Using Deep Learning to do Real-Time Scoring in Practical Applications

Deep Learning Applications Meetup, Monday, 12/14/2015, Mountain View, CA
http://www.meetup.com/Deep-Learning-Applications/events/227217853/

By Greg Makowski
www.Linkedin.com/in/GregMakowski
greg@LigaDATA.com

Try out kamanja

Liga Data

Community @ http://Kamanja.org

Deep Learning - Outline

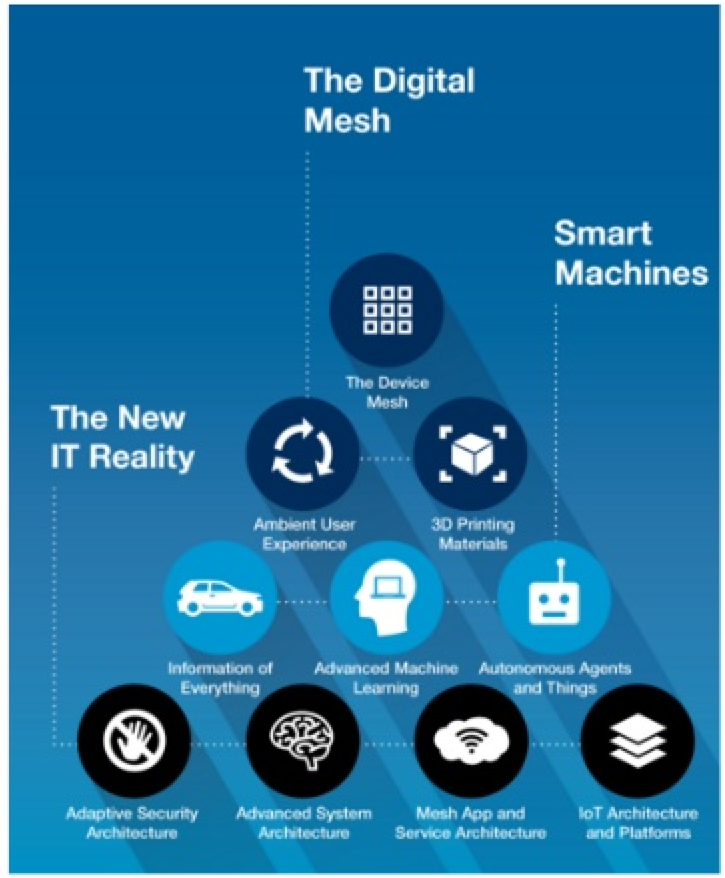
- Big Picture of 2016 Technology
- Neural Net Basics
- Deep Network Configurations for Practical Applications
  - Auto-Encoder (i.e. data compression or Principal Components Analysis)
  - Convolutional (shift invariance in time or space for voice, image or IoT)
  - Real Time Scoring and Lambda Architecture
  - Deep Net libraries and tools (R, H2O, DL4J, TensorFlow, Gorila, Kamanja)
  - Reinforcement Learning, Q-Learning (i.e. beat people at Atari games, IoT)
  - Continuous Space Word Models (i.e. word2vec)

Gartner Identifies the Top 10 Strategic Technology Trends for 2016

http://www.gartner.com/newsroom/id/3143521 Oct 6, 2015

Gartner Tech Trends	Description
Advanced Machine Learning	Deep Neural Nets
Device Mesh	Mobile, wearable, home, auto, IoT
Adaptive Security Architecture	Move from static rules and patterns to understand user and systems
Information of Everything	Contextual, integrated
Ambient User Experience	Over environments, time location
Autonomous Agents and Things	Smart advisors
Advanced System Architectures	Train DNN with GPUs and FPGAs, cloud architectures
Mesh App and Service Architecture	3 tier --> loosely coupled apps and services for web scale performance & flexibility
Internet of Things Platforms	Complements mesh app and service arch, implmenetations of IoT

David Clearley

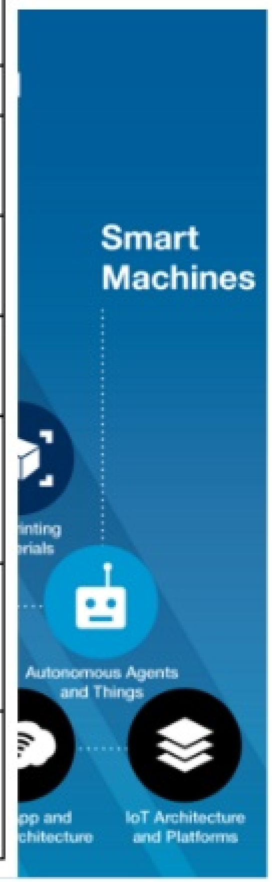


Gartner Identifies the Top 10 Strategic Technology Trends for 2016

http://www.gartner.com/newsroom/id/3143521 Oct 6, 2015

Gartner Tech Trends	Description	Relates to this talk
Advanced Machine Learning	Deep Neural Nets	DNN to solve application needs
Device Mesh	Mobile, wearable, home, auto, IoT	Practical applications, input data
Adaptive Security Architecture	Move from static rules and patterns to understand user and systems	Practical applications
Information of Everything	Contextual, integrated	Input data
Ambient User Experience	Over environments, time location	Output to users (i.e. real time scoring)
Autonomous Agents and Things	Smart advisors	Output to users (i.e. real time scoring)
Advanced System Architectures	Train DNN with GPUs and FPGAs, cloud architectures	Train DNN
Mesh App and Service Architecture	3 tier --> loosely coupled apps and services for web scale performance & flexibility	Application deployment architecture
Internet of Things Platforms	Complements mesh app and service arch, implmenetations of IoT	Input data system integrating with deployment architecture
3D Printing	Expect annual growth rate of 64% for enterprise printers through 2019	n/a

