Applied Deep Learning

Nitish Bhardwaj

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Summary of Session I

- AI to Deep Learning
- Neural networks to Deep nets
- Problem Solving
- Coding session
  - Image Classification
  - Convolutional Neural network
  - Object Detection
  - Segmentation
- More examples

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

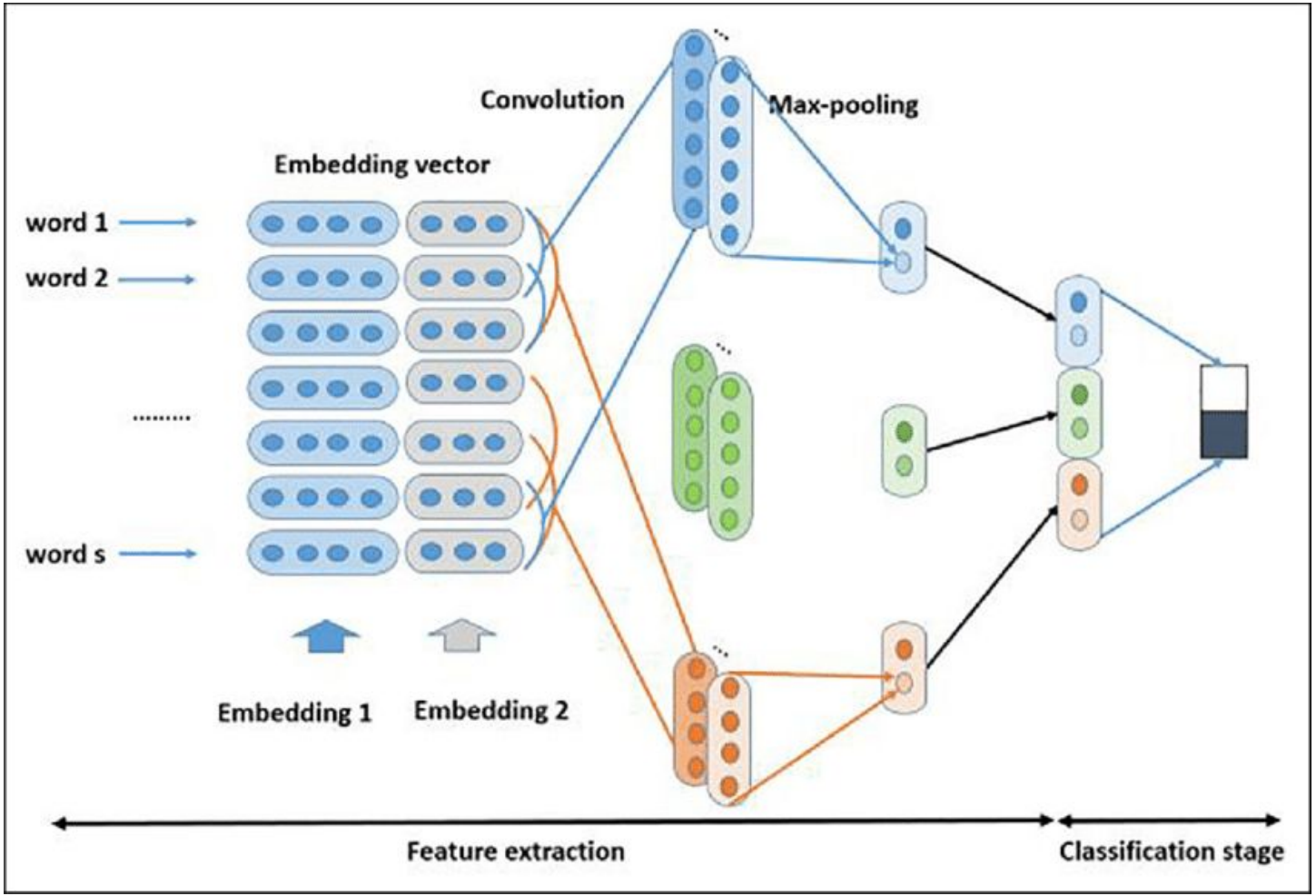
Overview of Session II :

- Text Classification
- AI Pipeline
- Industrial Requirement
- Research Topics in Deep Learning
- GAN

