Current State of Unsupervised Deep Learning

William Falcon, PhD Student

facebook Artificial Intelligence

NYU

AGENDA

AGENDA

Unsupervised vs self-supervised vs supervised learning

Why we don't like supervised learning

Cost of supervised learning

Theoretical approaches to unsupervised learning

Current State-of-the-art

Closing thoughts

Unsupervised vs Supervised vs Self-supervised learning

Label this datapoint

Cutest thing ever

Dog

Dancing dog

Pet in living room

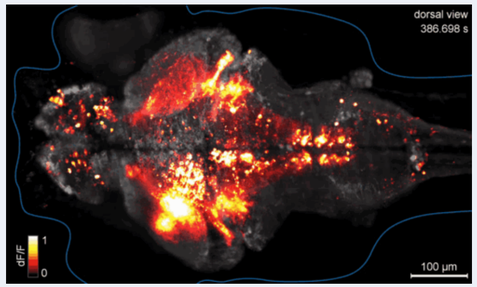
Pet on floor

Dog evolving

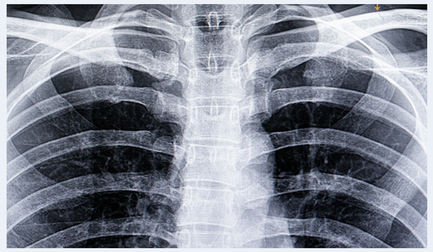
Humans are biased

