From Deep BP-Learning to Internal X-Learning

S.Y. Kung

Princeton
Outline

A. From AI1.0 to AI2.0 to XAI3.0

B. X-Learning: Enhance Accuracy via Model Reduction
   • ITL: node ranking and structural learning
   • DI: Internal Optimization Metrics (IOM)

C. Iterative X-Learning, DNs, & Results
   • NP iteration: Joint PS Learning
   • Deleterious Nodes (DNs)
   • Xnet: MNIST, CIFAR, & ImageNet

D. Extension to Broader Applications
   • LAX-learning, EX-learning
   • Regression Analysis: RX-Learning
   • IMaX-learning (5G-AI)
<table>
<thead>
<tr>
<th>Year</th>
<th>Technology</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1950</td>
<td>BP</td>
<td>Training by Examples</td>
</tr>
<tr>
<td></td>
<td></td>
<td>M. Minsky, first neural network simulator, Princeton Ph. D. 1951.</td>
</tr>
<tr>
<td>1950</td>
<td>AI 1.0</td>
<td>Knowledge Systems Laboratory, (1970 Feigenbaum)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rule-based</td>
</tr>
<tr>
<td>2000</td>
<td>NN 1.0 (MLP)</td>
<td>⇒</td>
</tr>
<tr>
<td>2000</td>
<td>NN 2.0 (DNN)</td>
<td>Deep BP-Learning</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MIT AI Lab (1958, M. Minsky)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>⇒</td>
</tr>
<tr>
<td>2020</td>
<td>Xnet</td>
<td>Internal X-Learning</td>
</tr>
<tr>
<td></td>
<td></td>
<td>⇒</td>
</tr>
<tr>
<td></td>
<td>AI 2.0</td>
<td>⇒</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data-driven</td>
</tr>
<tr>
<td></td>
<td></td>
<td>⇒</td>
</tr>
<tr>
<td></td>
<td>XAI 3.0</td>
<td>⇒</td>
</tr>
</tbody>
</table>
AI2.0: DLN = Feature Engineering + Label Engineering

NN2.0 = MLP + ConvNet + Deep BP

Learning, Memorization vs. Generalization... doing BP on thousands (if not millions) of parameters

Deep learning networks represent the state-of-the-arts of ML.
From Rule-based AI1.0 to Data-Driven AI2.0

ML ≅ AI

NN ≅ DL

NN2.0 ↔ AI2.0

 nature
THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE
Characterization of Convolutional Neural Nets

\[
\{28^2 \rightarrow [f_1 \ n_1 \ m_1 \ \mu_1] \rightarrow [f_2 \ n_2 \ m_2 \ \mu_2]\} - 800 - 500 - 10
\]

LeNet-5[Han15]: \(\{28^2 \rightarrow [5 \ 20 \ 24 \ 12] \rightarrow [5 \ 50 \ 8 \ 4]\} - 800 - 500 - 10\)

X-learning

\[24 = 28 - 5 + 1\]
\[8 = 12 - 5 + 1\]
\[800 = 50 \times 4^2\]
From Deep BP-Learning to Internal X-Learning

Back-propagation (BP) is only for parameter learning.

X-learning adopts internal & Explainable Learning (Xnet), going beyond BP to train both NN’s structure and parameters.

DARPA XAI ⇒ X-Learning
Motivated by XAI, X-learning will play vital roles in XAI.
Biology for Internal X-Learning
Why Internal X-Learning?

Bird

or Drone
B. X-Learning

X-Learning facilitates Both Parameter/Structure Learning!!

Parameter Learning:
EOM Gradient Descent

Structural Learning:
IOM Guided Adaptation

Compute $\frac{\partial L}{\partial w_1}, \frac{\partial L}{\partial w_2}$

Structural characterization:
- The number of layers in the model.
- The number of nodes in each layer.
A quantum jump from finite to zero:

- individually (connection)
- or collectively (node).