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Text Classification

Text Classification



Embedding Vector

- Word index
- TF-IDF
- word2vec
- Doc2vec
- Glove
- Transformers
- BERT

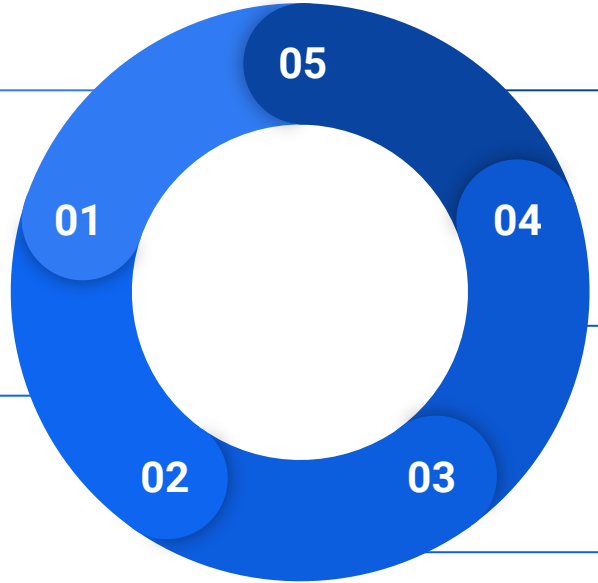
Resume Shortlisting

Sentiment Analysis : Walkthrough

AI Pipeline

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Business Use-case :
Break into Problem Statements



AI R&D for each problem

Deployment

Rephrase / Recycle

Aggregate Results

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

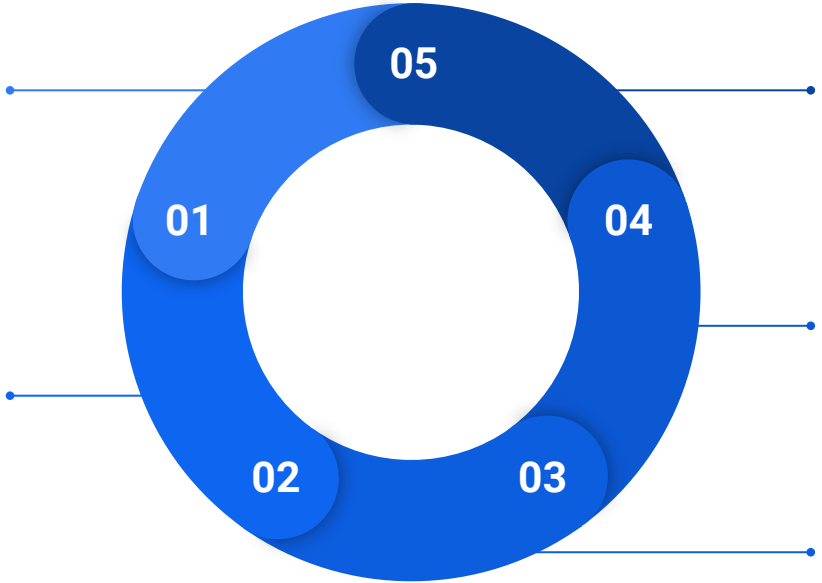
Example : Smart Retail Store

Smart Retail Store :