David Clearley



Advantages of a Net
over Regression

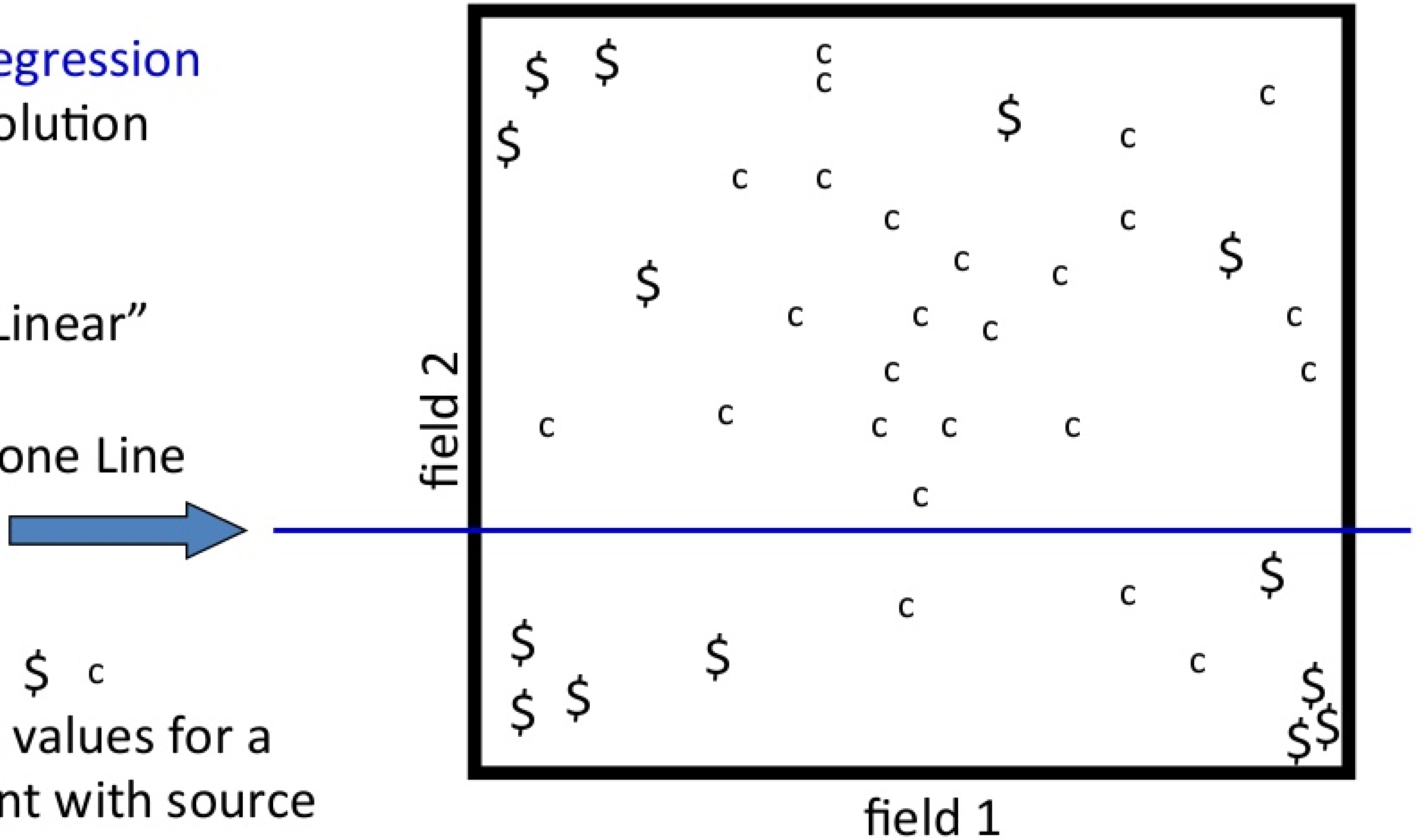
A Regression Solution

"Linear"

Fit one Line

$ c

Target values for a data point with source field values graphed by "field 1" and "field 2"



Showing ONE target field, with values of $ or c
https://en.wikipedia.org/wiki/Regression_analysis