Transfer Learning

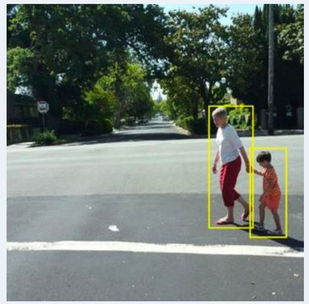
Transfer Learning



Neuroscience

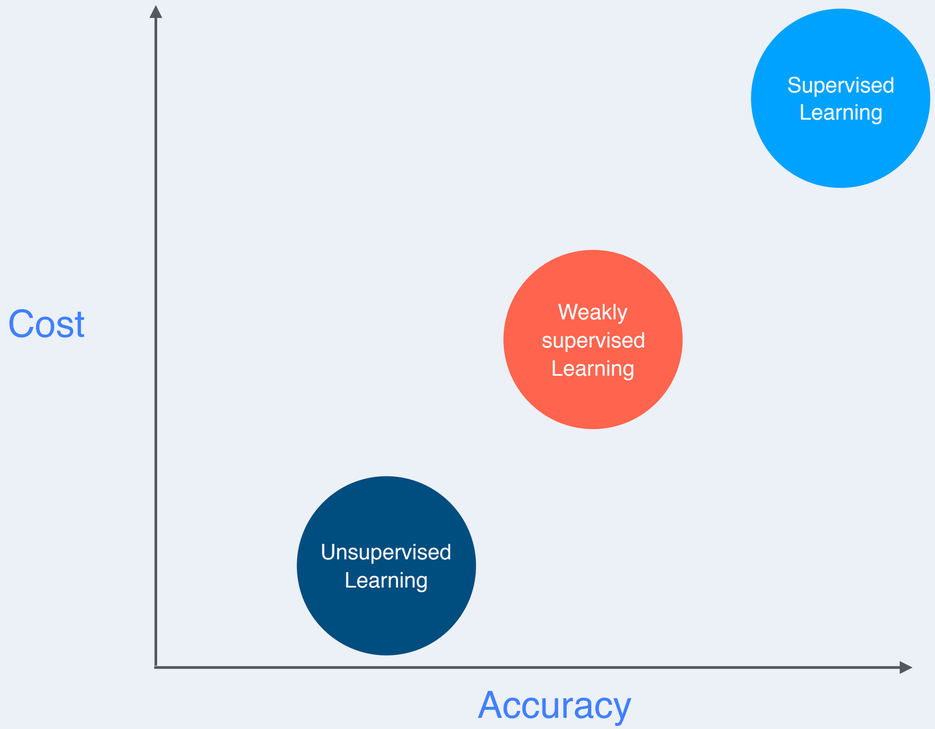


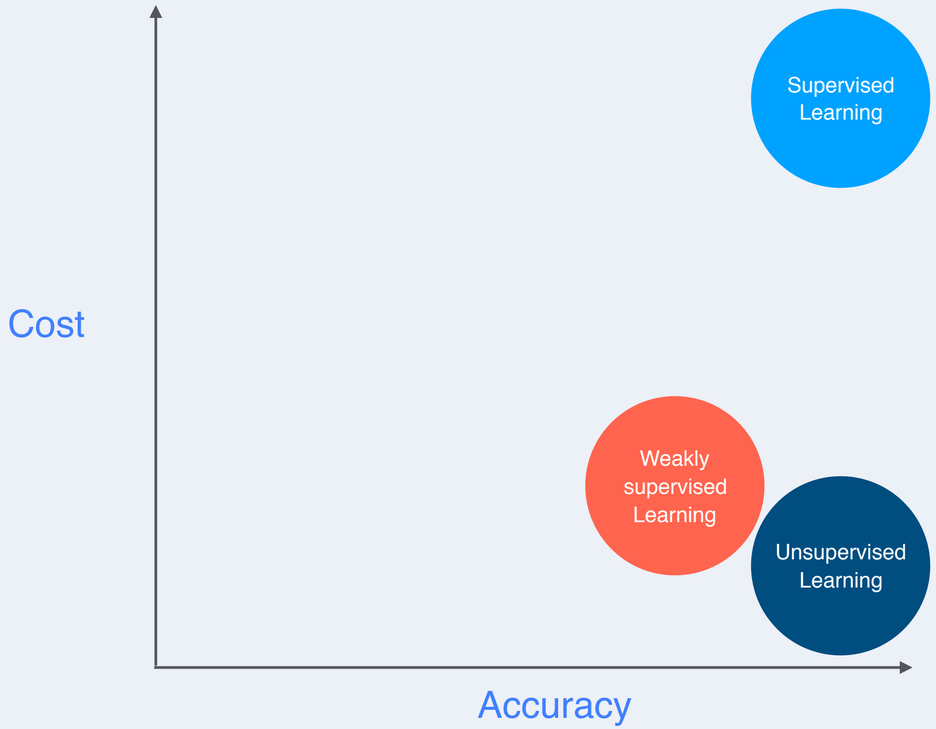
Medical Imaging



Self-driving cars

Cost

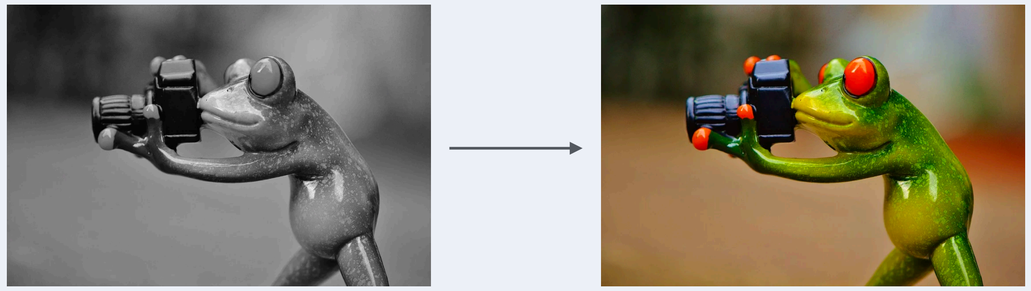

Supervised
Learning
Weakly
supervised
Learning
Unsupervised
Learning
Cost
Accuracy


Supervised
Learning
Cost
Weakly
supervised
Learning
Unsupervised
Learning
Accuracy

Unsupervised Learning vs self-supervised learning

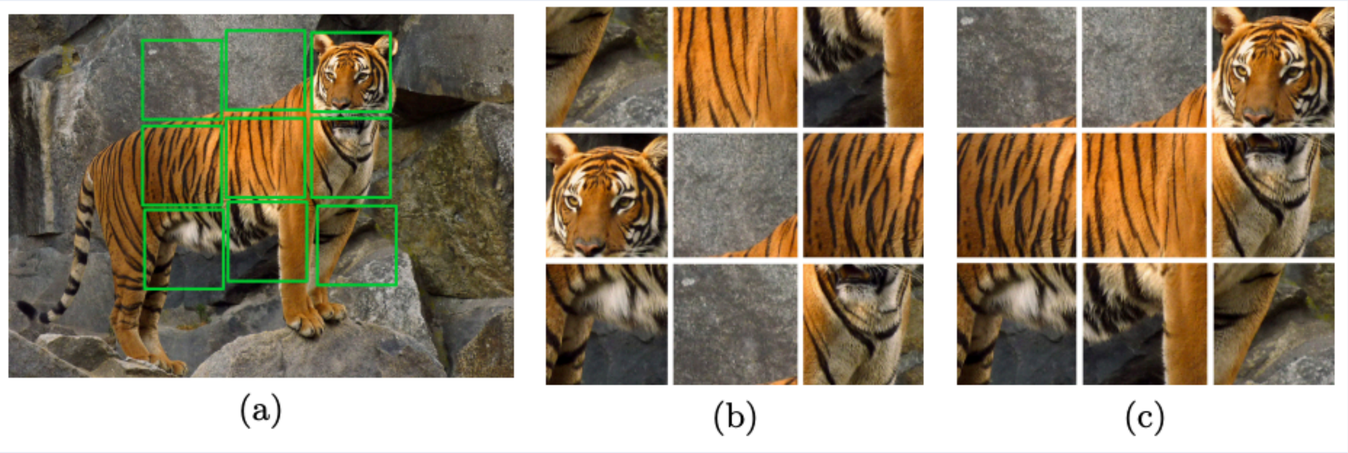
self-supervised learning

Colorful Image Colorization (Zhang et al 2016)