- **Smart Basket**
- **Chatbot for Assistant**
- **…….**

AI R&D for each problem

- **Computer Vision**
- **Natural Language Processing**
- **……..**



Rephrase / Recycle

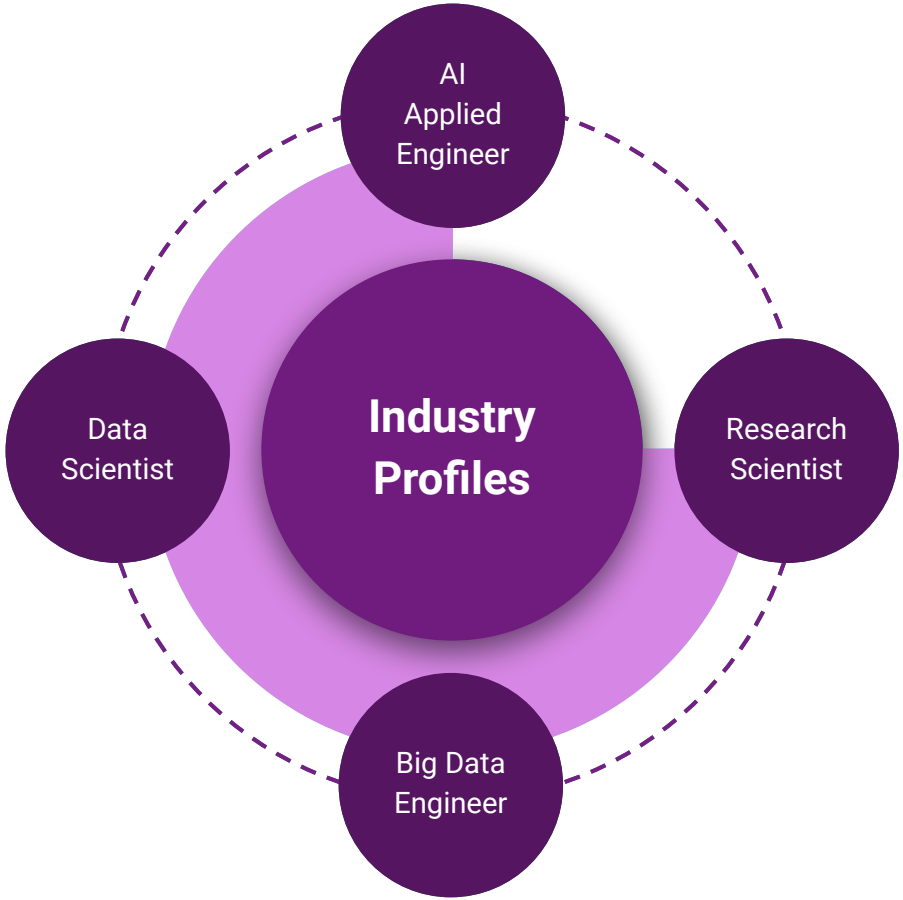
Deployment

Aggregate Results

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Industrial Requirement

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/



Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Industry Profiles

- Use of existing AI models
- Which models for which problem statement?
- How to re-train using Transfer Learning?

AI Applied Engineer

- Research for unsolved problems
- Approach to improve accuracy
- Academic Collaboration

Research Scientist

- Important Data Extraction
- Which Data models?
- How to show results?

Data Scientist

- Capturing Data
- Storing Data
- Streaming Data
- Processing Data

Big Data Engineer

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Industry Academic Collaboration

- Research to find new mathematical approach, ex. Loss functions
- Exploration of new domain like 3D, Graph neural Nets
- A lot of experimentations and Trials-Errors
- Solving new problems : COVID-19

Transfer Learning

Using Pre-trained Models

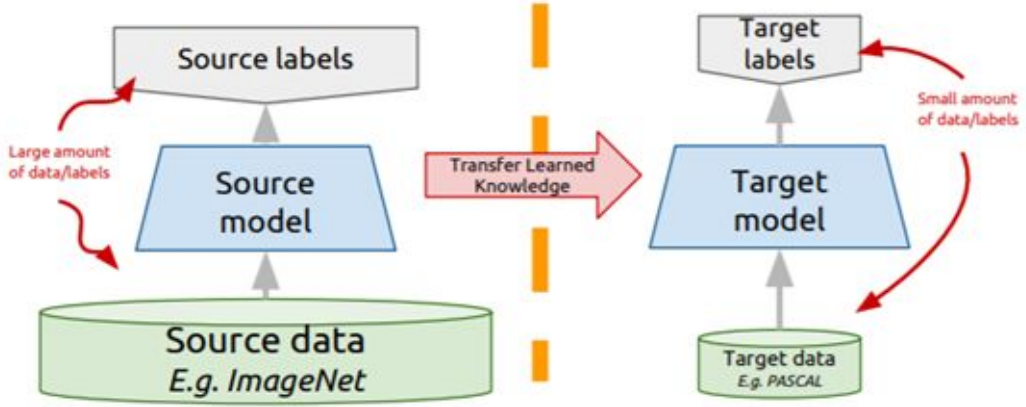
Transfer learning: idea

Instead of training a deep network from scratch for your task:

- Take a network trained on a different domain for a different **source task**
- Adapt it for your domain and your **target task**

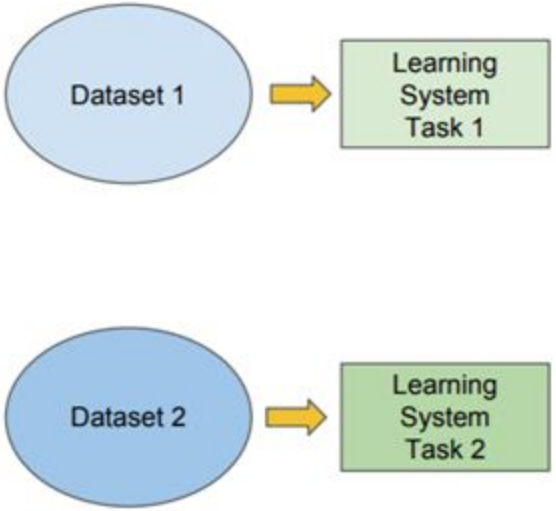
Variations:

