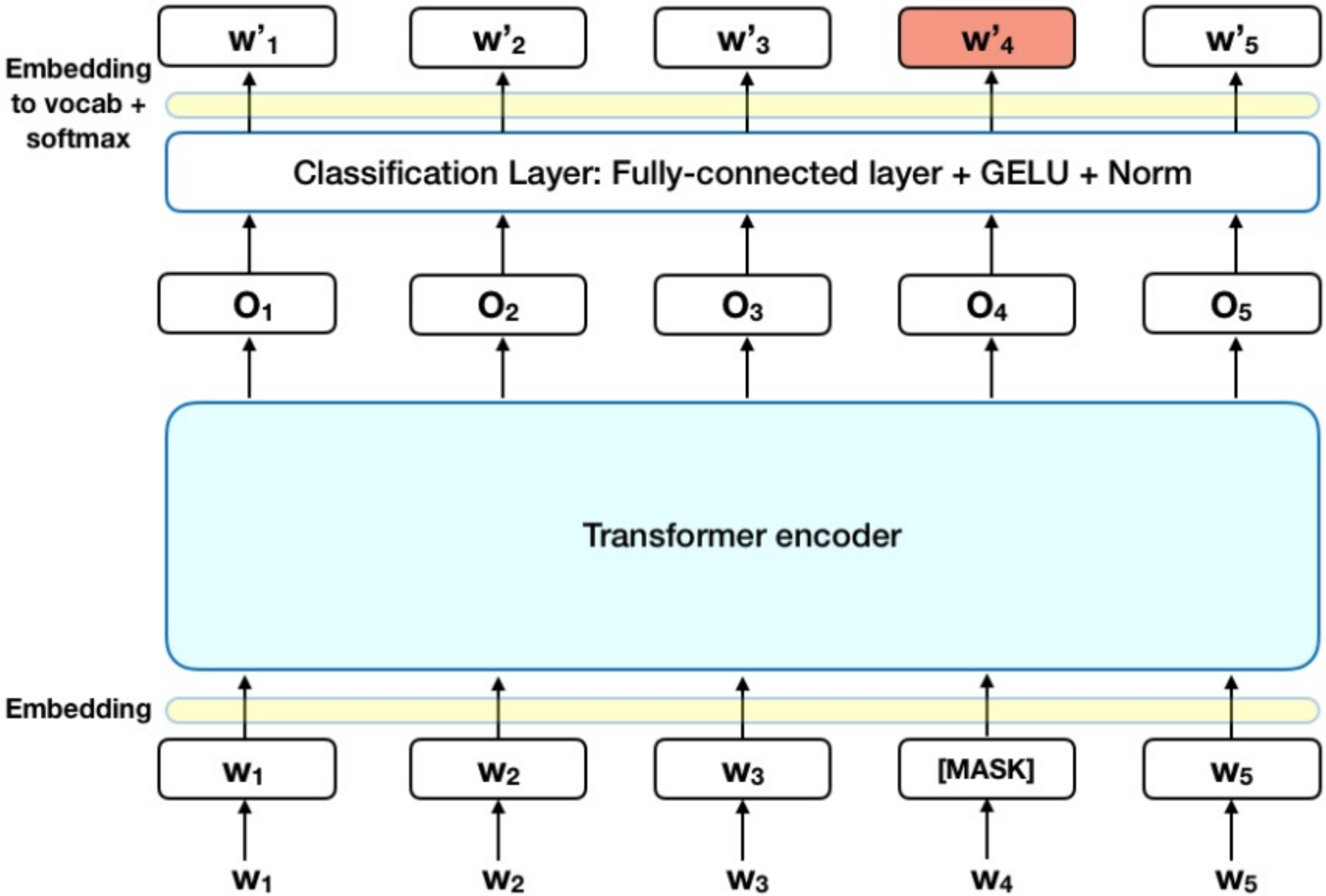


Goodfellow et al., 2016

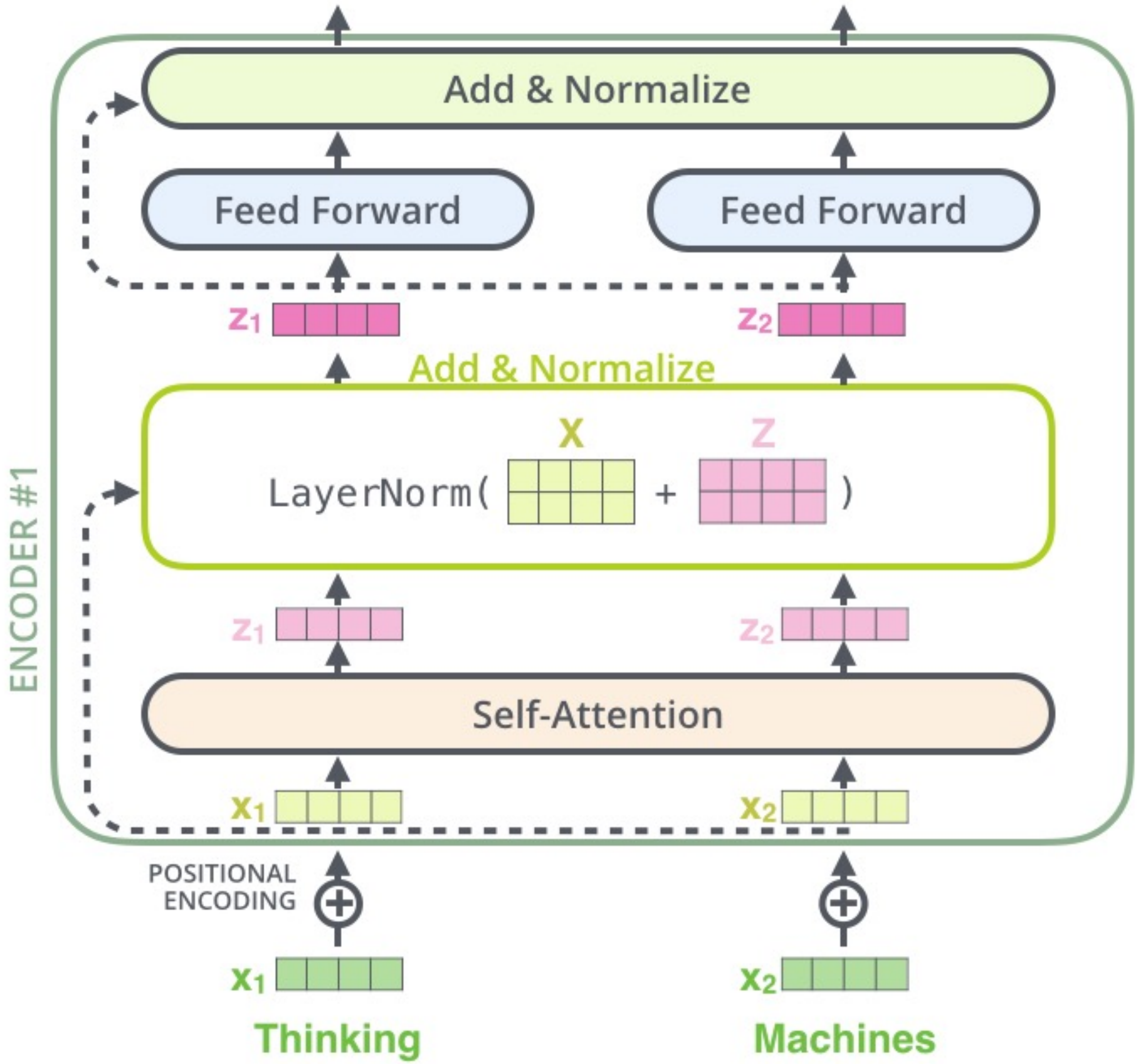
Sequence-to-sequence model



Transformer



Attention

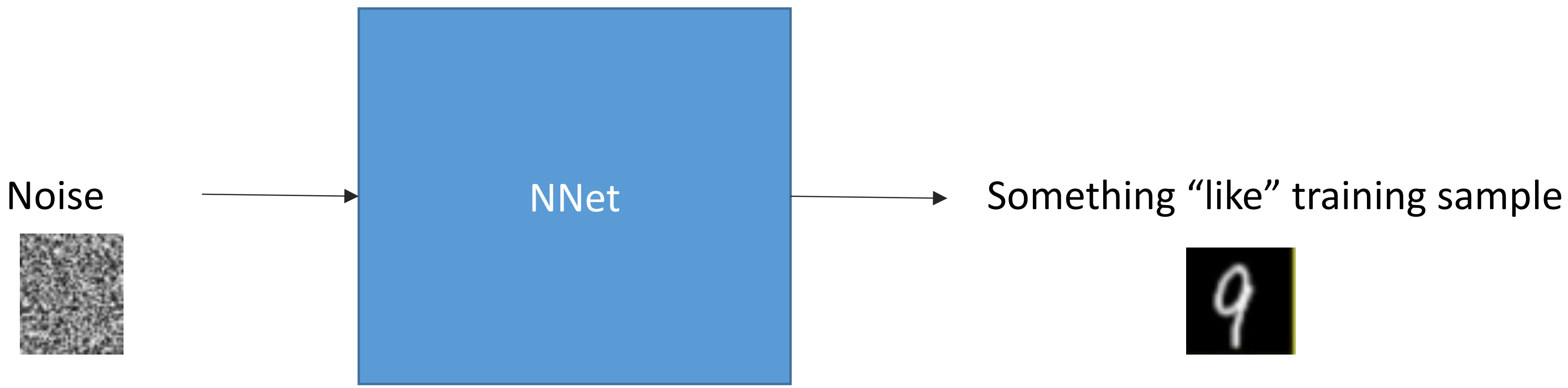


Add & Normalize
Feed Forward
Feed Forward
z_1
z_2
Add & Normalize
X
Z
LayerNorm(+)
z_1
z_2
Self-Attention
x_1
x_2
ENCODER #1
POSITIONAL ENCODING
x_1
x_2
Thinking
Machines