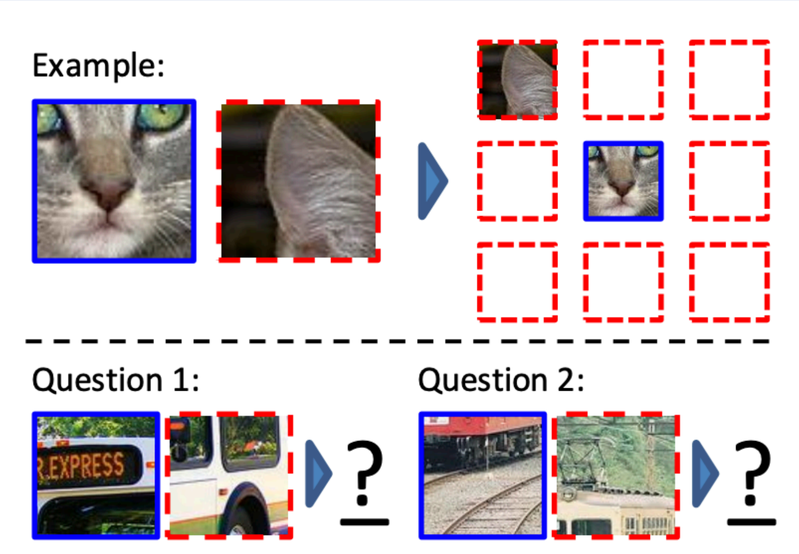
Zhang, R., Isola, P. and Efros, A.A., 2016, October. **Colorful image colorization**. In *European conference on computer vision* (pp. 649-666). Springer, Cham

Unsupervised Learning of Visual Representations by solving Jigsaw Puzzles (Mehdi et al, 2016)



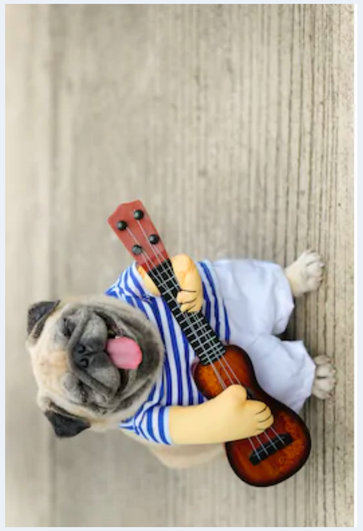
Noroozi, M. and Favaro, P., 2016, October. Unsupervised learning of visual representations by solving jigsaw puzzles. In European Conference on Computer Vision (pp. 69-84). Springer, Cham.

Unsupervised Visual Representation Learning by Context Prediction (Doersch et al, 2016)



Doersch, C., Gupta, A. and Efros, A.A., 2015. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1422-1430).

Unsupervised Representation Learning By Predicting Image Rotations
(Giradis et al, 2018)



Gidaris, S., Singh, P. and Komodakis, N., 2018. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*.

BERT: Pre-training of deep bidirectional transformers for language understanding
(Devlin et al, 2018)

Masked word prediction

This is a [MASK] long sentence with missing [MASK]

Next sentence prediction

i love AI because ⟶ it's crazy that it works

Why is this bad?