- Same domain, different task
- Different domain, same task

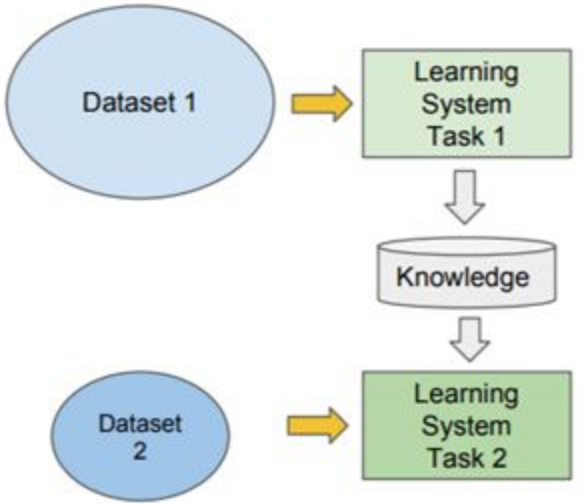


Traditional ML vs Transfer Learning

- Isolated, single task learning:
  - Knowledge is not retained or accumulated. Learning is performed w.o. considering past learned knowledge in other tasks



- Learning of a new tasks relies on the previous learned tasks:
  - Learning process can be faster, more accurate and/or need less training data



TensorFlow

Reusing Pretrained models : Transfer Learning

PyTorch

Transfer Learning Approaches

- Method 1 : *Using online repository of the trained model*
- Method 2 : *Using models defined in the model in the framework*
- Method 3 : *Downloading the model in your local and using it*

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Method 1 : *Using online repository of the trained model*

- Using tensorflow hub
  - Online pre-trained models
  - https://www.tensorflow.org/tutorials/images/transfer_learning_with_hub
  - https://github.com/tensorflow/hub/blob/master/examples/colab/object_detection.ipynb

- Using pytorch hub
  - https://pytorch.org/hub/research-models
  - https://pytorch.org/hub/nvidia_deeplearningexamples_ssd/
  - ```
    import torch
    ```
  - ```
    precision = 'fp32'
    ```
  - ```
    ssd_model = torch.hub.load('NVIDIA/DeepLearningExamples:torchhub', 'nvidia_ssd', model_math=precision)
    ```

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/
    ```

Method 2 : *Using models defined in the model in the framework*

- Using tensorflow trained models defined in library as application
  - https://www.tensorflow.org/tutorials/images/transfer_learning#create_the_base_model_from_the_pre-trained_convnets
  - ```
    # Create the base model from the pre-trained model MobileNet V2
    base_model = tf.keras.applications.MobileNetV2(input_shape=IMG_SHAPE, include_top=False,
                                                   weights='imagenet')
    ```

- Using pytorch trained models defined in library
  - https://pytorch.org/docs/stable/torchvision/models.html
  - ```
    import torchvision.models as models
    ```
  - ```
    resnet18 = models.resnet18(pretrained=True)
    ```
  -

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/
    ```

Method 3 : *Downloading the model in your local and using it*

- Keras Model : .h5
- Pytorch model : .pt, .pth, .onnx
- Tf Model : .ckpt, .pb

- Tensorflow :
  - Exporting a model : https://www.tensorflow.org/api_docs/python/tf/saved_model/save
  - Re-using the exported model : https://www.tensorflow.org/guide/saved_model
- Pytorch:
  - Approach 1: https://pytorch.org/tutorials/advanced/super_resolution_with_onnxruntime.html
  - Approach 2: https://pytorch.org/tutorials/beginner/saving_loading_models.html

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Keras Examples : walkthrough

Research Topic in Deep Learning

- **A Comprehensive Survey on Graph Neural Networks** Wu, Zonghan, et al. in cs.LG and stat.ML, latest revision 12/4/2019 : 1901.00596v4: Abstract – Full Paper (pdf)