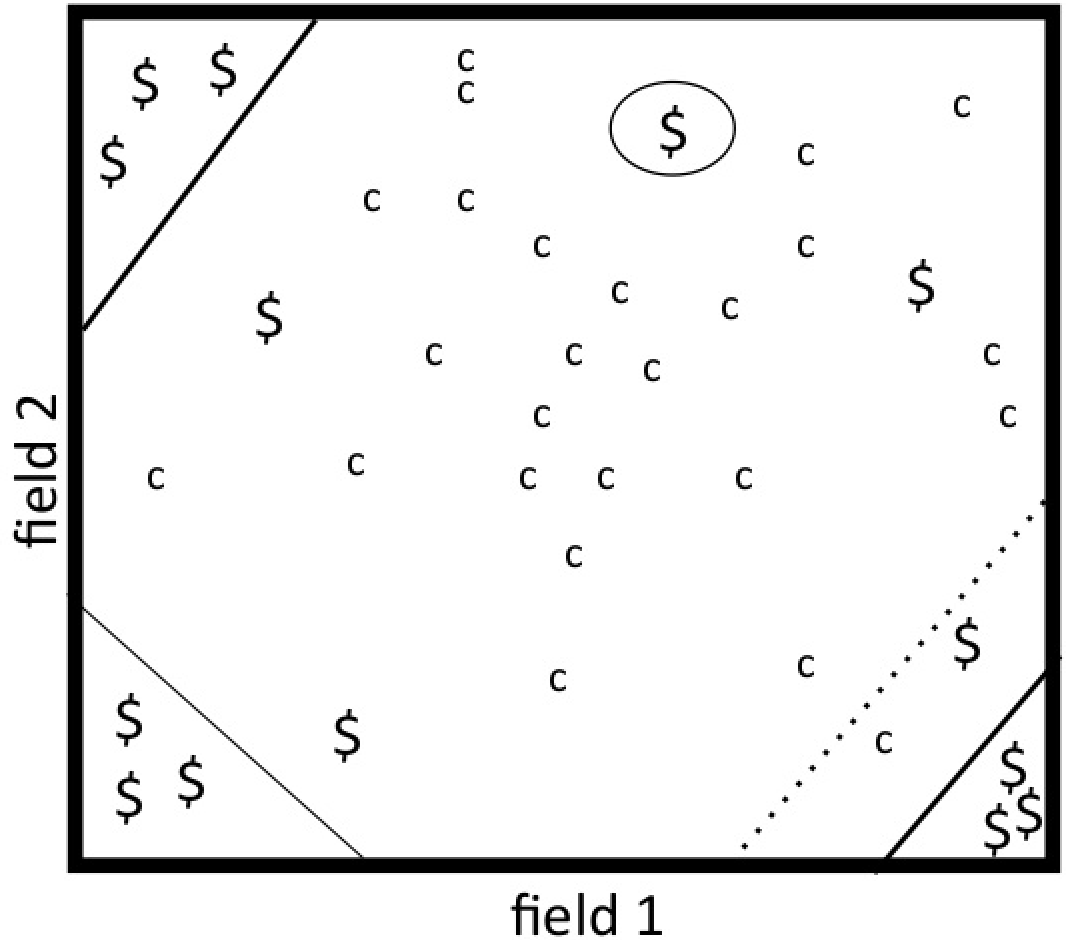
Advantages of a Net
over Regression

A Neural Net Solution

"Non-Linear"

Several regions which are not adjacent

Hidden nodes can be line or circle



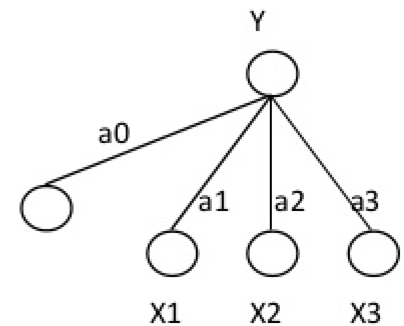
https://en.wikipedia.org/wiki/Artificial_neural_network

A Comparison of a Neural Net and Regression

A Logistic regression formula:

Y = f(a0 + a1*X1 + a2*X2 + a3*X3)

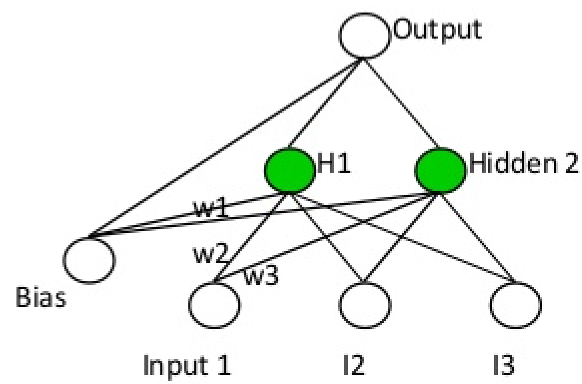
a* are coefficients



Backpropagation, cast in a similar form:

H1 = f(w0 + w1*I1 + w2*I2 + w3*I3)
H2 = f(w4 + w5*I1 + w6*I2 + w7*I3)
:
Hn = f(w8 + w9*I1 + w10*I2 + w11*I3)

O1 = f(w12 + w13*H1 + ... + w15*Hn)
On = ...



w* are weights, AKA coefficients
I1..In are input nodes or input variables.
H1..Hn are hidden nodes, which extract features of the data.
O1..On are the outputs, which group disjoint categories.

Look at ratio of training records v.s. free parameters (complexity, regularization)

Think of Separating Land vs. Water

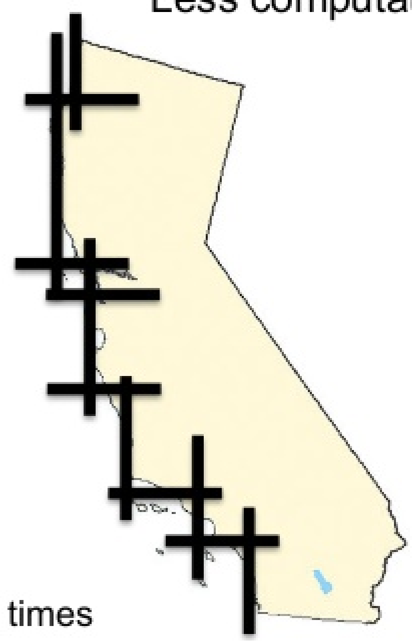
1 line,
Regression
(more errors)

5 Hidden Nodes in
a Neural Network

Different algorithms use different **Basis Functions**:

- One line
- Many horizontal & vertical lines
- Many diagonal lines
- Circles

Decision Tree
12 splits
(more elements,
Less computation)



Q) What is too detailed? "Memorizing high tide boundary" and applying it at all times

Deep Learning - Outline

- Big Picture of 2016 Technology
- Neural Net Basics
- Deep Network Configurations for Practical Applications
  - Auto-Encoder (i.e. data compression or Principal Components Analysis)
  - Convolutional (shift invariance in time or space for voice, image or IoT)
  - Real Time Scoring and Lambda Architecture
  - Deep Net libraries and tools (R, H2O, DL4J, TensorFlow, Gorila, Kamanja)
  - Reinforcement Learning, Q-Learning (i.e. beat people at Atari games, IoT)
  - Continuous Space Word Models (i.e. word2vec)

http://deeplearning.net/
http://www.kdnuggets.com/
http://www.analyticbridge.com/

Leading up to an Auto Encoder

- Supervised Learning
  - Regression, Tree or Net: 50 inputs → 1 output
  - Possible nets:
    - 256 → 120 → 1
    - 256 → 120 → 5 (trees, regressions and most are limited to 1 output)
    - 256 → 120 → 60 → 1
    - 256 → 180 → 120 → 60 → 1 (start getting into training stability problems, with old processes)

- Unsupervised Learning
  - Clustering (traditional unsupervised):
    - 60 inputs (no target); produce 1-2 new (cluster ID & distance)

Auto Encoder (like data compression)
Relate input to output, through compressed middle