X-learning $\Rightarrow$ Structural Gradient Optimization Algorithm $\Rightarrow$ LASSO (Link/Node)
B(i): Why X-Learning?

Biological Motivation

Synaptic Density

At birth

6 year old

14 year old

Source: Rethinking the Brain, Families and Work Institute, Rima Shore, 1997; Founders Network slide
Biological Justification of Neuron Pruning

Human brain prunes more than 5000 brain neurons daily.
Evaluation of Links Vs. Evaluation of Neurons

Reduction Effectiveness

Accuracy
B(ii): ITL and IOM

For BP learning, all neuron are created equal.

Goal: Ranking of Neurons to facilitate structural gradient.

Internal X-learning for node ranking and structural learning!!
Local Metrics for classification-type

The internal learning paradigm facilitates structural learning.

(1) Internal Teacher Labels (ITL)
For classification problem, the internal teacher labels can be metaphorically hidden in “Trojan-horses” and transported (along with the data) from the input layer to all hidden nodes. promising ones:

(2) Internal Optimization Metrics (IOM)
Local Metrics for classification-type
Nonlinear transformations in the neural network is meant to improve the IOM (e.g. DI) from layer to layer.
ETL is space-divided, the labels correspond to the output nodes.

In contrast, ITL is time-divided, the labels are assigned to each of incoming input samples at one instance.
ETL is space-divided, the labels correspond to the output nodes.
B(iii): Internal Optimization Metrics (IOM)
Standing on Giant’s Shoulders

Carl Friedrich Gauss – Born 1777
The numerous things named in honor of Gauss include: The normal distribution, also known at the Gaussian distribution, the most common bell curve in statistics.

Sir Ronald Aylmer Fisher FRS (1890 –1962) was a British statistician and "a genius who almost single-handedly created the foundations for modern statistical science" and "the single most important figure in 20th century statistics".

Claude Elwood Shannon was "the father of information theory". Shannon is noted for having founded information theory with a landmark paper, A Mathematical Theory of Communication 1948.
DI (Discriminant Information)

Three Scatter Matrices

**Scatter Matrix**
\[ \bar{S} = \bar{X}\bar{X}^T = \sum_{i=1}^{N} [x_i - \bar{\mu}] [x_i - \bar{\mu}]^T \]

**Between Class Scatter Matrix**
\[ S_B = \sum_{\ell=1}^{L} N_\ell \left[ \bar{\mu}_\ell - \bar{\mu} \right] \left[ \bar{\mu}_\ell - \bar{\mu} \right]^T = \Delta \Xi \Delta^T \]

**Within Class Scatter Matrix**
\[ S_W = \sum_{\ell=1}^{L} \sum_{i=1}^{N_\ell} \left[ x_{j}^{(\ell)} - \bar{\mu}_\ell \right] \left[ x_{j}^{(\ell)} - \bar{\mu}_\ell \right]^T \]

C.R. Rao vs. S.Y. Kung

\[ \text{DI} = \text{DI}(I) = \text{trace} \left( \left[ \bar{S} + \rho I \right]^{-1} S_B \right) \]
DI (Discriminant Information)

Numerically, the optimal DCA projection matrix can be derived from the principal eigenvectors of the Discriminant Matrix.

$$\text{DI} = \text{DI}(I) = \text{trace} \left( \tilde{S} + \rho I \right)^{-1} S_B \right)$$

A vital condition is that there exists no inter-component redundancy, a condition termed by C.R. Rao as "Canonical Orthogonality". (For DCA, all the eigenvalues are generically distinct, therefore, all the columns of $V$ are canonically orthogonal to each other.)