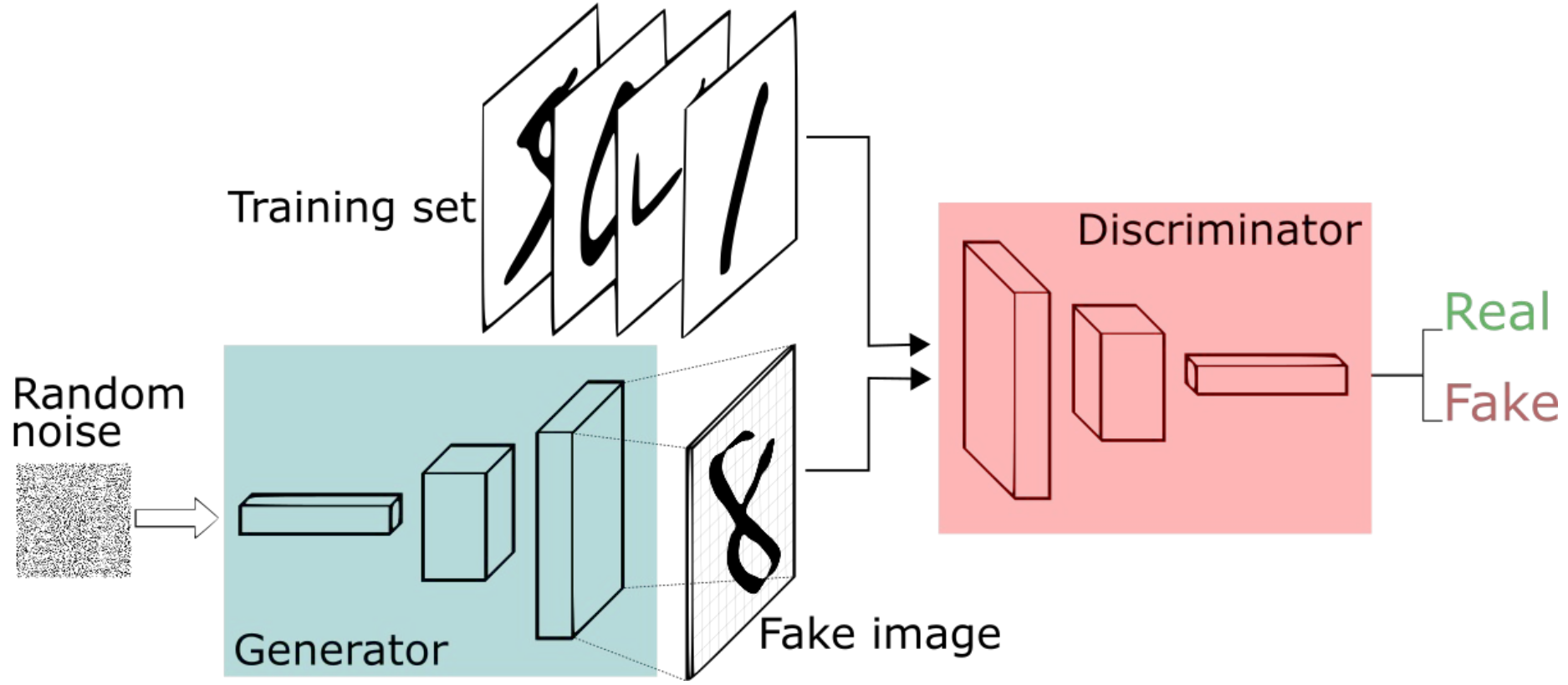
- **EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks** Tan, Mingxing and Le, Quoc in cs.LG, cs.CV and stat.ML, latest revision 11/23/2019 : 1905.11946v3: Abstract – Full Paper (pdf)

