Deep Learning developed and evolved for image processing and computer vision applications.

It is now increasingly and successfully used on signals and time series.
The Use of Deep Learning is Growing Across Industries

**Aerospace, Defense and Communications**
- Communications devices, security
- Multi-standard communications receivers, drone recognition

**Consumer Electronics and Digital Health**
- Voice assistants
- Digital health

**Automotive**
- Voice control enabled Infotainment
- Sensor processing, automated driving

**Industrial Automation**
- Condition monitoring
- Predictive maintenance

MATLAB EXPO 2019

The Use of Deep Learning is Growing Across Industries
Application Examples Using MATLAB – Audio and Speech

Speech Command Recognition (a.k.a. "Keyword Spotting")

Music Genre Classification
Application Examples Using MATLAB – Industrial and physiological sensors

Human Activity Recognition


ECG Signal Classification

Application Examples Using MATLAB – Radar and Communications

Modulation Classification

Agenda

- Deep Learning – Basic ideas
- Deep Learning Model Development for Signals, Time Series, and Text
- Conclusions
What is Deep Learning?

Deep learning is a type of machine learning in which a model learns from examples.
Common Network Architectures - Signal Processing

Time-Frequency Transformation

Convolutional Neural Networks (CNN)

Long Short Term Memory (LSTM) Networks

Feature Engineering
Common Network Architectures – Text Analytics

Convolutional Neural Networks (CNN)

Long Short Term Memory (LSTM) Networks
Deep Learning Workflow

CREATE AND ACCESS DATASETS

Data sources
- Simulation and augmentation
- Data Labeling

PREPROCESS AND TRANSFORM DATA

Pre-Processing
- Transformation
- Feature extraction

DEVELOP PREDICTIVE MODELS

Import Reference Models/Design from scratch
- Hardware-Accelerated Training
- Analyze and tune hyperparameters

ACCELERATE AND DEPLOY

Desktop Apps
- Enterprise Scale Systems
- Embedded Devices and Hardware
Deep learning models only as good as training data

- **Data Labeling**

- **Limited Data Availability**

Deep Learning Workflow Challenges – Signals and Time Series

- Domain-specific data processing desirable
- Limited reference research
- Deployment and Scaling to various platforms
Agenda

- Deep Learning – Basic ideas

- Deep Learning Model Development for Signals, Time Series, and Text
  - Data
  - Processing and transformation
  - Model design and optimization
  - Acceleration, prototyping, and deployment

- Conclusions
Agenda

▪ Deep Learning – Basic ideas

▪ Deep Learning Model Development for Signals, Time Series, and Text
  – Data
  – Processing and transformation
  – Model design and optimization
  – Acceleration, prototyping, and deployment

▪ Conclusions
Current Investments – Models vs. Data

From "Troubleshooting deep neural networks" (Josh Tobin et al., Jan 2019)
What does a large dataset look like?

How to navigate, index, read (aka "ingest") a large dataset?

How to...

▪ Build a list of all data and labels?

▪ Review basic statistics about available data?

▪ Select data subsets without nested for loops, `dir`, `ls`, `what`, ... aplenty?

▪ Jointly read data and labels?

▪ Automatically distribute computations?

**MathWorks**

**MATLAB EXPO 2019**
Label quality impacts model performance as much as the quality and quantity of the actual recordings
Use appropriate tools to help you label signals

- Programmmatically…
- … or via Apps
What if available data isn't enough?

Data augmentation allows building more complex and more robust models.

- Pitch shift
- Time stretch
- Add noise
- Add reverberation

Original Dataset

Augmented Dataset

N times as much data
Simulation is key if recording and labelling real-world data is impractical or unreasonable – **Communications Signals**
Simulation is key if recording and labelling real-world data is impractical or unreasonable – Radar Signals

Radar Target Simulation

Micro-Doppler Analysis
Agenda

- Deep Learning – Basic ideas

- Deep Learning Model Development for Signals, Time Series, and Text
  - Data
  - Processing and transformation
    - Model design and optimization
    - Acceleration, prototyping, and deployment

- Conclusions
Common types of network architectures used in signal processing and text analytics applications

- **Feature Engineering**
- **Time-Frequency Transformation**
  - Convolutional Neural Networks (CNN)
  - Long Short Term Memory (LSTM) Networks

MATLAB EXPO 2019
Time-Frequency Transformations

Reframe (e.g. Buffer) → To frequency (e.g. FFT)

- Basic spectrogram
  - Easiest to understand and implement
- Perceptually-spaced (e.g. Mel, Bark) Spectrogram
  - More compact for speech & audio applications
- Wavelet scalogram
  - Best resolution, for non-periodic signals
- Constant Q transform
  - Better resolution at low frequencies

Easiest to understand and implement
More compact for speech & audio applications
Best resolution, for non-periodic signals
Better resolution at low frequencies
Extracting Features from Signals: Application-Agnostic Examples

**Frequency domain**

- BW measurements
- Spectral statistics
- Harmonic analysis
- Octave spectrum

**Time domain**

- Find peaks
- Find signal patterns
- Detect change points
- Find signal envelope

MATLAB EXPO 2019
Domain-Specific Features and Transformations – Examples

Speech and Audio
- MFCC
- GTCC
- MDCT
- Pitch, harmonicity
- Spectral shape descriptors
- ...