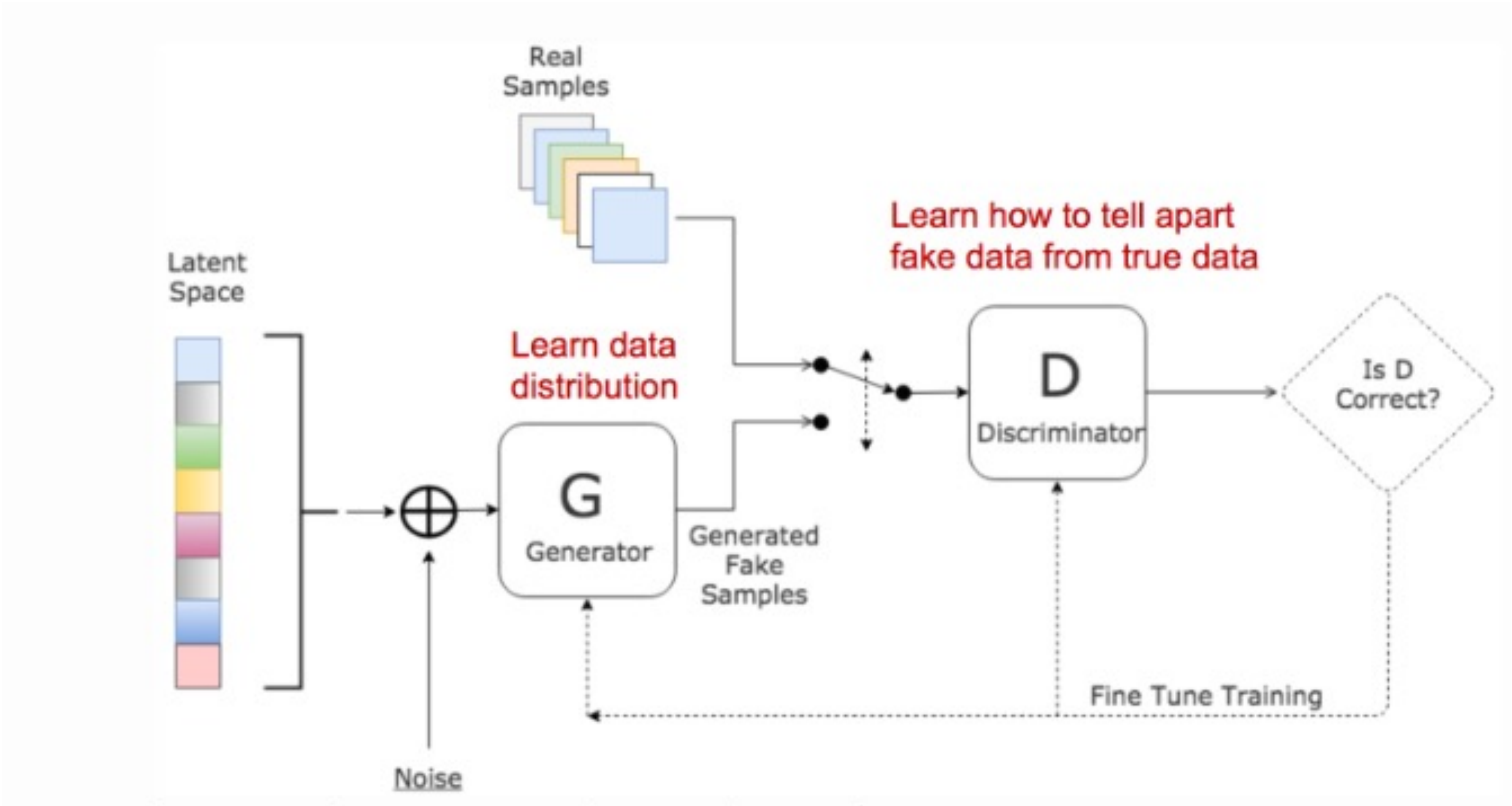
Generative Models

- Create a map from random noise into distribution of training data to generate samples
- Generative Adversarial Net (GAN)
  - Generative model is pitted against an discriminative model that determines whether a sample is from the model or the data
  - Improves both generation and discrimination