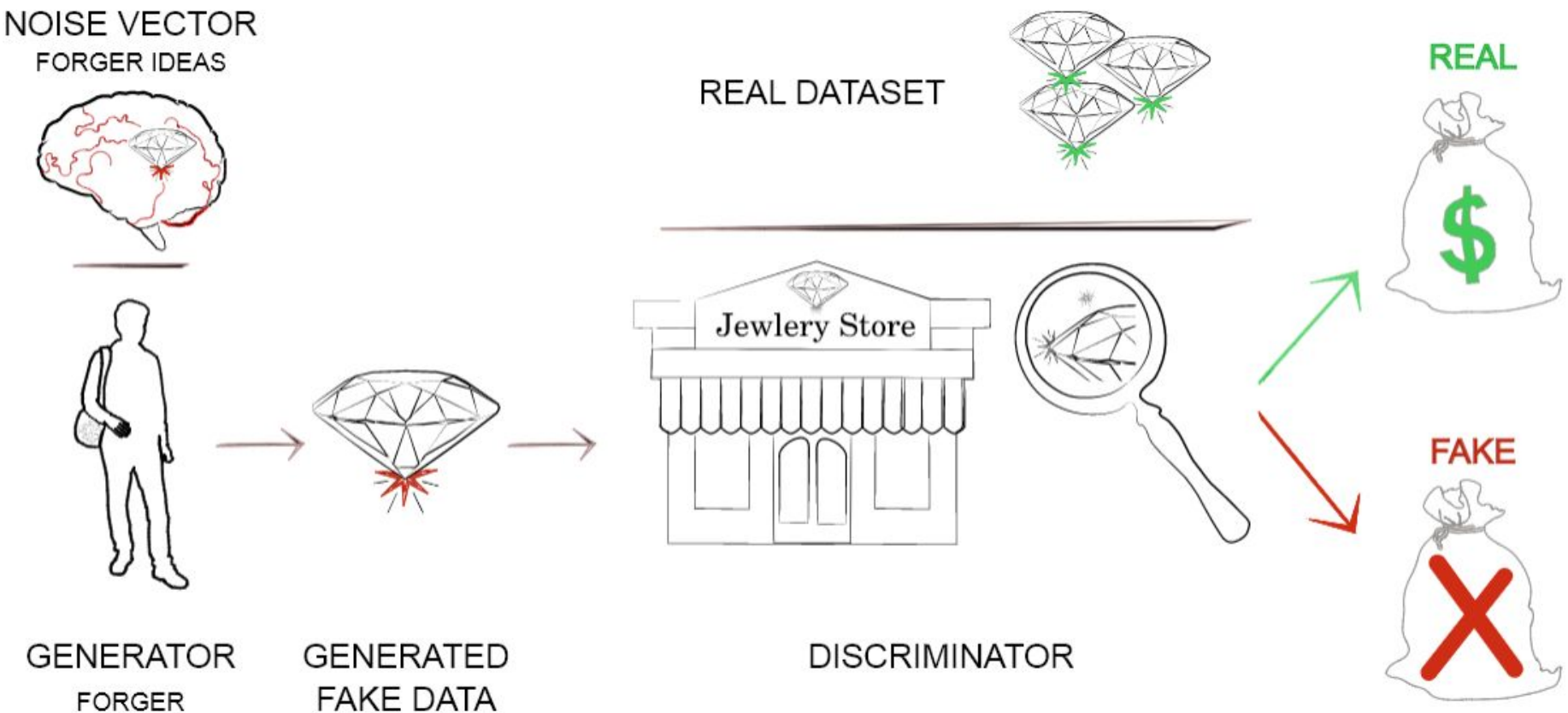
- **Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context :** Dai, Z., et al. in cs.LG | cs.CL | stat.ML, latest revision 6/2/2019 1901.02860v3: Abstract – Full Paper (pdf)

