Self-Supervised Learning

Danna Gurari
University of Colorado Boulder
Spring 2024
Review

- Last lecture on Practical Systems-Level Development Challenges:
  - Motivation
  - Data curation
  - Model maintenance
  - TAs’ experiences

- Assignments (Canvas)
  - Final project outlines due in 1 week

- Questions?
Today’s Topics

• Overview of self-supervised learning

• Generative-based methods

• Generative adversarial networks

• Context-based methods
Today’s Topics

• Overview of self-supervised learning

• Generative-based methods

• Generative adversarial networks

• Context-based methods
Goal: Create Generalizable Features

Key observation: features from a pretrained network can be useful for other datasets/tasks

Focus: Avoid Relying on Large **Labeled** Datasets

- **Expensive**
- **Relatively Slow to Build Dataset**
Self-Supervised Learning: Data Gives Supervision

- Relatively Cheap
- Can Collect Data Fast

https://lovevery.com/community/blog/child-development/the-surprising-learning-power-of-a-household-mirror/

https://www.rockettes.com/blog/how-to-use-the-mirror-in-dance-class/
Self-Supervised Learning: Data Gives Supervision

Approach: create features that are useful for other datasets/tasks

Self-Supervised Learning Methods Already Covered in This Course (Many NLP Methods)

Character prediction with RNNs

Word embeddings
(e.g., word2vec; predict nearby word for given word)

Transformers
(e.g., BERT and LXMERT with masking)


https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b

Today’s Topics

• Overview of self-supervised learning

• Generative-based methods

• Generative adversarial networks

• Context-based methods
Generative-based Methods

- **Autoencoder**: predict self
- **Colorization**: convert grayscale to color
- **Video prediction**: predict future frames
Generative-based Methods

- **Autoencoder**: predict self
- **Colorization**: convert grayscale to color
- **Video prediction**: predict future frames
Image Autoencoder Architecture

- Learn to copy the input to the output

Figure Credit: https://lazyprogrammer.me/a-tutorial-on-autoencoders/
Image Autoencoder Architecture

- Consists of two parts:
  - **Encoder**: compresses inputs to an internal representation
  - **Decoder**: tries to reconstruct the input from the internal representation

Figure Credit: https://www.datacamp.com/community/tutorials/autoencoder-keras-tutorial
Image Autoencoder Architecture

• Given this input 620 x 426 image (264,120 pixels):

  • What would a perfect autoencoder predict?
    • Itself
  • What number of nodes are in the final layer?
    • 264,120

Figure Credit: https://lazyprogrammer.me/a-tutorial-on-autoencoders/
Image Autoencoders

• Intuition: which number sequence is easier to remember?
  • A: 30, 27, 22, 11, 6, 8, 7, 2
  • B: 30, 15, 46, 23, 70, 35, 106, 53, 160, 80, 40, 20, 10, 5

• B: need learn only two rules
  • If even, divide by 2
  • If odd, multiply by 3 and add 1

Figure Credit: https://lazyprogrammer.me/a-tutorial-on-autoencoders/
Image Autoencoder Training

Repeat until stopping criterion met:

1. **Forward pass**: propagate training data through network to make prediction
2. **Backward pass**: using predicted output, calculate error gradients backward
3. Update each weight using calculated gradients
Image Autoencoder Features

• e.g., training data:
  • 1 image taken from 10 million YouTube videos
  • Each image is in color and 200x200 pixels

• What features do you think it learned?

Quoc V. Le et al., Building High-level Features Using Large Scale Unsupervised Learning; ICML 2013.
Image Autoencoder Features

• e.g., features learned include:

human face  cat face  human body

Quoc V. Le et al., Building High-level Features Using Large Scale Unsupervised Learning; ICML 2013.
Video Autoencoder

• Train RNN to predict input sequence

Srivastava et al., Unsupervised Learning of Video Representations using LSTMs; ICML 2015.
Generative-based Methods

- Autoencoder: predict self
- **Colorization**: convert grayscale to color
- Video prediction: predict future frames
Colorization: *Plausible* Coloring Results

Colorization: *Plausible* Coloring Results

Figure Sources: https://www.flickr.com/photos/applesnpearsau/12197380673/in/photostream/; https://commons.wikimedia.org/wiki/File:JACQUES_VILET_-_1982,_Les_Fruits_du_Jardin.jpg
Image Colorization Architecture

Image Colorization Architecture: CIE Lab Color

$L$ indicates grayscale information whereas $a$ and $b$ represent colors.

Figure source: https://www.researchgate.net/figure/The-cubical-CIE-Lab-color-space_fig3_23789543
Create image by combining predicted $a$ and $b$ channels with the $L$ channel.
Image Colorization Architecture

Grayscale image: $L$ channel
$$X \in \mathbb{R}^{H \times W \times 1}$$

$X$

Color information: $ab$ channels
$$\hat{Y} \in \mathbb{R}^{H \times W \times 2}$$

$ab$

Figure source: http://videolectures.net/eccv2016_zhang_image_colorization/
Image Colorization Architecture

Grayscale image: $L$ channel

$X \in \mathbb{R}^{H \times W \times 1}$

$\mathcal{F}$

Concatenate $(L, ab)$

$(X, \hat{Y})$

Figure source: http://videolectures.net/eccv2016_zhang_image_colorization/
Image Colorization Training

For 1.3 million ImageNet images, repeat until stopping criterion met:

1. **Forward pass**: propagate training data through network to make prediction
2. **Backward pass**: using predicted output, calculate error gradients backward
3. Update each weight using calculated gradients

Image Colorization Features

Task requires understanding an image at the pixel and semantic-level

Figure source: http://richzhang.github.io/colorization/
Generative-based Methods

• Autoencoder: predict self

• Colorization: convert grayscale to color

• **Video prediction**: predict future frames
Video Prediction

• Train RNN to predict future frames
• Limitations: identifying new objects and background as a camera moves

What type of features might be learned?