- Supervised Learning
  - Regression, Tree or Net: 50 inputs → 1 output
  - Possible nets:
    - 256 → 120 → 1
    - 256 → 120 → 5 (trees, regressions, SVD and most are limited to 1 output)
    - 256 → 120 → 60 → 1
    - 256 → 180 → 120 → 60 → 1 (start getting long training times to stabilize, or may not finish, The BREAKTHROUGH provided by DEEP LEARNING)
- Unsupervised Learning
  - Clustering (traditional unsupervised):
    - 60 inputs (no target); produce 1-2 new (cluster ID & distance)
  - Unsupervised training of a net, assign (target record == input record) AUTO-ENCODING
  - Train net in stages, freezing some connections at different stages
    - 256 → 180 → 256
    - 256 → 180 → 120 → 180 → 256
    - 256 → 180 → 120 → 120 → 120 → 180 → 256
    - 256 → 180 → 120 → 120 → 120 → 120 → 120 → 180 → 256

Because of symmetry,
Only need to update
mirrored weights once

- Add supervised layer to forecast 10 target categories
  - 256 → 180 → 120 → 120 → 120 → 10

https://en.wikipedia.org/wiki/Deep_learning

4 hidden layers w/ unsupervised training
1 layer at end w/ supervised training

Auto Encoder
How it can be generally used to solve problems

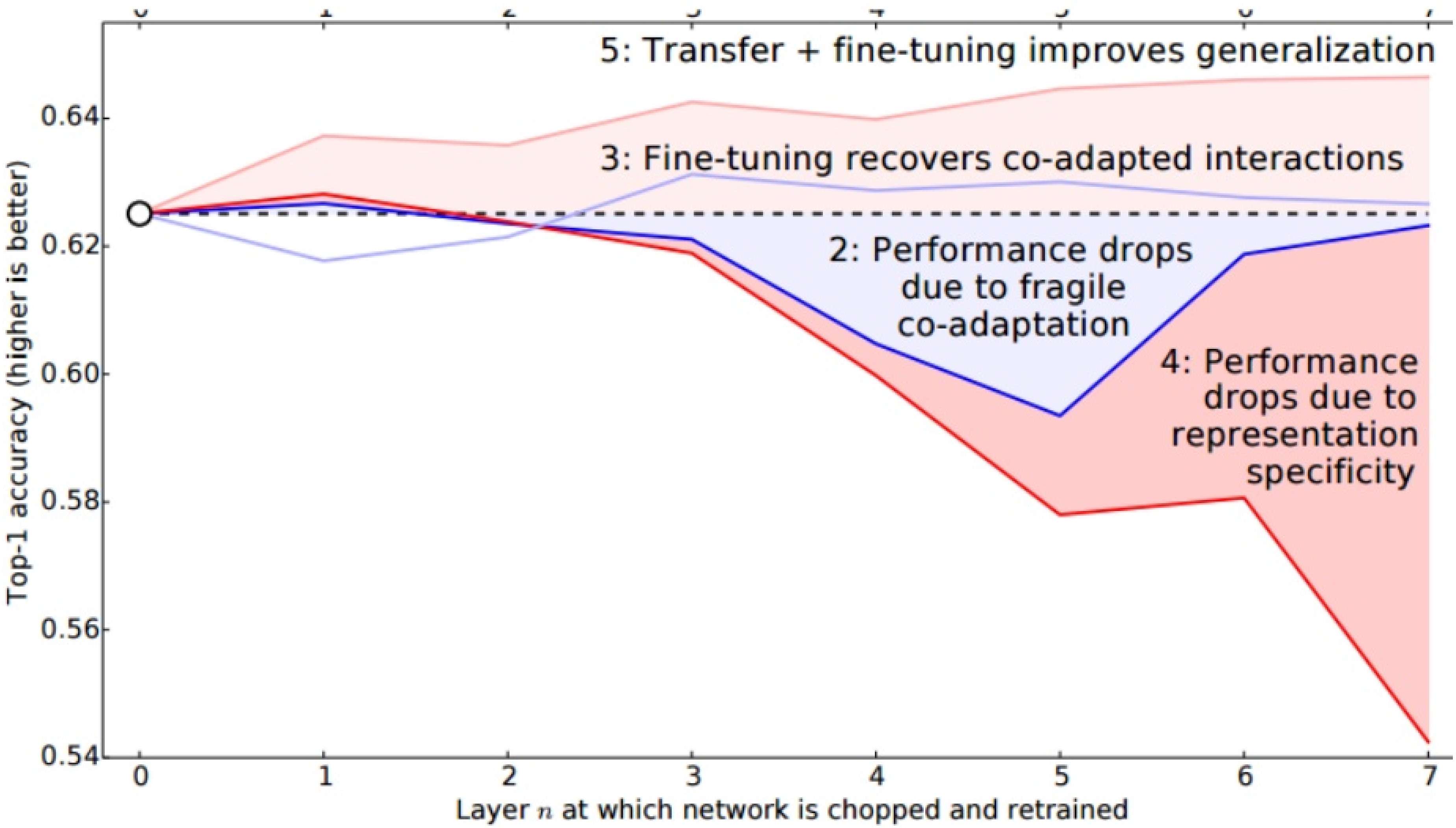
- Add supervised layer to forecast 10 target categories
  - 4 hidden layers trained with unsupervised training, then freeze those weights
  - 1 new layer, trained with supervised learning
    - 256 → 180 → 120 → 120 → 120 → 10

- Outlier detection
    - 256 → 180 → 120 → 120 → 120
      - The "activation" at each of the 120 output nodes indicates the "match" to that cluster or compressed feature
      - When scoring new records, can detect outliers with a process like

        If (max_output_match < 0.333) then suspected outlier

- How is it like PCA?
  - Individual hidden nodes in the same layer are "different" or "orthogonal"

How Transferable are Features in Deep Neural Networks?