Generative Models



Generative Adversarial Network

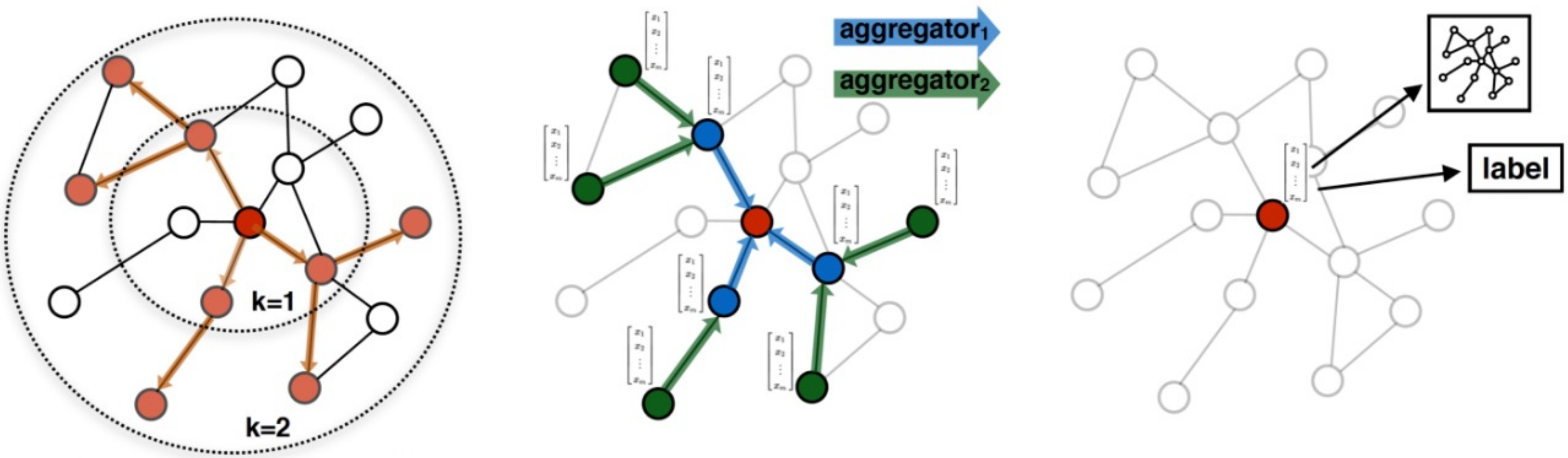


Goodfellow et al. 2014

Generative Models

Goodfellow

Graph Neural Network



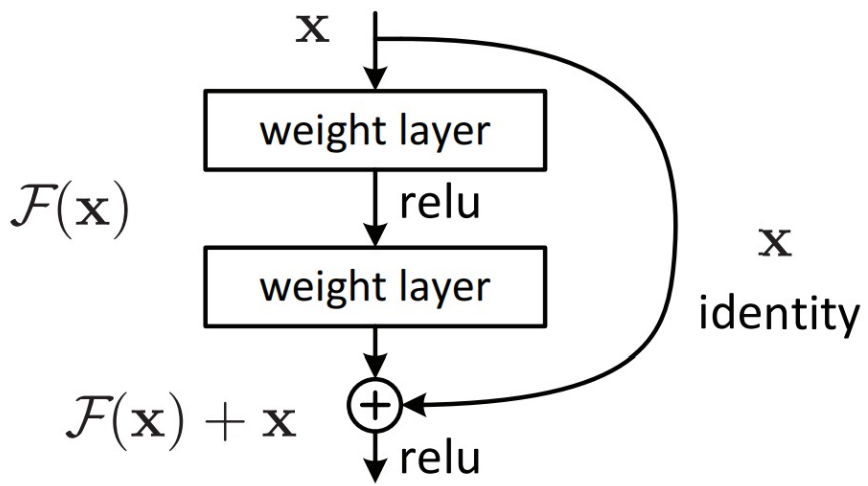
1. Sample neighborhood

2. Aggregate feature information from neighbors

3. Predict graph context and label using aggregated information