Srivastava et al., Unsupervised Learning of Video Representations using LSTMs; ICML 2015.
Generative-based Methods

- Autoencoder: predict self
- Colorization: convert grayscale to color
- Video prediction: predict future frames
Today’s Topics

• Overview of self-supervised learning

• Generative-based methods

• Generative adversarial networks

• Context-based methods
Generative adversarial networks

• Generative adversarial networks (GANs)

• Context encoder
Generative adversarial networks

- Generative adversarial networks (GANs)
- Context encoder
GAN: Basic Architecture

- **Random Input Vector**
- **Generator Model**
  - e.g., upsample noise to image of desired size
- **Discriminator Model**
  - classify which examples are real

Consists of two models that compete against each other

https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/
The two models are iteratively trained separately:
- Train discriminator using fake and real images
- Train generator using just fake images and penalize it when the discriminator recognizes images are fake
GAN: Discriminator Loss Function

Discriminator tries to minimize classification error

\[ J(D) = -\frac{1}{2} \mathbb{E}_{x \sim p_{\text{data}}} \log D(x) - \frac{1}{2} \mathbb{E}_{z} \log (1 - D(G(z))) \]

Discriminator wants a value of 1 for real images
Discriminator wants a value of 0 for fake images

Real image
Input noise

GAN: Generator Loss Function

Generator tries to maximize classification error

\[ J^{(G)} = - J^{(D)} \]

Want the discriminator to mistakenly arrive at a value of 1 for fake images

\[ J^{(G)} = - \frac{1}{2} \mathbb{E}_{z} \log D(G(z)) \]

Input noise

DGANs: GANs that Use Convolutional Layers

Bedrooms generated by observing over 3M bedroom images

DGANs: GANs that Use Convolutional Layers

What objects does it learn to generate?

DGANs: GANs that Use Convolutional Layers

What objects may it not have learned to generate?

DGANs: GANs that Use Convolutional Layers

Faces generated by observing over 3M images of 10K people

DGANs: GANs that Use Convolutional Layers

What does it generate poorly or not all?

Generative adversarial networks

• Generative adversarial networks (GANs)

• Context encoder
Task: Hole Filling

• What might fit into this hole?

• Many items may plausibly fit into the hole:

• Challenge: have up to 1 known ground truth region per hole
Architecture

Follows AlexNet architecture

Encoded context

Transposed convolutions

Patch for hole

Figure source: https://medium.com/knowledge-engineering-seminar/context-encoder-image-inpainting-using-gan-ccd6a1ea5fb7
Training: Loss Functions (\( \mathcal{L} = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{rec}\mathcal{L}_{rec} \))

Figure source: https://medium.com/knowledge-engineering-seminar/context-encoder-image-inpainting-using-gan-ccd6a1ea5fb7
Training: Reconstruction Loss (i.e., Self-Supervised Learning Approach)

Learns to fit into the context by computing the L2 loss to compare the original patch content (P) to the predicted patch content created by the model when given the image with hole (CE(X')).

\[
L_{rec} = \| P - CE(X') \|^2_2
\]

Pathak et al., Context encoders: Feature learning by inpainting; CVPR 2016
Training: Reconstruction Loss (i.e., Self-Supervised Learning Approach)

Why might training with this loss function alone lead to blurry results?
- It averages the multiple plausible inpaintings for a hole

Pathak et al., Context encoders: Feature learning by inpainting; CVPR 2016
Training: Loss Functions (\( \mathcal{L} = \lambda_{adv} \mathcal{L}_{adv} + \lambda_{rec} \mathcal{L}_{rec} \))

Remedy is to introduce an adversarial loss:

Figure source: https://medium.com/knowledge-engineering-seminar/context-encoder-image-inpainting-using-gan-ccd6a1ea5fb7
Training: Datasets

Pathak et al., Context encoders: Feature learning by inpainting; CVPR 2016

Training completed on ImageNet (all 1.2M and a 100K subset) for three hole types

(a) Central region  (b) Random block  (c) Random region
What type of features might be learned?

Pathak et al., Context encoders: Feature learning by inpainting; CVPR 2016
Today’s Topics

• Overview of self-supervised learning

• Generative-based methods

• Generative adversarial networks

• Context-based methods
Context-based Methods

- **Spatial context**: predict relative positions of image patches
- **Timing context**: predict relative positions of video frames
- **Similarity context**: clustering
Spatial Context: Predict Image Index Per Patch

What type of features might be learned?

Timing Context: Predict Order of Video Frames

What type of features might be learned?

Lee et al., Unsupervised Representation Learning by Sorting Sequences; ICCV 2017.
CNNS are trained to identify cluster assignments OR to recognize whether images belong to the same cluster

Raschka and Mirjalili; Python Machine Learning
Clustering

Create groupings so entities in a group will be similar to each other and different from the entities in other groups.

Raschka and Mirjalili; Python Machine Learning
Clustering: Key Questions

• How many data clusters to create?
• What “algorithm” to use to partition the data?
Clustering: How Many Clusters to Create?

Number of clusters can be ambiguous.

Slide adapted from: https://www-users.cs.umn.edu/~kumar001/dmbook/slides/chap7_basic_cluster_analysis.pdf
Create groupings so entities in a group will be similar to each other and different from the entities in other groups.

What type of features might be learned?

Raschka and Mirjalili; Python Machine Learning
Today’s Topics

• Overview of self-supervised learning

• Generative-based methods

• Generative adversarial networks

• Context-based methods
The End