http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks.pdf

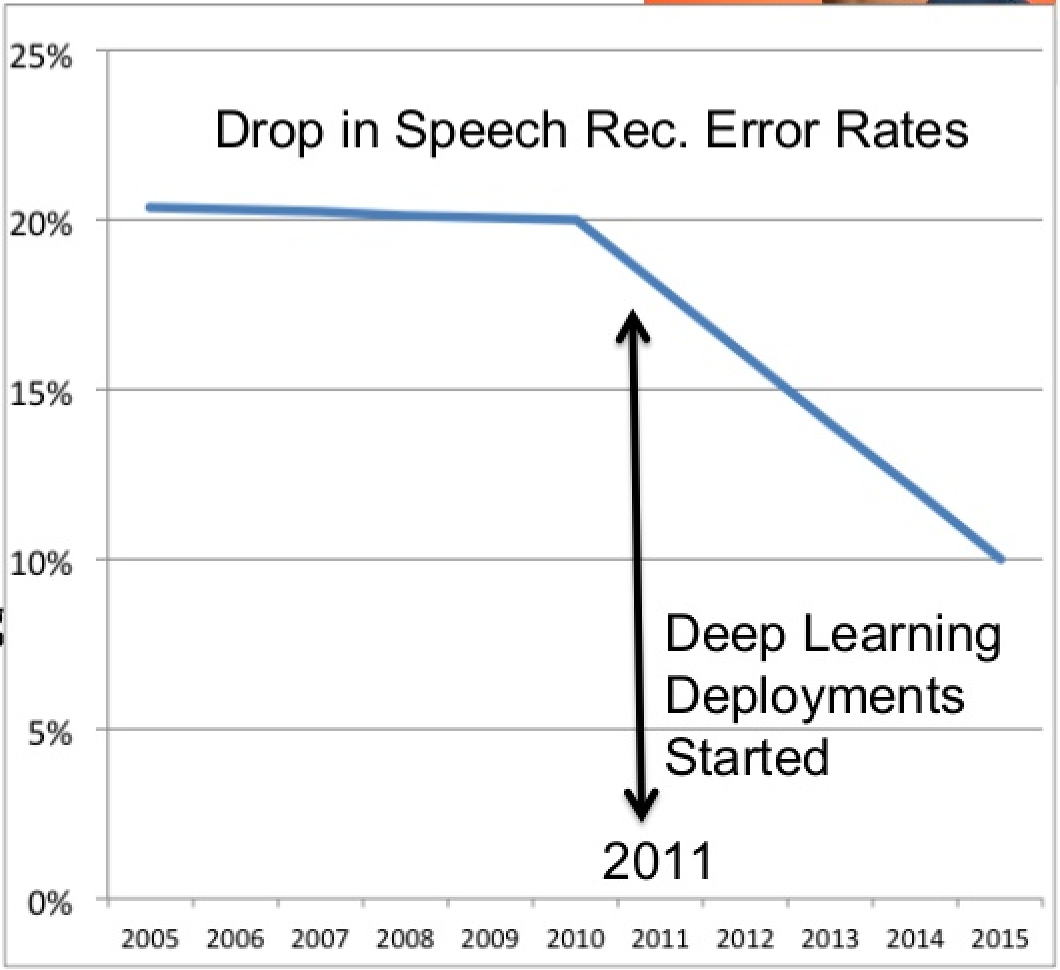
Deep Learning - Outline

- Big Picture of 2016 Technology
- Neural Net Basics
- Deep Network Configurations for Practical Applications
  - Auto-Encoder (i.e. data compression or Principal Components Analysis)
  - Convolutional (shift invariance in time or space for voice, image or IoT)
  - Real Time Scoring and Lambda Architecture
  - Deep Net libraries and tools (R, H2O, DL4J, TensorFlow, Gorila, Kamanja)
  - Reinforcement Learning, Q-Learning (i.e. beat people at Atari games, IoT)
  - Continuous Space Word Models (i.e. word2vec)

Deep Learning Caused a 50% Reduction in Speech recognition error rates in 4 yrs

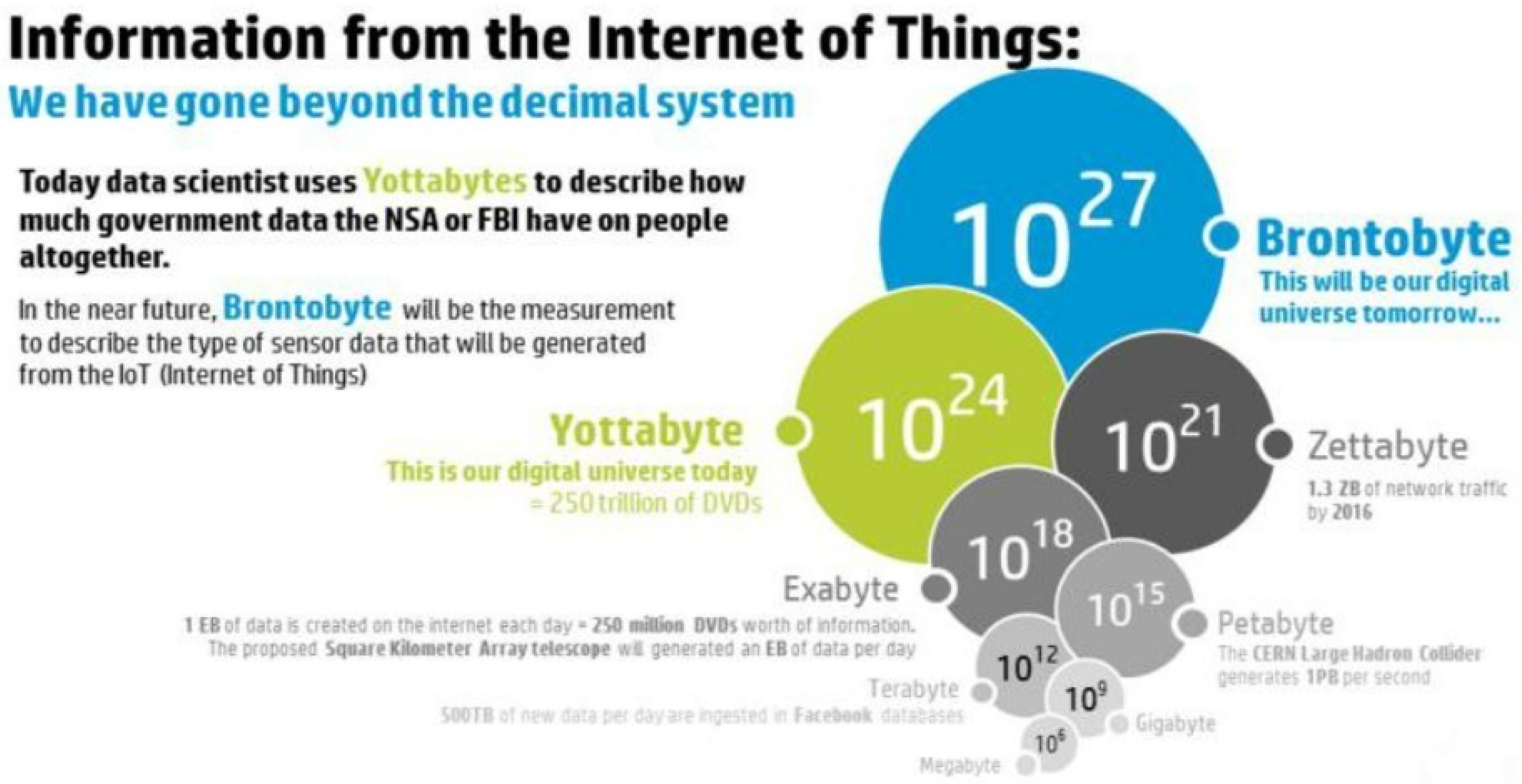
"The use of deep neural nets in production speech systems really started more like in 2011...