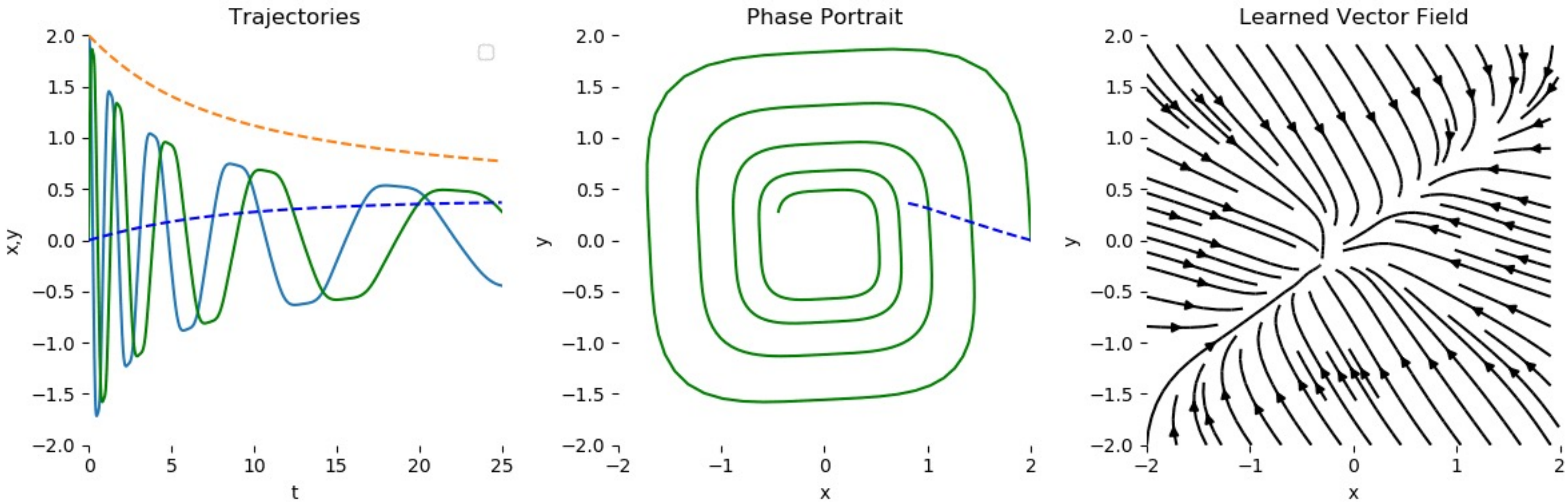
Ultra Deep Learning (e.g. ResNet)

- Very deep neural nets are difficult to train
  - Accuracy can degrade with deeper networks
- ResNet developed a framework to address this degradation
  - Successfully trained a 152 layer network
  - Won the ILSVRC 2015 image classification task