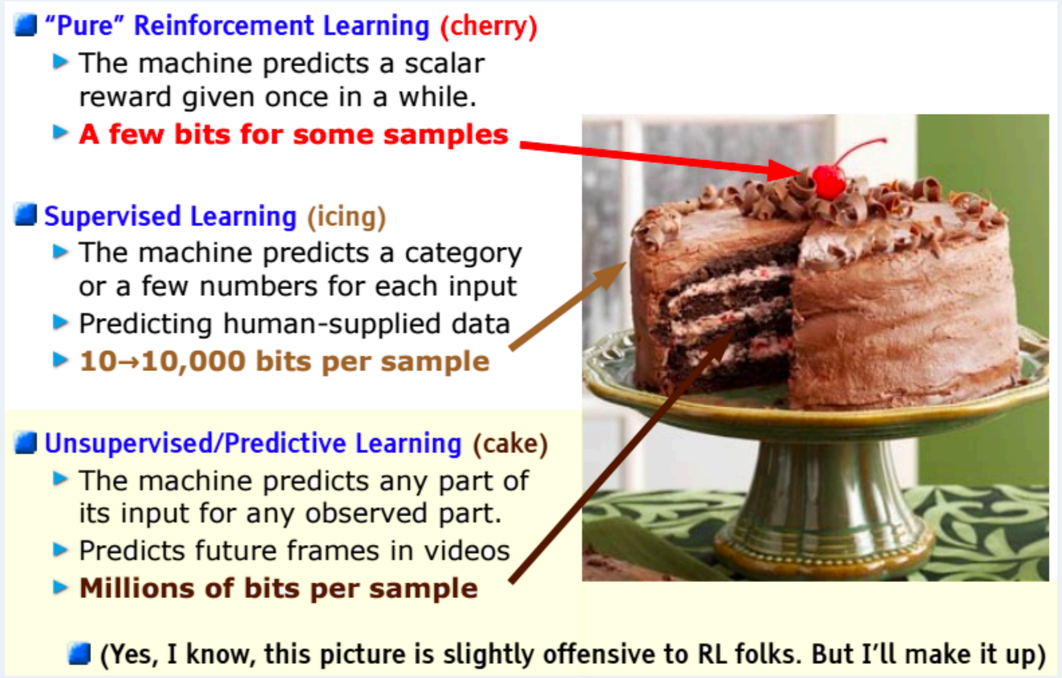
Humans don't likely learn like this

birth

4 months

5 months

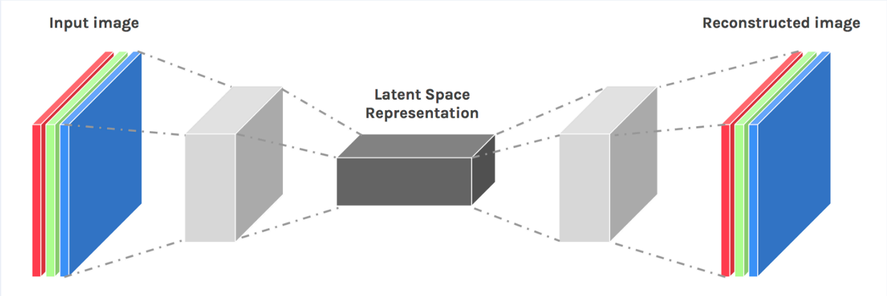
12 months



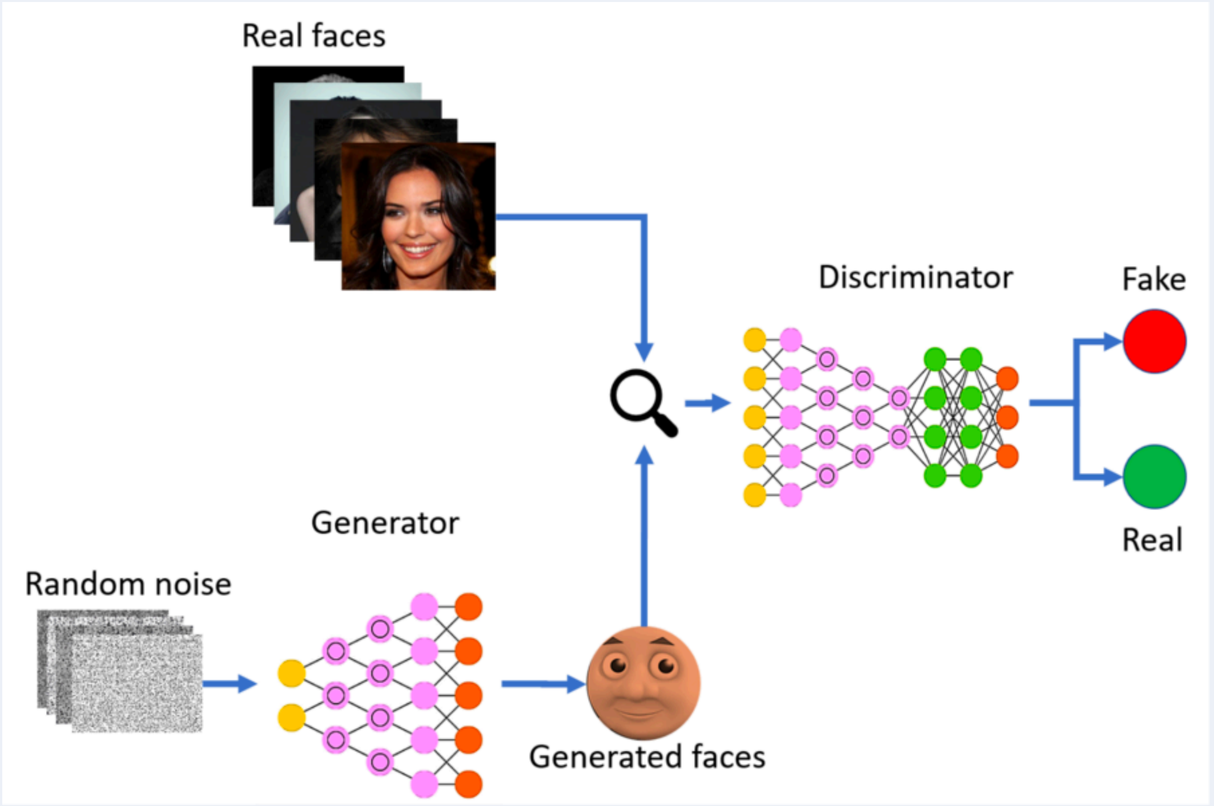
(Credit: Yann LeCun)

unsupervised learning

Autoencoder



Generative Adversarial Networks
(Goodfellow et al. 2015)



Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)