I would estimate that from the time before deep neural nets were used until now, the error rate on production speech systems fell from about 20% down to below 10%, so more than a 50% reduction in error rate." - Jeff Dean email to Greg 12/13/2015



http://research.google.com/people/jeff/
Senior Fellow in the Knowledge Group
Google

Internet of Things (IoT) is heavily signal data



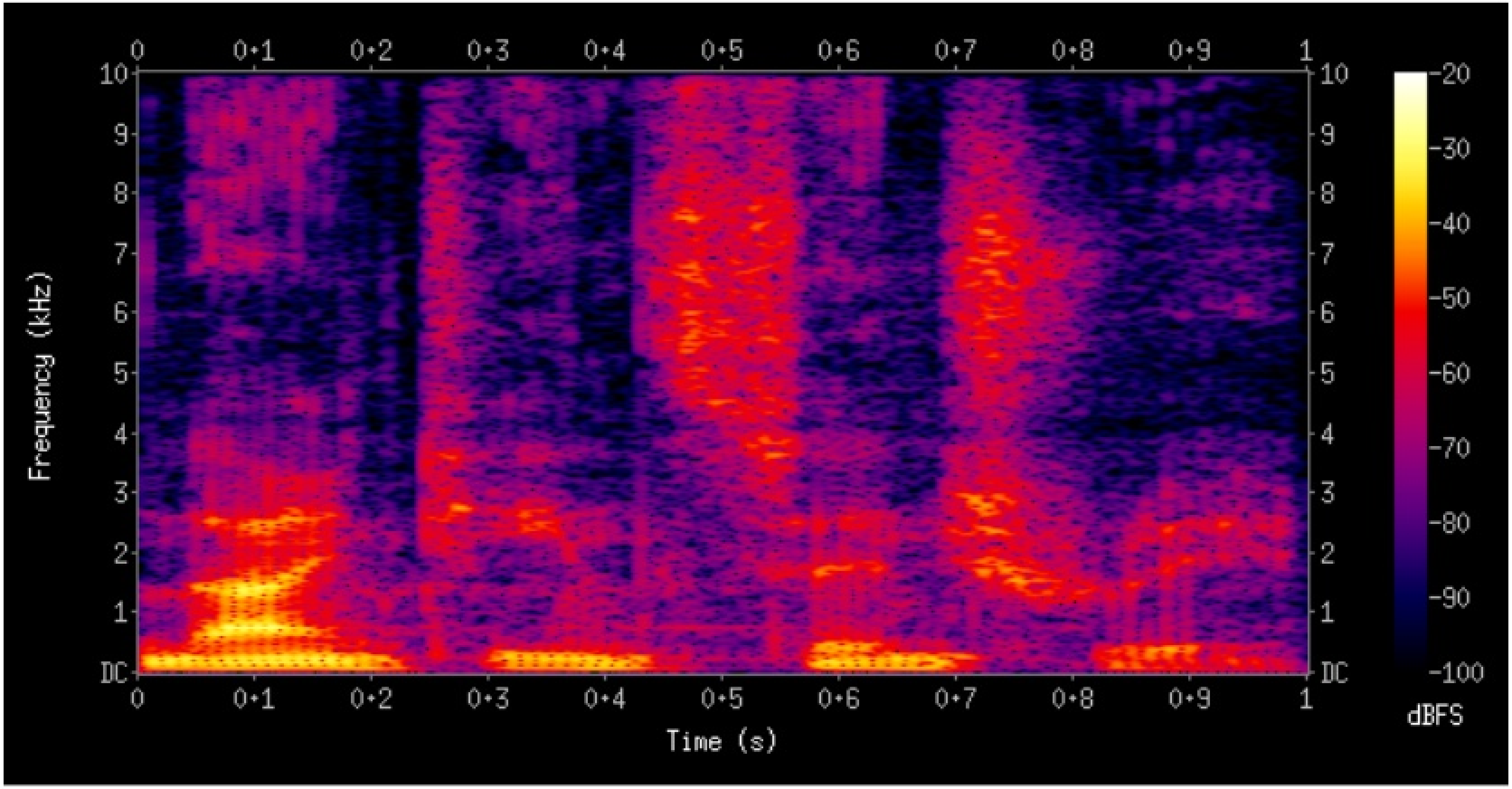
http://www.datasciencecentral.com/profiles/blogs/the-internet-of-things-data-science-and-big-data

Convolutional Neural Net (CNN)
Enables detecting shift invariant patterns

Internet of Things Signal Data

In Speech and Image applications, patterns vary by size, can be shifted right or left
Challenge: finding a bounding box for a pattern is almost as hard as detecting the pat.