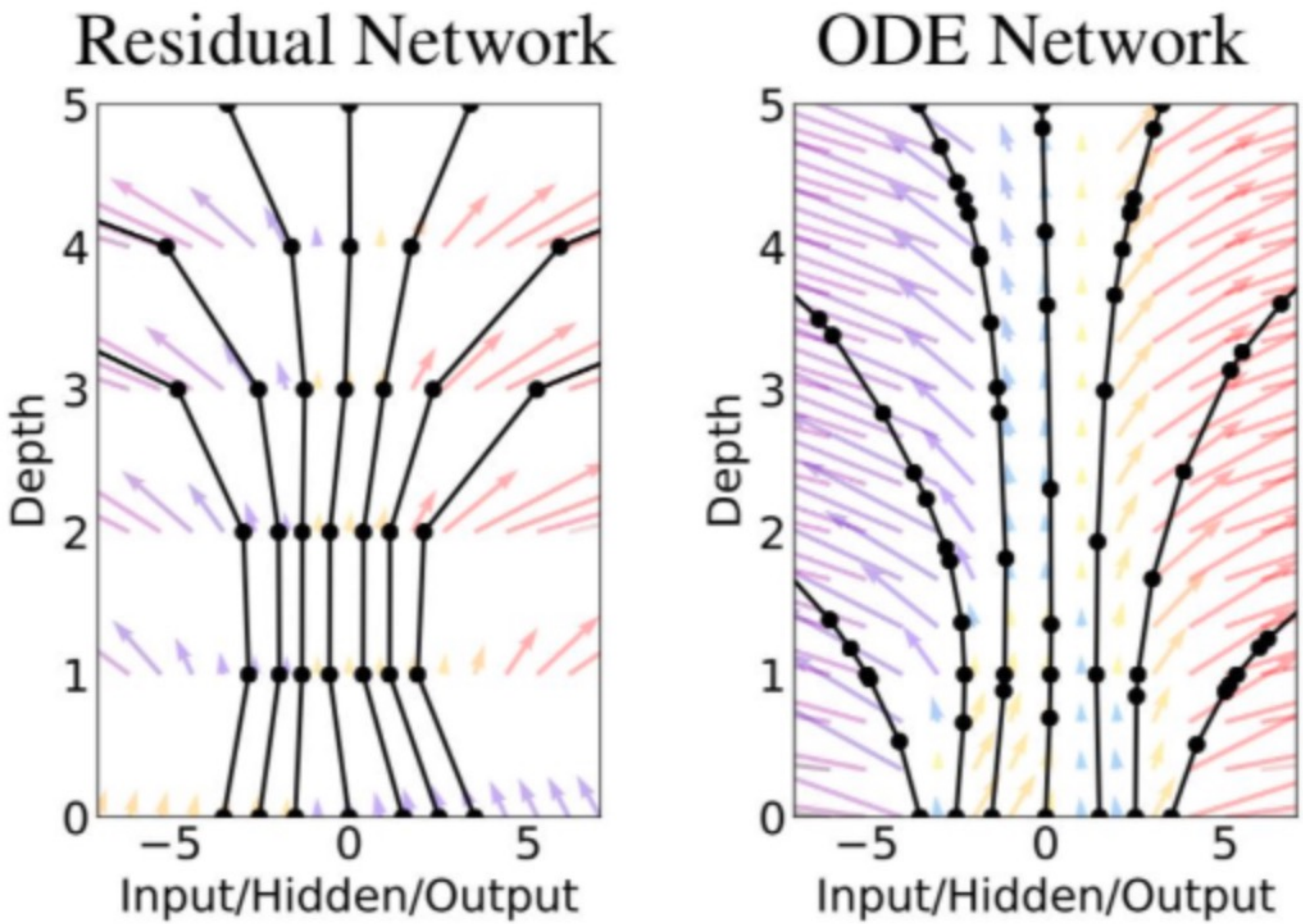
arxiv.org/abs/1512.03385

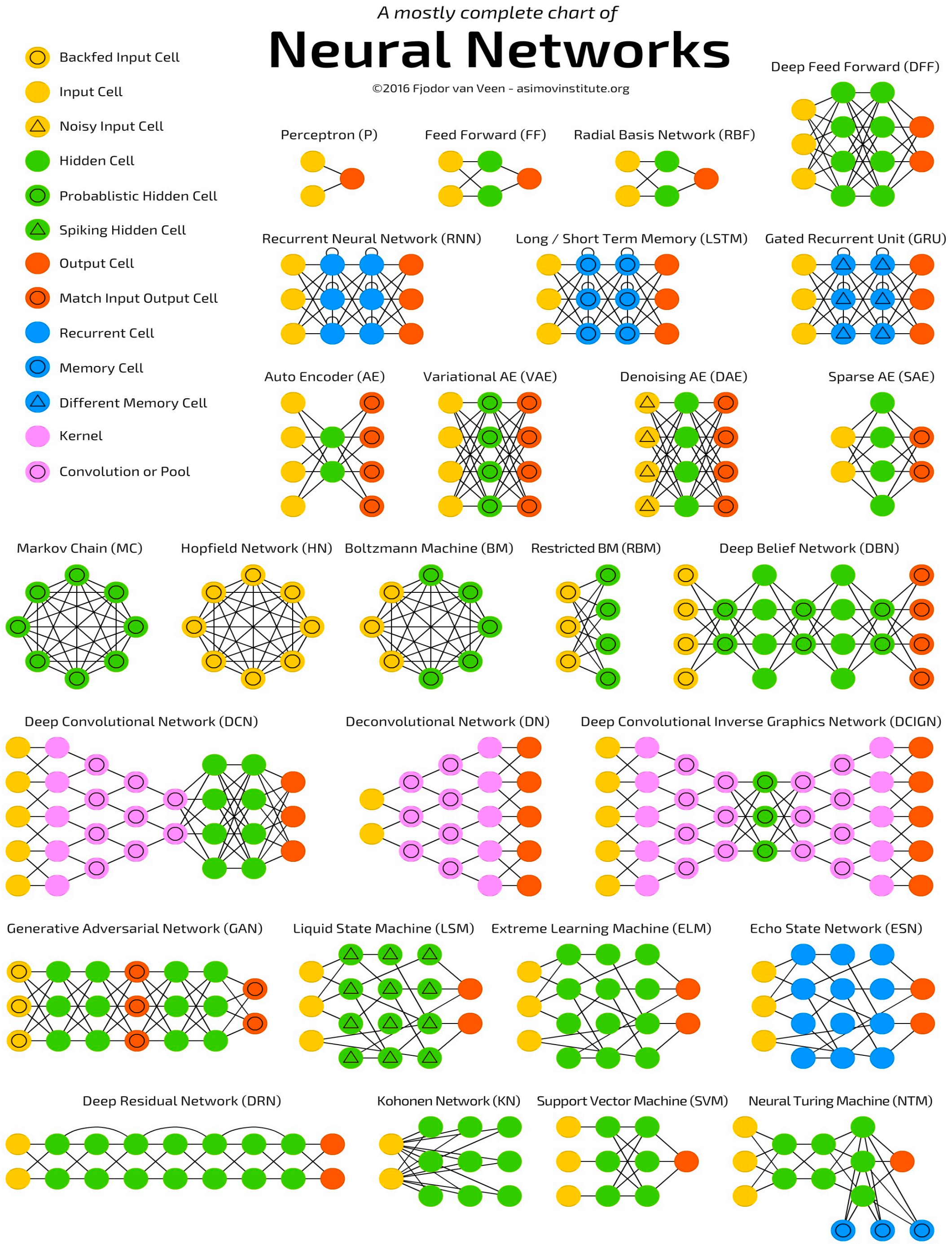
Neural ODE



Trajectories
x,y
t
Phase Portrait
y
x
Learned Vector Field
y
x

Infinite Depth




A mostly complete chart of
Neural Networks
©2016 Fjodor van Veen - asimovinstitute.org
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)

Final Project

- To be done in groups of 3-4
- Progress report in the middle of the semester
- Report and Code turned the last week of classes
- Graduate students turn in a longer report explaining each member’s specialization in the group (must have a specialization)
- Undergraduate students can turn in the project without this

Applications

- To create a deep learning framework for processing a particular type of dataset (genomic data, health data, physics simulation data).
  - Ex: Graph Neural Network for looking at spatial transcriptomic data
  - Recurrent neural networks for classifying or generating music
  - Proteomic or chemical data analysis
- Deepsea: https://www.nature.com/articles/nmeth.3547
- Deepbind: https://www.nature.com/articles/nbt.3300