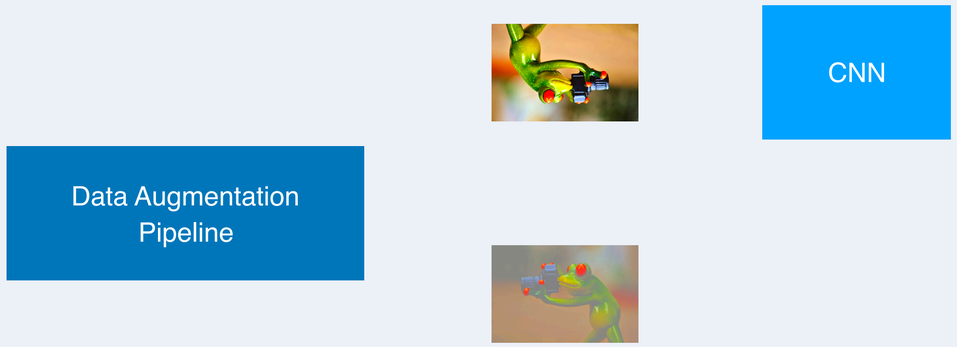
Data Augmentation
Pipeline

Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)

Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)

Data Augmentation
Pipeline

Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)



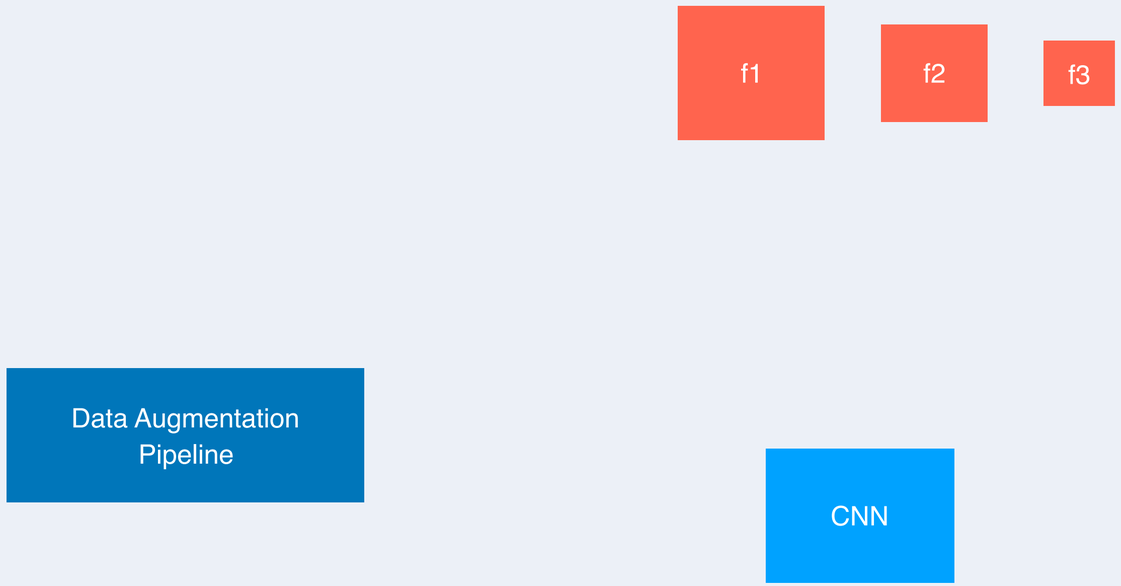
Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)

CNN

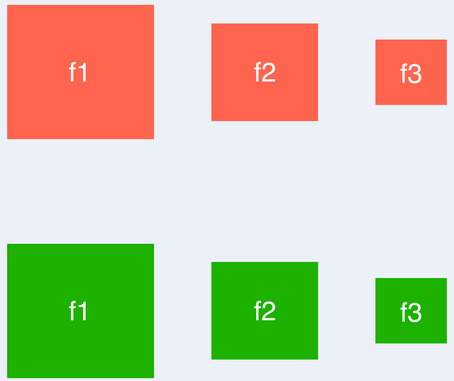
Data Augmentation
Pipeline

Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)

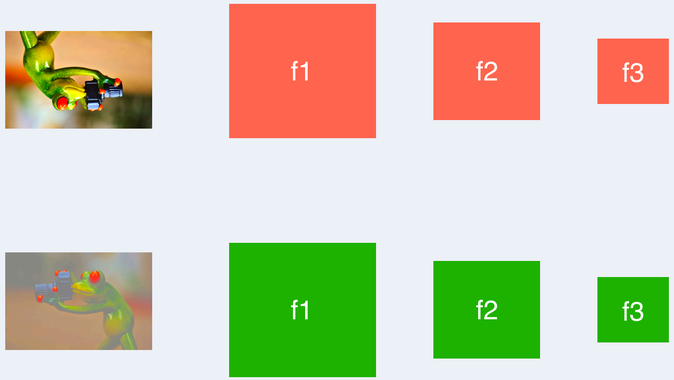
Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)