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/

Elephant in the House : **GAN**

Compiled By Nitish Bhardwaj
https://www.linkedin.com/in/nitish-bhardwaj/


Training set
Discriminator
Real
Fake
Random noise
Generator
Fake image


NOISE VECTOR
FORGER IDEAS
REAL DATASET
REAL
Jewlery Store
$
FAKE
GENERATOR
FORGER
GENERATED
FAKE DATA
DISCRIMINATOR

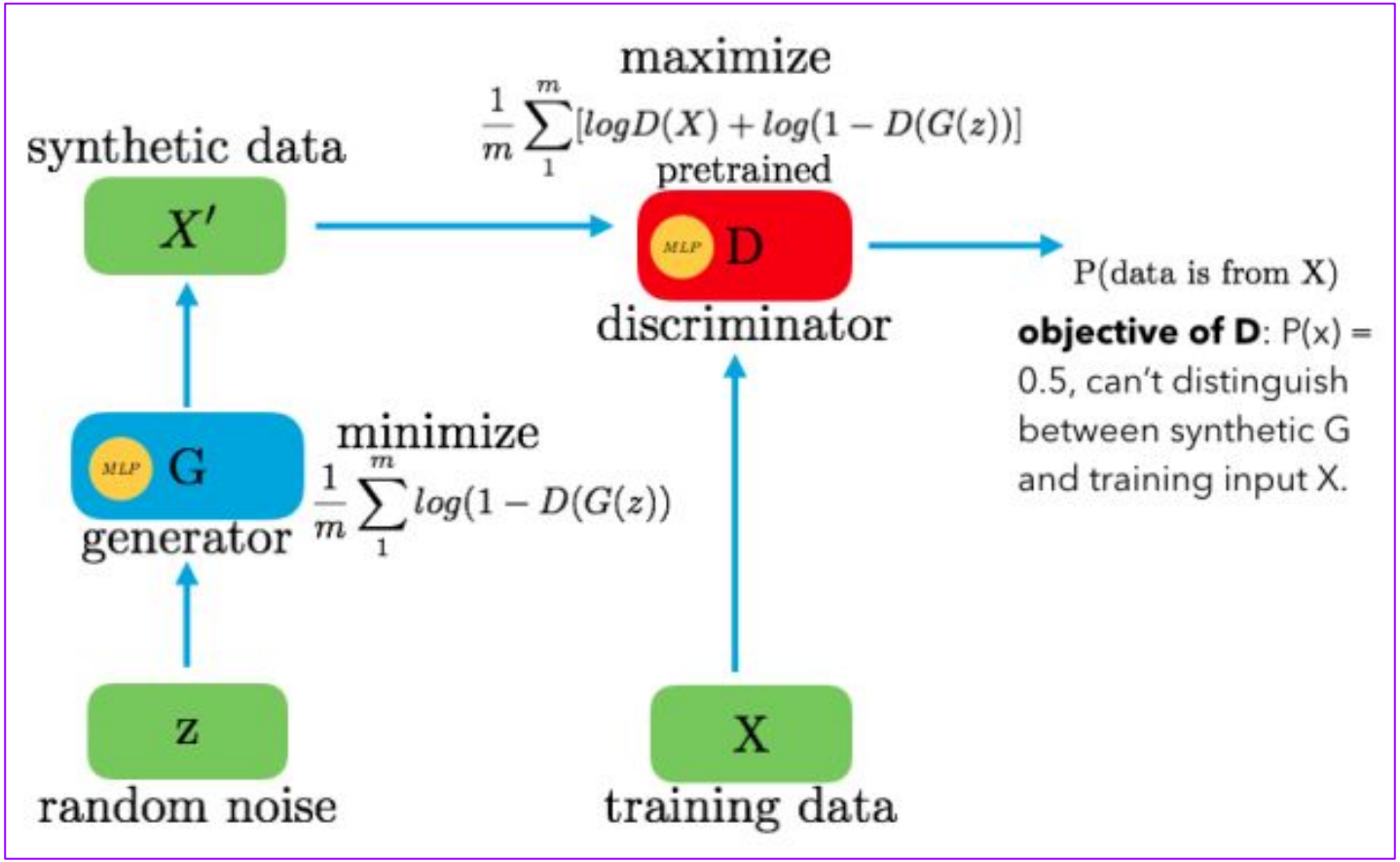
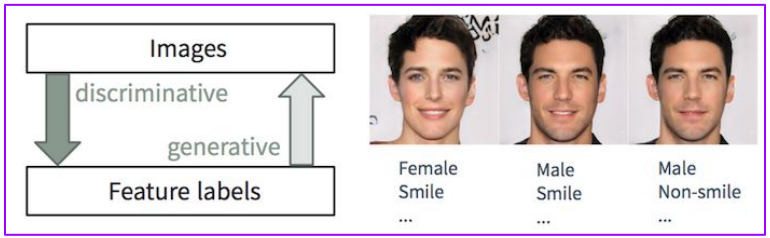
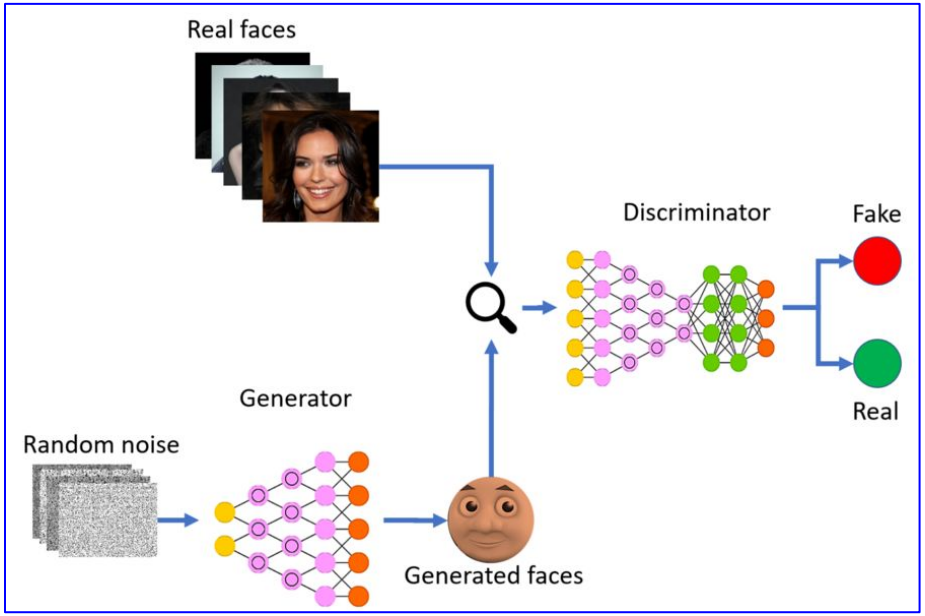

maximize
$\frac{1}{m}\sum_{1}^{m}[logD(X) + log(1 - D(G(z))]$
synthetic data
X'
pretrained
MLP D
discriminator
P(data is from X)
objective of D: P(x) = 0.5, can't distinguish between synthetic G and training input X.
MLP G
generator
minimize
$\frac{1}{m}\sum_{1}^{m} log(1 - D(G(z))$
z
random noise
X
training data