C.R. Rao vs. S.Y. Kung

- **Rao**: the final score is equal to the product of individual scores, i.e. the volumetric score.
- **Kung**: the total score is equal to the sum of individual scores.
Suppose a hidden layer has 2 nodes (or more):

Low-DI Space (2 nodes) → NN → High-DI Space (2 nodes)
C. Iterative X-Learning, DNs, & Results

• DI with SOTP Property

• NP iteration: Joint PS Learning

• Deleterious Nodes (DNs)

• Xnet: MNIST, CIFAR, & ImageNet
B(iii): Internal Optimization Metrics (IOM)

**Beauty of Math:** A simple change (from **determinant** to **trace**) leads to

\[ \text{total} = \text{sum of individuals} \]

which then bears offer a vital monotonic increasing property, crucial for using DI to rank the nodes and subspace.
C(i): Subspace/Component DI

\[ \text{DI} = \text{DI}(I) = \text{tr} \left( (\bar{S} + \rho I)^{-1} S_B \right) \]

Subspace DI:

\[ \text{DI}(W) = \text{tr} \left( W^T W \right) \]

\( \rho \): variance of additive noise

Three Scatter Matrices

**Scatter Matrix**

\[ \bar{S} = \bar{X} \bar{X}^T = \sum^N [x_i - \bar{\mu}] [x_i - \bar{\mu}]^T \]

**Between Class Scatter Matrix**

\[ S_B = \sum_{\ell=1}^{L} N_{\ell} [\bar{\mu}_\ell - \bar{\mu}] [\bar{\mu}_\ell - \bar{\mu}]^T = \Delta \Xi \Delta^T \]

**Within Class Scatter Matrix**

\[ S_W = \sum_{\ell=1}^{L} \sum_{j=1}^{N_{\ell}} \sum_{j=1}^{N_{\ell}} [x_j^{(\ell)} - \bar{\mu}_\ell] [x_j^{(\ell)} - \bar{\mu}_\ell]^T \]
To assess the IOM of the full space of a layer, we set $\mathbf{W} = \mathbf{I}$:

For (supervised) deep compression, we adopt $\mathbf{W}_{i\text{-keep}} / \mathbf{W}_{i\text{-drop}}$ to keep/drop only the $i$-th node/channel:

$$
\text{Subspace DI}
$$

$$
\text{DI}(\mathbf{W}) = \text{tr}\left((\mathbf{W}^T \mathbf{S} \mathbf{W} + \rho \mathbf{I})^{-1} \mathbf{W}^T \mathbf{S}_B \mathbf{W}\right)
$$
Compute $\frac{\partial L}{\partial w_1}, \frac{\partial L}{\partial w_2}$.
NP-Iterative Pruning Method

DI-based Pruning Method Offers Win-Win Deep Compression/Quantization
C(ii): Performance Enhancement via Model Reduction

DN= Deleterious neurons

- Constructive (discriminative) nodes
- Redundant (but robust) nodes
- *Deleterious nodes*

Theme: Finding/Pruning *deleterious neurons* enhances performance.
X-learning adopts a joint parameter/structural gradient learning to gradually reduce the network towards an optimal structure, while traversing across the winning structural space.
Differential-DI Subspace  (2 nodes: left-vs-right, top-down-neuron)

High-DI: left-vs-right neuron

Low-DI: top-down neuron

Our punchline:
\[ 1 + 1 > 2 \quad 2 - 1 > 2 \]
Visualization via Channel Images [KHL19]

DI metric is used to determine which channels to prune.
Deleterious neurons: Pruning enhances robustness

By simply removing redundant and harmful nodes, one can enhance the model robustness. This leads to a joint parameter/structural learning paradigm.
C(iii): Experimental Results