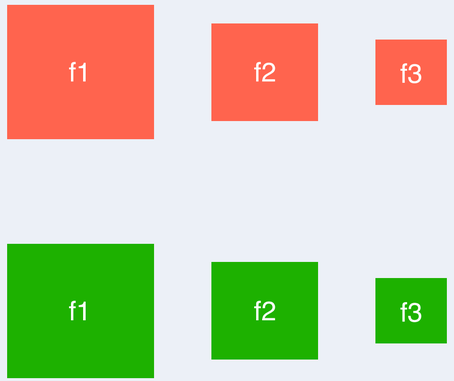
Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)



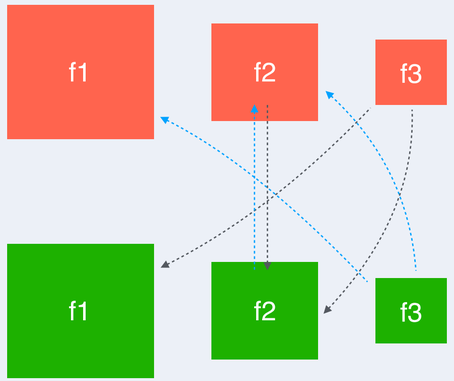
Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)



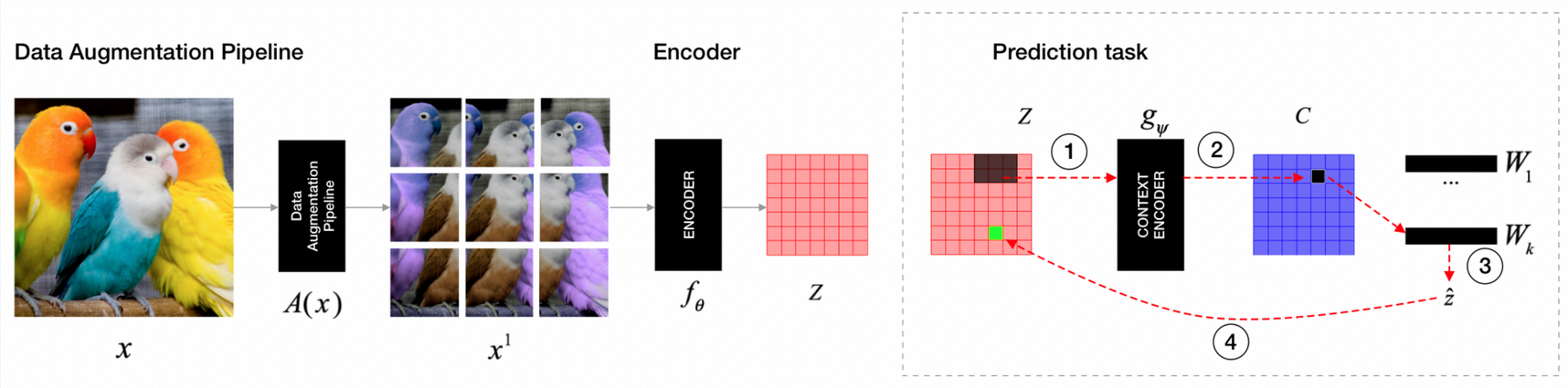
Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)



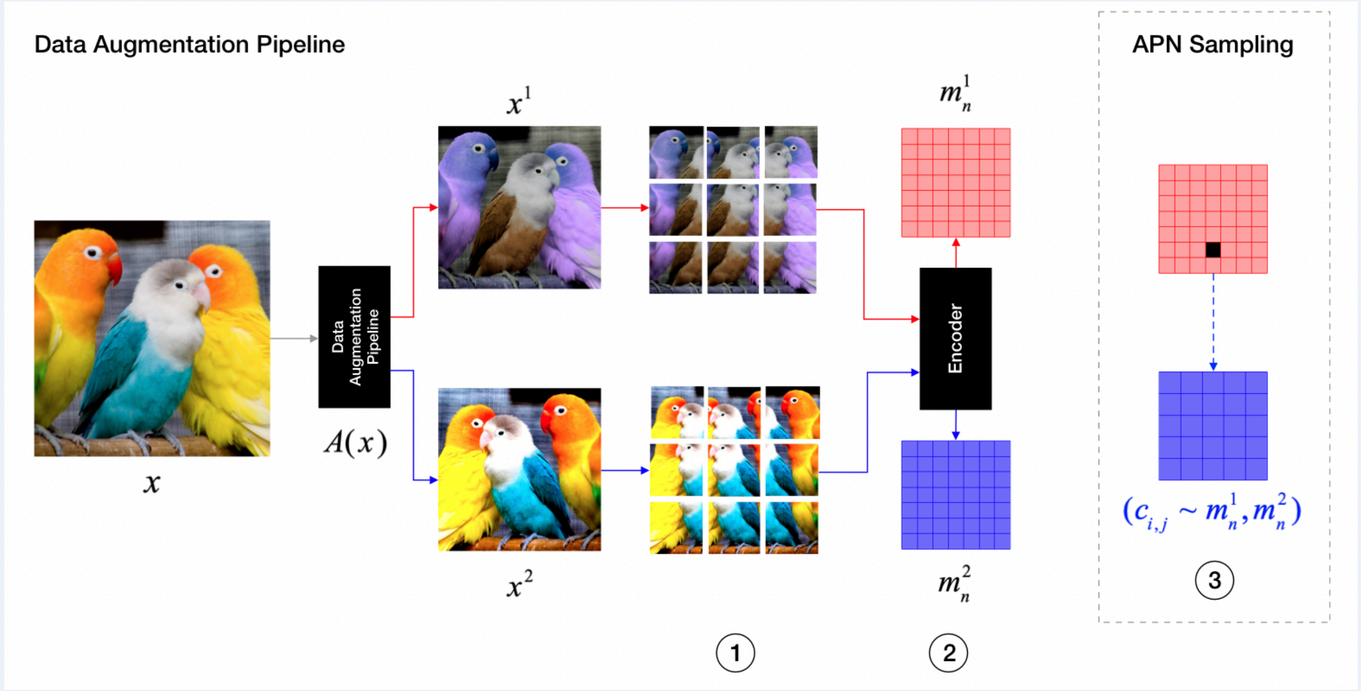
Learning Representations By Maximizing Mutual Information Across Views
(Bachman et al, 2019)