Architecture and Regularization

- To propose a new architectural feature or regularization to improve network performance
  - archetypal regularization
  - Consciousness prior
  - Attention layers
  - New activation function
- Deepmanreg: https://www.nature.com/articles/s43588-021-00185-x
- https://arxiv.org/abs/2006.12253

Generalization

- To examine minimal conditions for generalization (level of overparameterization),
  - Would a network without dimensionality reduction generalize?
  - Would a 2-level network generalize better or worse than an n-level network
  - Can a certain architecture generalize from just one example?
- Circumstances in which networks generalize poorly
  - Anomalies
  - Adversarial noise
  - Polynomially many neurons in the input size (with SGD)
- Inductive bias and how it changes with
  - Initialization
  - Activations
  - Depth

Examples

- Why do deep networks generalize so well?
  - Coherent gradients: https://arxiv.org/pdf/2002.10657.pdf
- Can we find the global minima in some cases?
  - Neural tangent kernel (infinitely wide networks) https://arxiv.org/abs/1806.07572

Loss Landscapes

- Effect the loss landscape has on training
  - Regularizations that make the loss landscape smoother
  - Assessing the quality of the reached minima by looking at the surrounding loss landscape
- Effect of dropout: http://proceedings.mlr.press/v80/mianjy18b.html
- Visualizing the loss landscape: https://arxiv.org/abs/1712.09913

Training Procedures

- Examination of a new training procedure
  - Second derivatives
  - Momentum
  - Equilibrium training

- https://iopscience.iop.org/article/10.1088/1757-899X/495/1/012003/pdf

Paradigms

- Exploration paradigms for neural networks
  - Quantum neural networks
  - Neural ODEs and infinite depth networks
  - Binarized or logical networks

- Neural ODE:https://arxiv.org/abs/1806.07366

- Logical Neural Networks: https://arxiv.org/pdf/2006.13155.pdf

Simulations/Algorithms

- Implementing an algorithm using a neural networks
  - Neural networks that compute diffusion maps, tSNE
  - Neural networks that compute earth mover's distance
  - MCMC simulators
  - Deep reinforcement learning

- Alphafold: https://www.nature.com/articles/s41586-021-03819-2
- Fluid dynamics: https://www.pnas.org/content/118/21/e2101784118

Project Proposal

- 1. What is the problem that you will be investigating? Why is it interesting?
- 2. What are the challenges of this project?
- 3. What dataset are you using?
- 4. What Deep learning approach will you use/develop?
- 5. How will you evaluate your results?
  - Qualitatively, what kind of results do you expect (e.g. plots or figures)?
  - Quantitatively, what kind of analysis will you use to evaluate and/or compare your results (e.g. what performance metrics or statistical tests)?

Project Report (4-5 pages)

- Introduction/Motivation - What is the problem you are solving? What need does it solve?
- Background/Related Work - What has been done before? What frameworks have been proposed? What are the limitations of the previous work?
- Model: What is your neural network model/architectural feature etc? What interesting properties does it have? <Model Schematic>
- Empirical/Theoretical Results: Validation on relevant datasets, evidence of model generalization, proofs of algorithms etc. <Results Tables>
  - This could be a where grads talk about what each member focused on
- Conclusions/Future work: Where can this work go in the future?

Example from 2019: Style Transfer

- This project was centered around the problem of Style Transfer
  - Images of a Tabby or Egyptian cat and performing a transformation on the images to yield images of the opposite cat breed.
  - The methods used to achieve this style transfer were several Generative Adversarial Network architectures, specifically DiscoGAN, CycleGAN, and UNIT.
  - While none of these networks was ultimately able to provide excellent performance for the problem at hand, they each ran into distinct issues due to their different architectures and designs.

Example from 2019: Music Generation

- Music generation poses an interesting problem in the world of modern deep learning. Deep learning has progressed to a point where the domain of art, previously believed a solely creativity- driven human pursuit, has been infringed upon by computers. Methods like style transfer and DeepDream have shown deep networks ability to generate novel images. [1][2] Further, Google's Project Magenta group has shown a networks ability to create meaningful music from midi data. [3] However, the midi format can often ignore the structure of a musical piece, and thus a network has more diculty replicating the signal behind it. By the use of two Seq2Seq networks and a SeqGAN, we have created a network capable of algorithmic music generation that is designed around the structure of music.