Neural Nets can be explicitly trained to provide a FFT (Fast Fourier Transform)
to convert data from time domain to the frequency domain – but typically an explicit FFT is used



Convolutional Neural Net (CNN)
Enables detecting shift invariant patterns

In Speech and Image applications, patterns vary by size, can be shifted right or left
Challenge: finding a bounding box for a pattern is almost as hard as detecting the pat.
Solution: use a siding convolution to detect the pattern
CNN can use very long observational windows, up to 400 ms, long context

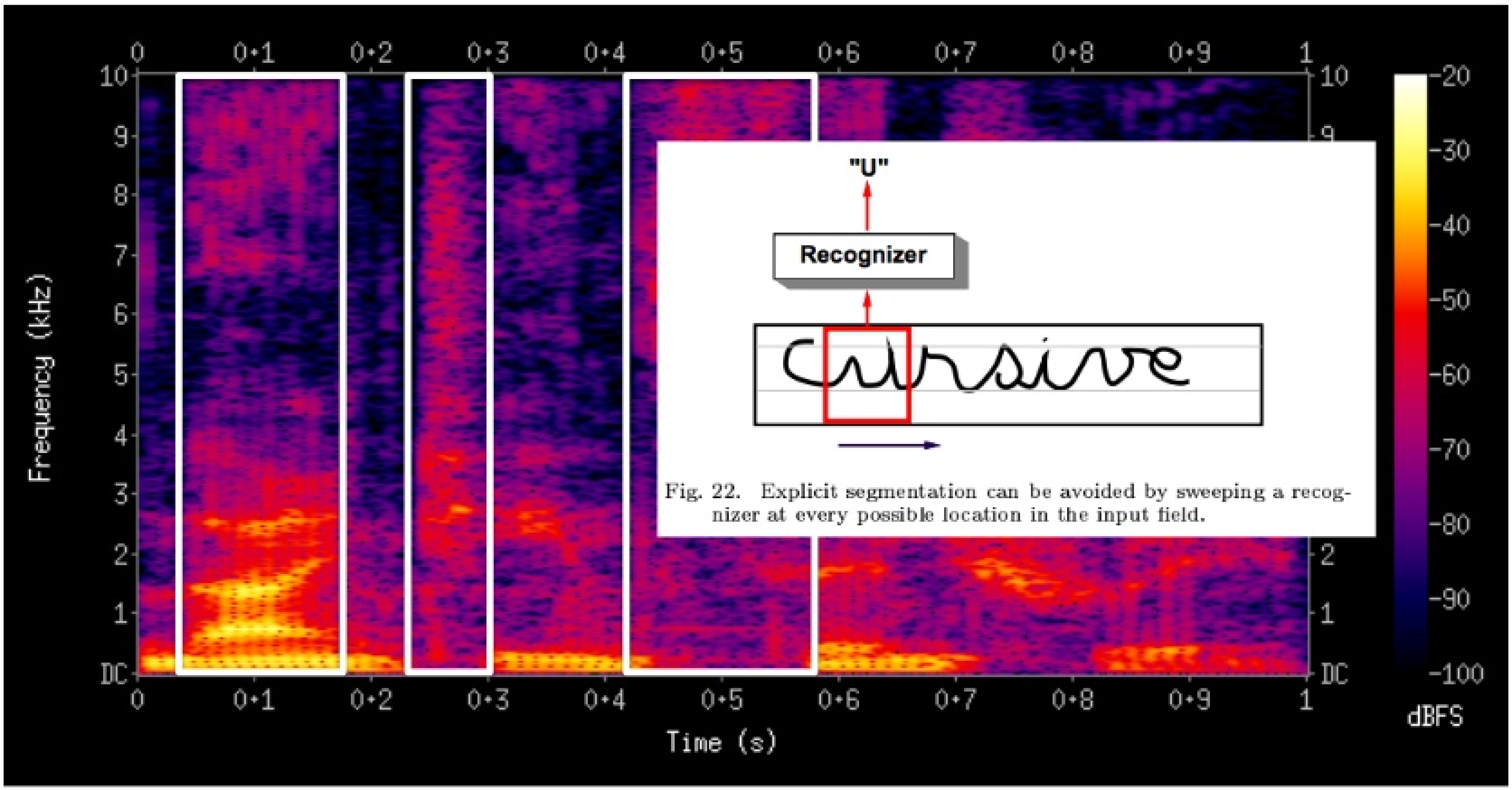
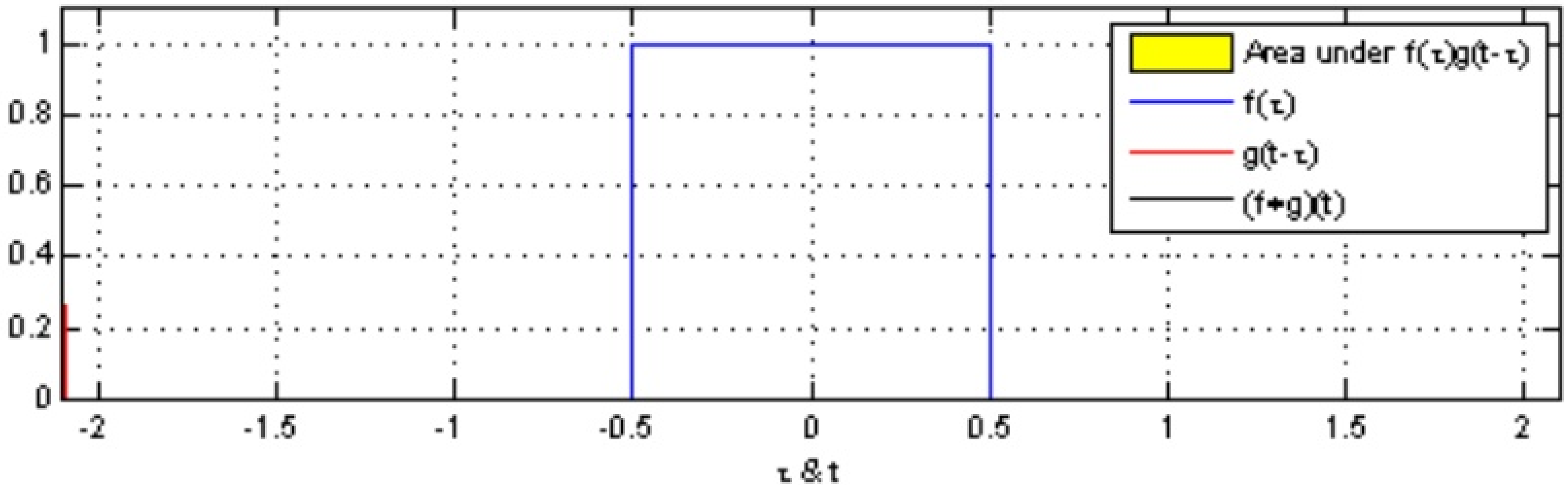
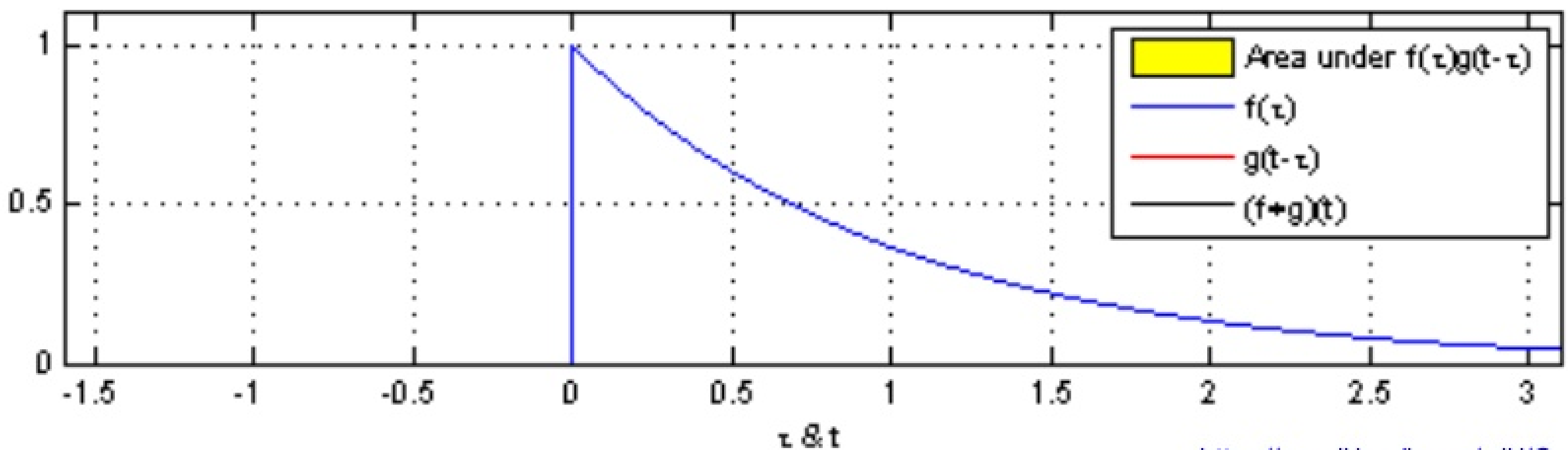