Data-efficient Image Recognition with Contrastive Predictive Coding
(Hennaff, 2019)



A General Framework For Self-Supervised Image Representation Learning and PatchedDIM
(Falcon, Cho, 2019)



Scaling

williamFalcon / pytorch-lightning

Used by 25 | Unwatch 60 | Star 2.7k | Fork 239

Code | Issues 43 | Pull requests 7 | Projects 1 | Wiki | Security | Insights | Settings

1,609 commits | 5 branches | 64 releases | 1 environment | 47 contributors | Apache-2.0

PyTorch Lightning

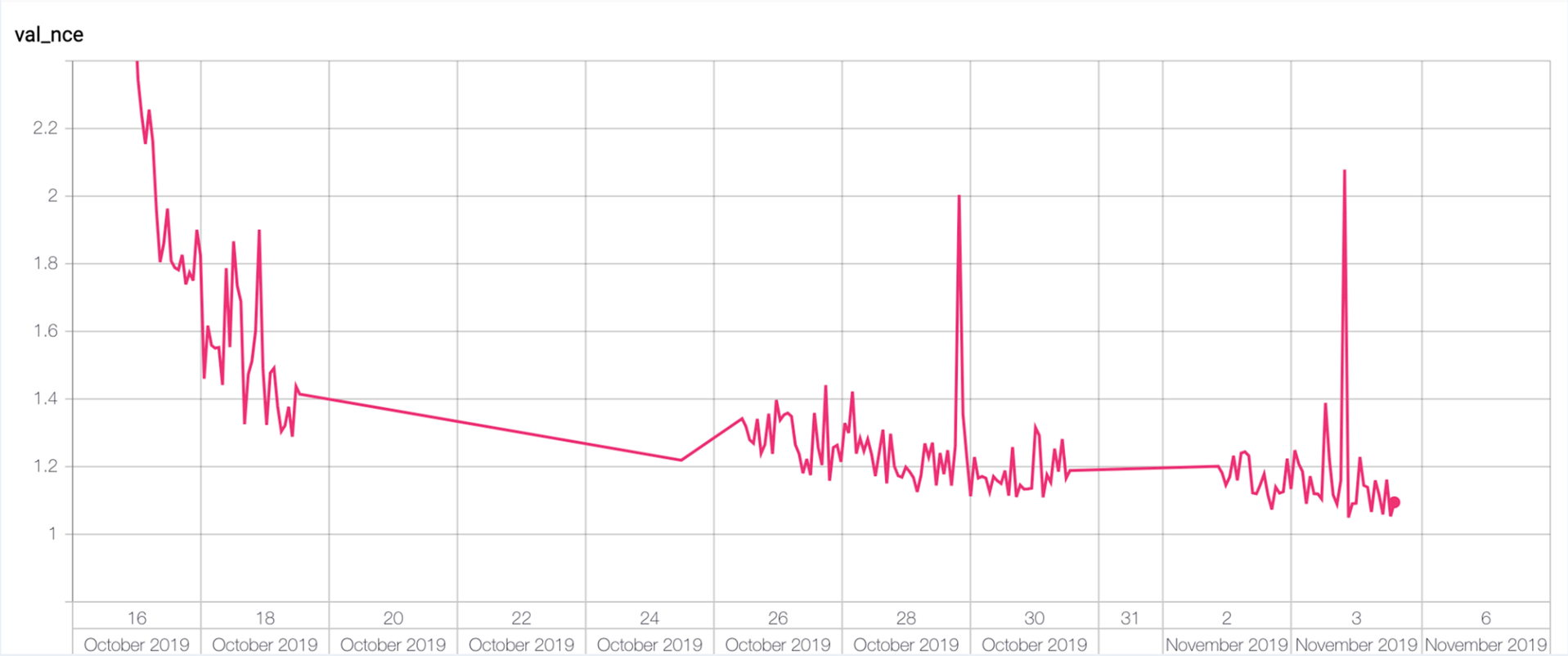
The lightweight PyTorch wrapper for ML researchers. Scale your models. Write less boilerplate.