- MNIST, CIFAR Datasets
- ImageNet
- LPIRC
(c) CIFAR10: VGG16 speedup

<table>
<thead>
<tr>
<th>Task</th>
<th>Models</th>
<th>Accuracy %</th>
<th>FLOPs</th>
<th>Params</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VGG-16 (baseline)</td>
<td>93.25</td>
<td>6.26x10^8</td>
<td>1.5x10^7</td>
</tr>
<tr>
<td></td>
<td>VGG-16 (Li et al., 2017)</td>
<td>93.41</td>
<td>4.12x10^8 (65.81%)</td>
<td>5.4x10^6 (36%)</td>
</tr>
<tr>
<td></td>
<td>VGG-16 (FDR)</td>
<td>93.61</td>
<td>1.35x10^8 (21.49%)</td>
<td>7.1x10^5 (4.7%)</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>VGG-16 (DILoss)</td>
<td>94.07</td>
<td>1.28x10^8 (20.45%)</td>
<td>5.32x10^5 (3.55%)</td>
</tr>
<tr>
<td></td>
<td>ResNet-56 (baseline)</td>
<td>93.04</td>
<td>2.5x10^8</td>
<td>8.5x10^5</td>
</tr>
<tr>
<td></td>
<td>ResNet-56 (Li et al., 2017)</td>
<td>93.06</td>
<td>1.81x10^8 (72.4%)</td>
<td>7.3x10^5 (85.88%)</td>
</tr>
<tr>
<td></td>
<td>ResNet-56 (Yu et al., 2018)</td>
<td>93.01</td>
<td>1.41x10^8 (56.4%)</td>
<td>4.94x10^5 (58.12%)</td>
</tr>
<tr>
<td></td>
<td>ResNet-56 (He et al., 2017)</td>
<td>91.9</td>
<td>1.25x10^8 (50%)</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>ResNet-56 (Zhuang et al., 2018)</td>
<td>93.49</td>
<td>1.25x10^8 (50.25%)</td>
<td>4.3x10^5 (50.76%)</td>
</tr>
<tr>
<td></td>
<td>ResNet-56 (DILoss)</td>
<td>93.84</td>
<td>8.38x10^7 (33.52%)</td>
<td>3.12x10^5 (35.52%)</td>
</tr>
<tr>
<td></td>
<td>ResNet-56 (bootstrap/DILoss+Zhuang)</td>
<td>93.84</td>
<td>7.58x10^7 (30.32%)</td>
<td>2.81x10^5 (33.05%)</td>
</tr>
</tbody>
</table>
MINIST: Speedup & Storage

### Table: Accuracy, FLOPs, and Params

<table>
<thead>
<tr>
<th>Task</th>
<th>Models</th>
<th>Accuracy %</th>
<th>FLOPs</th>
<th>Params</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td><em>Lenet-5 (baseline)</em></td>
<td>99.2</td>
<td>$4.59 \times 10^6$</td>
<td>$4.3 \times 10^5$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-5 (Han et al., 2015)</em></td>
<td>99.23</td>
<td>$8.3 \times 10^5 (18.1%)$</td>
<td>$3.6 \times 10^4 (8.4%)$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-5 (Louizos et al., 2018)</em></td>
<td>99</td>
<td>$7.85 \times 10^5 (17.1%)$</td>
<td>$1.22 \times 10^4 (2.83%)$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-5 (FDR)</em></td>
<td>99.33</td>
<td>$2.6 \times 10^5 (5.74%)$</td>
<td>$4.9 \times 10^3 (1.1%)$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-5 (DILoss)</em></td>
<td><strong>99.35</strong></td>
<td><strong>$2.46 \times 10^5 (5.36%)$</strong></td>
<td><strong>$3.86 \times 10^3 (0.89%)$</strong></td>
</tr>
<tr>
<td></td>
<td><em>Lenet-300 (baseline)</em></td>
<td>98.36</td>
<td>-</td>
<td>$2.7 \times 10^5$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-300 (Han et al., 2015)</em></td>
<td>98.41</td>
<td>-</td>
<td>$2.24 \times 10^4 (8.3%)$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-300 (Louizos et al., 2018)</em></td>
<td>98.2</td>
<td>-</td>
<td>$2.7 \times 10^4 (10%)$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-300-100 (FDR)</em></td>
<td>98.42</td>
<td>-</td>
<td>$2.3 \times 10^4 (8.5%)$</td>
</tr>
<tr>
<td></td>
<td><em>Lenet-300 (DILoss)</em></td>
<td><strong>98.46</strong></td>
<td>-</td>
<td>$1.63 \times 10^4 (6.04%)$</td>
</tr>
</tbody>
</table>
Basic Xnet: Layer-Uniform (Layer-Independent) ITL