Navigation and Sensor Fusion
- Orientation
- Height
- Position
- Multi-object tracking
- Acceleration, angular velocity
- Magnetic field
- GPS reading

Text Analytics
- Train Word Embeddings
- Word2Vec
- Topic Modeling
- ...

Radar
- Micro-Doppler analysis
- Range-Doppler processing
- Synthetic aperture imaging
- Spectral analysis
- Waveform ambiguity
- ...
Automated Feature Extraction: Wavelet Scattering

- Can relieve requirements on amount of data and model complexity
  - Featured in leader-boards a number of research competitions

- Framework for extracting features [1]

Agenda

- Deep Learning – Basic ideas

- Deep Learning Model Development for Signals, Time Series, and Text
  - Data
  - Processing and transformation
  - Model design and optimization
  - Acceleration, prototyping, and deployment

- Conclusions
Developing Deep Learning Models

Design Network

Design

Train

Accelerate Training

MATLAB as a container on NGC

Optimize

Model Exchange

Pre-trained Networks

Bayesian Hyperparameter Optimization

MATLAB EXPO 2019
Exchange Models With Deep Learning Frameworks

ONNX = Open Neural Network Exchange Format
Exchange Models With Deep Learning Frameworks

ONNX = Open Neural Network Exchange Format
Agenda

- Deep Learning – Basic ideas

- Deep Learning Model Development for Signals, Time Series, and Text
  - Data
  - Processing and transformation
  - Model design and optimization
  - Acceleration, prototyping, and deployment

- Conclusions
Deployment and Scaling for A.I.
Deploying Deep Learning Models for Inference

Application logic

Code Generation

Auto-generated Code (C/C++/CUDA)

Intel MKL-DNN Library

NVIDIA TensorRT & cuDNN Libraries

ARM Compute Library
With **GPU Coder**, MATLAB is fast

**Single Image Inference (Titan V, Linux)**

**R2019a**

GPU Coder is faster than TensorFlow, MXNet and Pytorch

- **TensorFlow**
- **MXNet**
- **GPU Coder**
- **PyTorch**

*Intel® Xeon® CPU 3.6 GHz - NVIDIA libraries: CUDA10 - cuDNN 7 - Frameworks: TensorFlow 1.13.0, MXNet 1.4.0 PyTorch 1.0.0*
Enterprise Deployment

Deployment to the cloud with MATLAB Compiler and MATLAB Production Server
Agenda

- Deep Learning – Basic ideas

- Deep Learning Model Development for Signals, Time Series, and Text
  - Data
  - Processing and transformation
  - Model design and optimization
  - Acceleration, prototyping, and deployment

- Conclusions
Deep Learning Workflow Challenges – Signals and Time Series

Deep learning models only as good as training data

Domain-specific data processing desirable

Application-specific algorithms and tools

Collaboration in the AI ecosystem

Data-labeling Apps and Examples

Augmentation and simulation algorithms

Deployment and Scaling to various platforms

CREATE AND ACCESS DATASETS

PREPROCESS AND TRANSFORM DATA

DEVELOP PREDICTIVE MODELS

ACCELERATE AND DEPLOY

Data sources

Pre-Processing

Import Reference Models/Design from scratch

Desktop Apps

Simulation and augmentation

Transformation

Hardware Accelerated Training

Enterprise Scale Systems

Data Labelling

Feature extraction

Analyze and tune hyperparameters

Embedded Devices and Hardware

Domain-proc data

Collab in AI ecosystem

Challenges – Signals and Time Series

38
Domain-Specific Features and Transformations – Examples

Audio
- Speech Command Recognition
- Voice Activity Detection in Noise
- Denoise Speech
- Classify Gender

Signal
- Music Genre Classification
- Human Activity Recognition
- ECG Signal Classification
- Waveform Segmentation

Time-Series and Text
- Classify Time Series Using Wavelet Analysis
- Sequence-to-Sequence Classification
- Classify Text Data Using LSTMs
- Classify Text Data Using CNNs

Comms and Radar
- Radar Waveform Classification
- Modulation Classification
Call to Action - to be edited by the local team

- Visit the booth
- Attend the talk on Deep Learning and Reinforcement Learning Workflows in A.I.
Back up
Summary - Deep learning workflow in MATLAB

Deep Neural Network Design + Training

- Keras
- TensorFlow
- ONNX
- Caffe
- Idea

Model Exchange

Deep Learning with MATLAB

Trained DNN

Transfer learning

Application design

Application logic

Standalone Deployment

Intel MKL-DNN Library

NVIDIA TensorRT cuDNN Libraries

ARM Compute Library

Coders

- Inception-v3
- AlexNet
- SqueezeNet
- ResNet-101
- GoogLeNet
- ...

MATLAB EXPO 2019