Fig. 22. Explicit segmentation can be avoided by sweeping a recognizer at every possible location in the input field.

Convolution





https://en.wikipedia.org/wiki/Convolution

Convolution Neural Net: from LeNet-5

Director
Facebook, AI Research
http://yann.lecun.com/

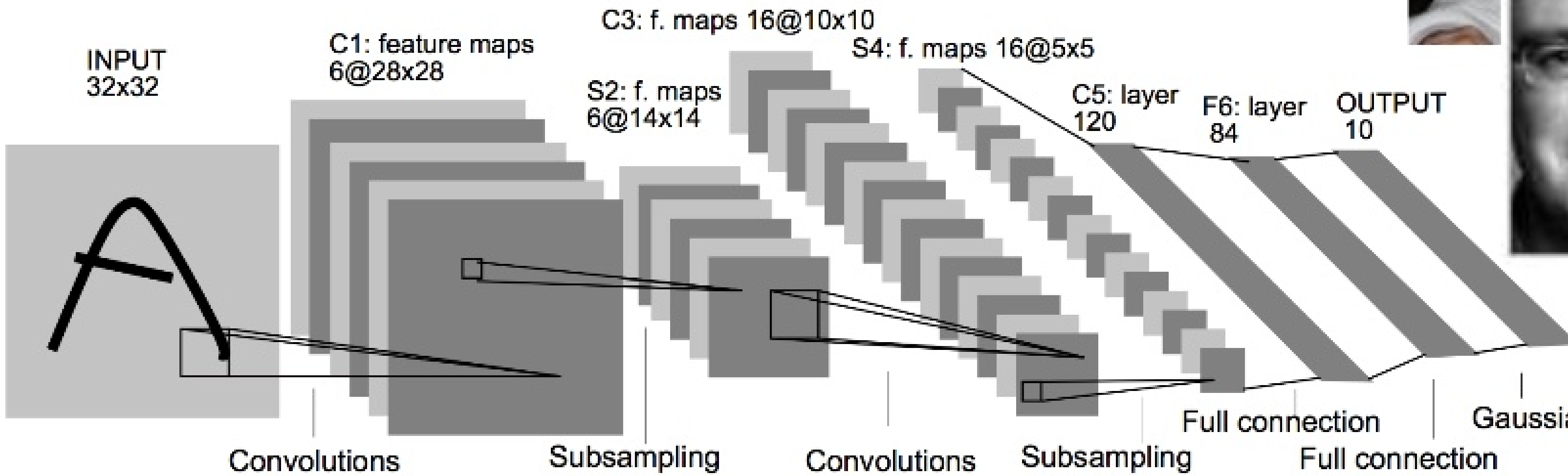


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

Gradient-Based Learning Applied to Document Recognition
Proceedings of the IEEE, Nov 1998
Yann LeCun, Leon Bottou, Yoshua Bengio and Patrick Haffner