pypi package 0.5.2.1 | downloads 62k | build passing | build unknown | coverage 99% | codefactor A

docs passing | chat on gitter | License Apache 2.0 | Next Release Nov 6

Simple installation from PyPI

```
pip install pytorch-lightning
```


val_nce
2.2
2
1.8
1.6
1.4
1.2
1
16
October 2019
18
October 2019
20
October 2019
22
October 2019
24
October 2019
26
October 2019
28
October 2019
30
October 2019
31
2
November 2019
3
November 2019
6
November 2019

Addressing Reproducibility Crisis

Reproducibility Challenge NeurIPS 2019 Task Description Resources Registration Important Dates Organizers

Reproducible Code

- If you are working in PyTorch, we strongly recommend using Pytorch Lightning, a framework which takes care of the boilerplate and provides highly reproducible standards of ML research pipeline. Check the seed project as a good starting point.
- Document your code appropriately
- Have a `README.md` file which describes the exact steps to run your code

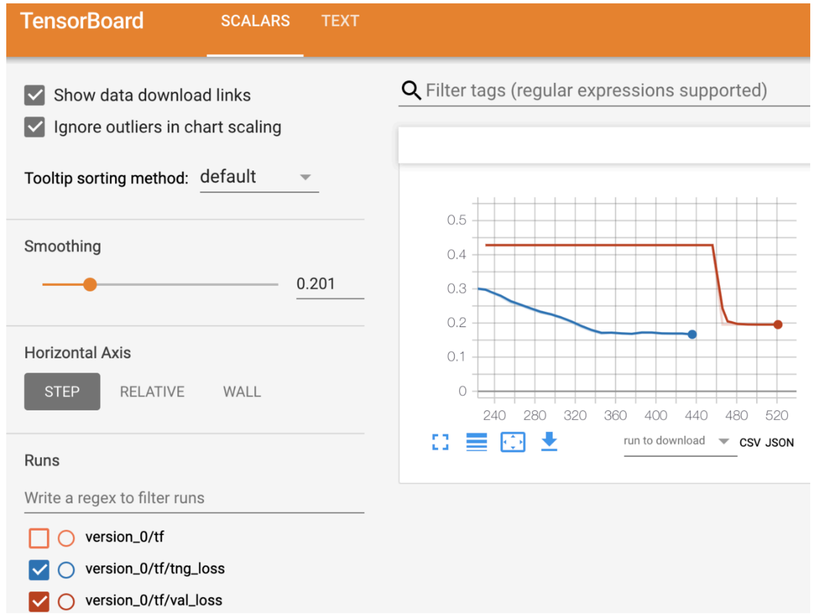
LightningModule

```
class CoolSystem(pl.LightningModule):

    def __init__(self):
        super(CoolSystem, self).__init__()
        self.l1 = torch.nn.Linear(28 * 28, 10)


    def forward(self, x):
        return torch.relu(self.l1(x.view(x.size(0), -1)))


    def training_step(self, batch, batch_nb):
        x, y = batch
        y_hat = self.forward(x)
        loss = F.cross_entropy(y_hat, y)

        tensorboard_logs = {'train_loss': loss}
        return {'loss': loss, 'log': tensorboard_logs}


    def validation_step(self, batch, batch_nb):
        x, y = batch
        y_hat = self.forward(x)
        return {'val_loss': F.cross_entropy(y_hat, y)}


    def validation_end(self, outputs):
        avg_loss = torch.stack([x['val_loss'] for x in outputs]).mean()

        tensorboard_logs = {'val_loss': avg_loss}
        return {'avg_val_loss': avg_loss, 'log': tensorboard_logs}


    def configure_optimizers(self):
        return torch.optim.Adam(self.parameters(), lr=0.02)


    @pl.data_loader
    def train_dataloader(self):
        return DataLoader(MNIST(os.getcwd(), train=True, download=True, transform=transforms.ToTensor()), batch_size=32)


    @pl.data_loader
    def val_dataloader(self):
        return DataLoader(MNIST(os.getcwd(), train=True, download=True, transform=transforms.ToTensor()), batch_size=32)

    @pl.data_loader
    def test_dataloader(self):
        return DataLoader(MNIST(os.getcwd(), train=True, download=True, transform=transforms.ToTensor()), batch_size=32)
```

LightningModule

```
model = CoolSystem()
```

```
trainer = Trainer()
trainer.fit(model)
```



Automatic Tensorboard

Automatic training loop

Automatic validation loop

Automatic checkpointing

Automatic early-stopping

In summary

Unsupervised is state-of-the-art in NLP (BERT, GPT-2)

Computer vision is lagging behind (transfer learning is ok but not great)

Unsupervised Learning will unlock new ways of using data

We need to move away from images and clever tasks

Self-supervised gains come from data processing NOT learning

Thank you
@_willfalcon