CIFAR-100: Speedup

<table>
<thead>
<tr>
<th>Task</th>
<th>Models</th>
<th>Accuracy %</th>
<th>FLOPs</th>
<th>Params</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIFAR100</td>
<td>Mobilnet-v2 (baseline)</td>
<td>73.68</td>
<td>1.8×10^8</td>
<td>2.4×10^6</td>
</tr>
<tr>
<td></td>
<td>Mobilnet-v2 (DILoss)</td>
<td>75.61</td>
<td>7.57×10^7 (42.06%)</td>
<td>1.07×10^6 (44.58%)</td>
</tr>
</tbody>
</table>
X-learning can spot

- Many a Redundant (but robust) nodes
- Very few Highly Deleterious nodes
### Speed Winner (power, energy, latency)

<table>
<thead>
<tr>
<th>Model</th>
<th>Top-1 Acc.</th>
<th>FLOPs</th>
<th>Param.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MobileNetV1</td>
<td>70.2%</td>
<td>569M</td>
<td>4.24M</td>
</tr>
<tr>
<td>WM (Sandler et al., 2018)</td>
<td>68.4%</td>
<td>325M</td>
<td>-</td>
</tr>
<tr>
<td>NetAdapt (Yang et al., 2018)</td>
<td>69.1%</td>
<td>285M</td>
<td>-</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td><strong>70.2%</strong></td>
<td><strong>285M</strong></td>
<td><strong>2.28M</strong></td>
</tr>
<tr>
<td>MobileNetV2</td>
<td>71.8%</td>
<td>300M</td>
<td>3.47M</td>
</tr>
<tr>
<td>WM-1 (Sandler et al., 2018)</td>
<td>69.8%</td>
<td>210M</td>
<td>2.61M</td>
</tr>
<tr>
<td><strong>Ours-1</strong></td>
<td><strong>70.8%</strong></td>
<td><strong>210M</strong></td>
<td><strong>2.33M</strong></td>
</tr>
<tr>
<td>1st place (LPIRC Challenge, 2018)</td>
<td><strong>65.2%</strong></td>
<td><strong>186M</strong></td>
<td>-</td>
</tr>
<tr>
<td>WM-2 (Sandler et al., 2018)</td>
<td>65.4%</td>
<td>170M</td>
<td>2.57M</td>
</tr>
<tr>
<td>ThiNet (Luo et al., 2017)</td>
<td>65.44%</td>
<td>170M</td>
<td>2.57M</td>
</tr>
<tr>
<td>DCP (Zhuang et al., 2018)</td>
<td>65.91%</td>
<td>170M</td>
<td>2.57M</td>
</tr>
<tr>
<td><strong>Ours-2</strong></td>
<td><strong>68.2%</strong></td>
<td><strong>170M</strong></td>
<td><strong>1.9M</strong></td>
</tr>
</tbody>
</table>

### Accuracy Winner

<table>
<thead>
<tr>
<th>Model</th>
<th>Top-1 Acc.</th>
<th>FLOPs</th>
<th>Param.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet-50</td>
<td>75.15%</td>
<td>4.09B</td>
<td>25.6M</td>
</tr>
<tr>
<td><strong>Ours-1</strong></td>
<td><strong>76.34%</strong></td>
<td><strong>2.63B</strong></td>
<td><strong>17M</strong></td>
</tr>
<tr>
<td>SSS (Huang et al., 2018)</td>
<td>71.82%</td>
<td>2.33B</td>
<td>15.6M</td>
</tr>
<tr>
<td>ThiNet (Luo et al., 2017)</td>
<td>72.04%</td>
<td>2.44B</td>
<td>16.94M</td>
</tr>
<tr>
<td>GDP (Lin et al., 2018)</td>
<td>72.61%</td>
<td>2.24B</td>
<td>-</td>
</tr>
<tr>
<td>SFP (He et al., 2018)</td>
<td>74.61%</td>
<td>2.42B</td>
<td>-</td>
</tr>
<tr>
<td><strong>Ours-2</strong></td>
<td><strong>76.1%</strong></td>
<td><strong>2.31B</strong></td>
<td><strong>16.72M</strong></td>
</tr>
<tr>
<td>CP (He et al., 2017)</td>
<td>72.26%</td>
<td>2B</td>
<td>-</td>
</tr>
<tr>
<td>DCP (Zhuang et al., 2018)</td>
<td>74.95%</td>
<td>2B</td>
<td>-</td>
</tr>
<tr>
<td><strong>Ours-3</strong></td>
<td><strong>75.6%</strong></td>
<td><strong>2B</strong></td>
<td><strong>15.6</strong></td>
</tr>
</tbody>
</table>
D. Extension