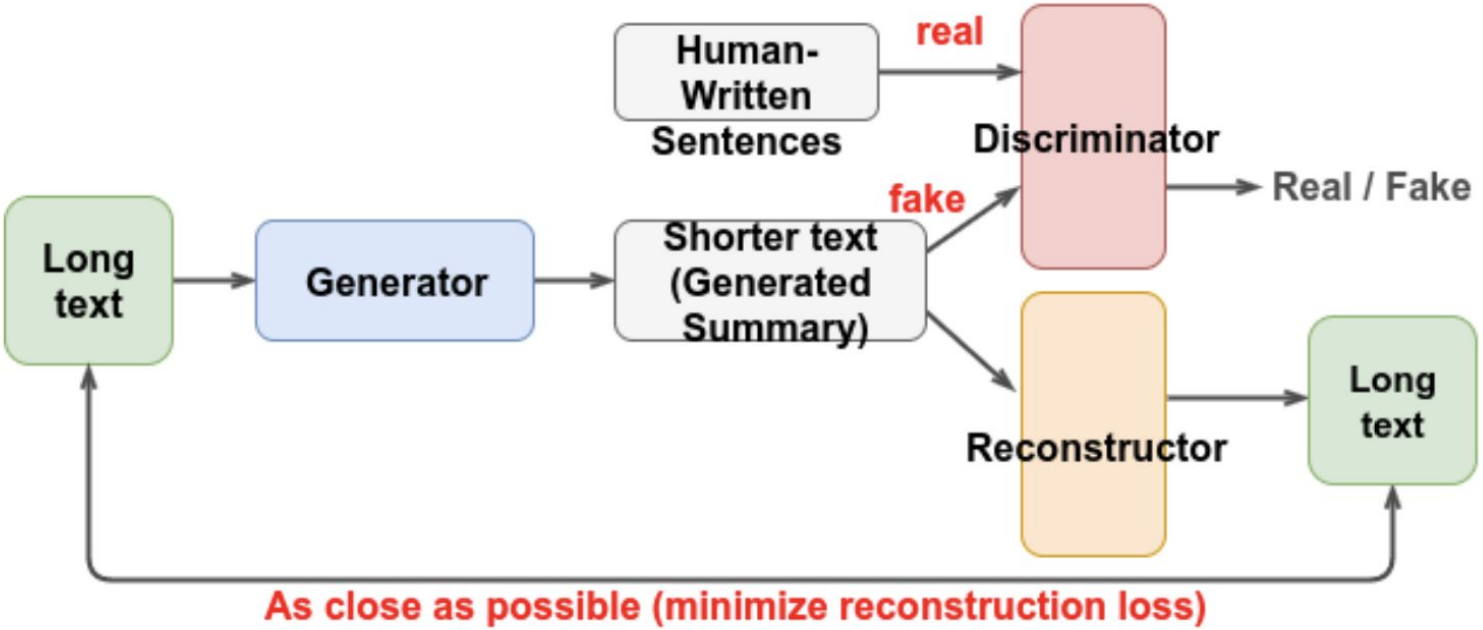
Image generation

[Image Generation Demo]





Sentence Generation



Code-Walkthrough

- PixelCNN
- Face Generation
- GANs

Summary

- Text classification
- Story of Resume Shortlisting
- Coding Session
  - Sentiment Analysis I
  - Sentiment Analysis II
- AI Pipeline
- Industrial roles
- Research Topics
- GAN models

References

- https://paperswithcode.com/
- https://www.kdnuggets.com/2020/01/top-10-ai-ml-articles-to-know.html
- https://www.aclweb.org/anthology/D18-1451.pdf
- https://medium.com/what-is-gan/conditional-gan-d62a76e1724f
- https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/
- https://colab.research.google.com/drive/1gAS_eDzGRhEznUosanlUvjv-g96jfQZE#scrollTo=39CgD8t68Otg
- https://colab.research.google.com/github/agungsantoso/deep-learning-v2-pytorch/blob/master/sentiment-rnn/Sentiment_RNN_Exercise.ipynb#scrollTo=TT8spavKpmxH
- https://github.com/hindupuravinash/the-gan-zoo

Thank You