- **LAXnet**: Adaptive (Layer-dependent) ITL
  - From coarse ITL to fine ITL
- **EXnet**: End-User Explainable Learning ITL
  - \( \text{DI}(A) \) for \( A \) or \( \tilde{A} \) (=not \( A \))
  - \( \text{DI}(A) \) and \( \text{DI}(A') \) for a special-interest group such as:
    - \( \tilde{A} \) but \( A' \)
    - \( A \) but \( \tilde{A}' \)
- **Regression/Generation**
  - Super-Resolution Image Enhancement
End User Explainability: “XAI is most interested in explanations of higher-level decisions that would be relevant to the end user and the missions he/she needs to manage.”[DARPA2016]

Motivated by XAI, X-learning will play vital roles in XAI.
Our Approach to XAI: EXnet

Bird

or Drone
EXnet’s Approach to XAI

ABC Example

End-user: (subset = top 3 ABC classes):

A= airplane,  B= bird,  and  C= car

From Low DI to High DI channels (neurons)
LAX-Learning: Different Teachers for Different Layers

Layer Structure in (Optical Preprocessing) Retina Nervous Systems
Multi-resolution LAX-Learning

Initial tantalizing layers  Final maturing layers

Initially: 3-class ITL

Finally: 6-class ITL

course explainability
(1/3 ITL)

fine explainability
(1/6 ITL)
Multi-resolution X-Learning

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>FLOPs</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet-164 (Baseline)</td>
<td>76.63%</td>
<td>505M</td>
<td>1.7M</td>
</tr>
<tr>
<td>ResNet-164 (Liu et al., 2017)</td>
<td>76.09%</td>
<td>247M</td>
<td>1.2M</td>
</tr>
<tr>
<td>X-ResNet-164 (Uni-resolution)</td>
<td>76.11%</td>
<td>210M</td>
<td>0.95M</td>
</tr>
<tr>
<td>X-ResNet-164 (multi-resolution)</td>
<td>76.96%</td>
<td>210M</td>
<td>0.95M</td>
</tr>
</tbody>
</table>

AXnet improves the accuracy by **0.85%** over the basic Xnet.
Regression Analysis: RX-Learning

JPEGX Examples on Super-Resolution Enhancement/Restoration
X Learning for 5G Inter-machine Intelligence

Transferable Learning

GROWTH IN THE INTERNET OF THINGS
THE NUMBER OF CONNECTED DEVICES WILL EXCEED 50 BILLION BY 2020

BILLIONS OF DEVICES

2016 22.9B
2015 18.2B
2014 14.2B
2013 11.2B
2012 8.7B
2009 IoT INCEPTION
2003 0.5B
1992 1M
1996
2000
2004
2008
2012
2018 34.8B
2019 42.1B
2020 50.1B

1M AI machines within 1 km²
IMaX inter-machine learning (5G-AI)

IMaX-Learning: parameter transfer vs. structure transfer

Structural Transfer Learning: ResNet164 on CIFAR (CIFAR10-to-CIFAR100)

Transferable pruning policy: 76.8%
Automation of X-Learning

Reduction/Trimming of Deep Learning Networks

Automatic Bigdata Deep Compiler (ABDC)
Thank You!!