IJCAI-ECAI-18

Online Deep Learning
Learning Deep Neural Networks on the Fly

Doyen Sahoo

School of Information Systems
Singapore Management University

Joint work with

Quang Pham, Jing Lu, Steven C. H. Hoi

SMU Singapore Management University

LARC Living Analytics Research Centre

Strategic Partner:
Carnegie Mellon University

Overview

Introduction & Motivation
- Online Learning and Deep Learning
- Challenges in using Deep Networks for Online Learning

Online Deep Learning (ODL)
- Shallow to Deep Principle
- Architecture for ODL
- Hedge Backpropagation

Experiments
- Online Performance
- Other Insights

Introduction & Motivation

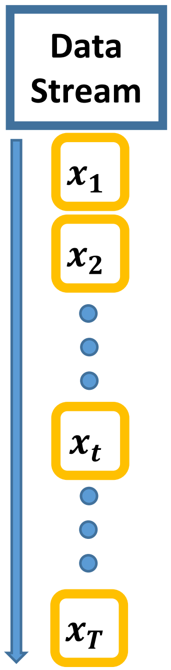
Online Learning and Deep Learning

Online Learning

Introduction & Motivation

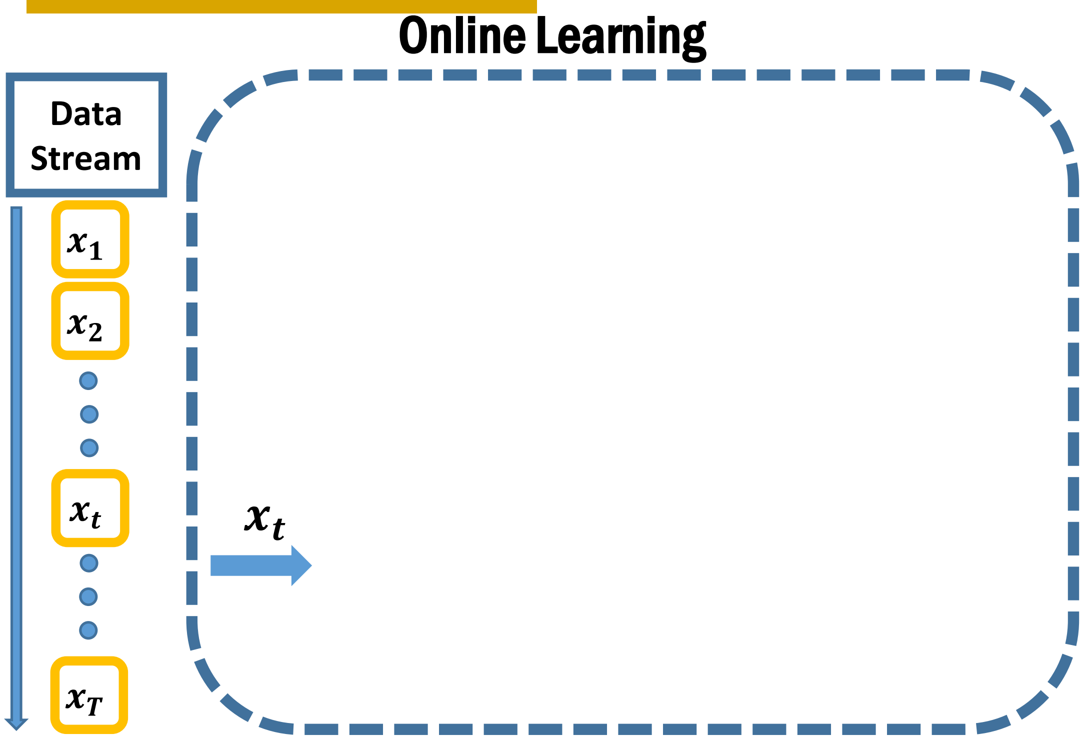
Online Learning and Deep Learning

Online Learning



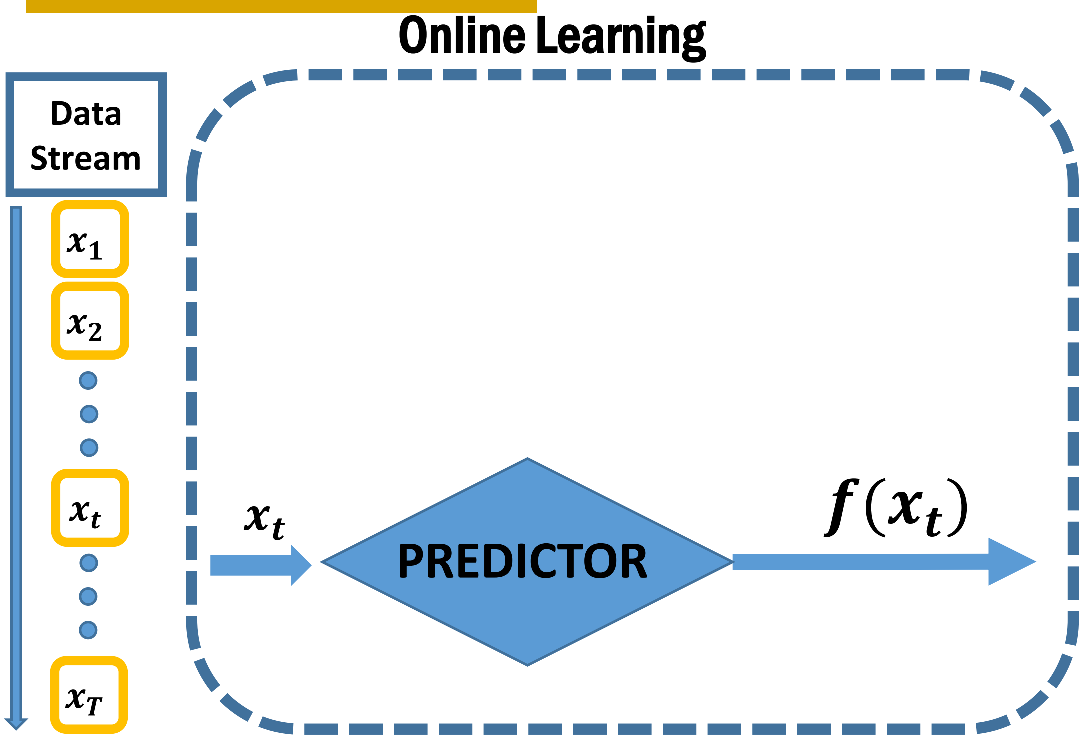
Introduction & Motivation

Online Learning and Deep Learning



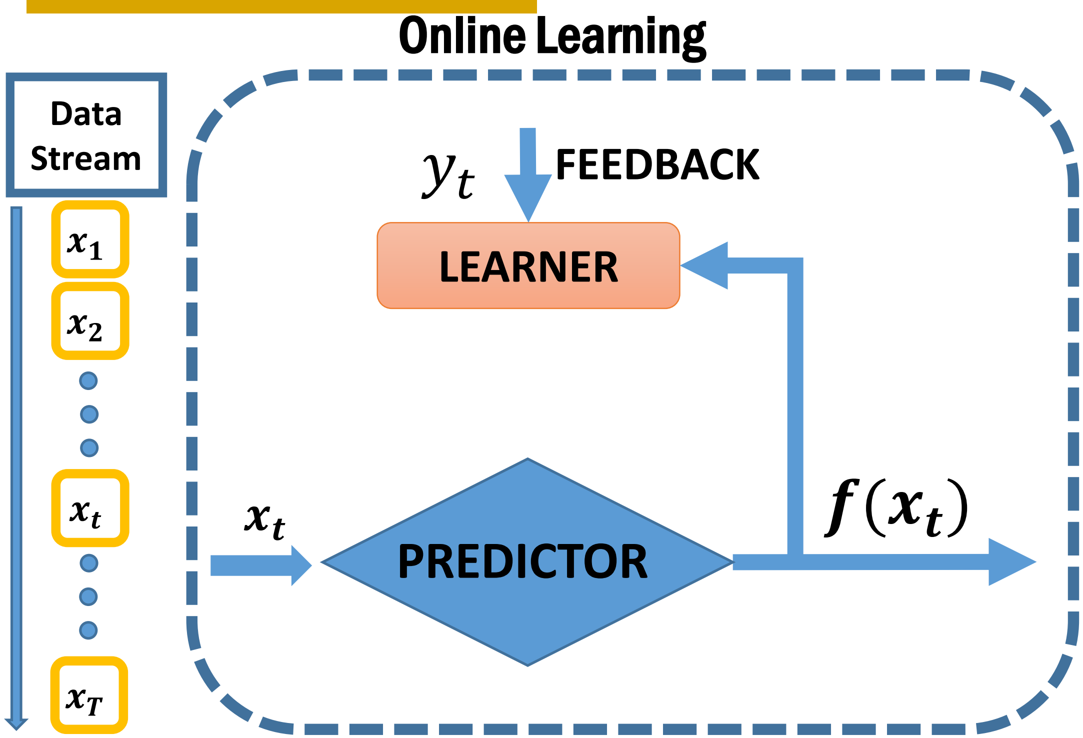
Introduction & Motivation

Online Learning and Deep Learning



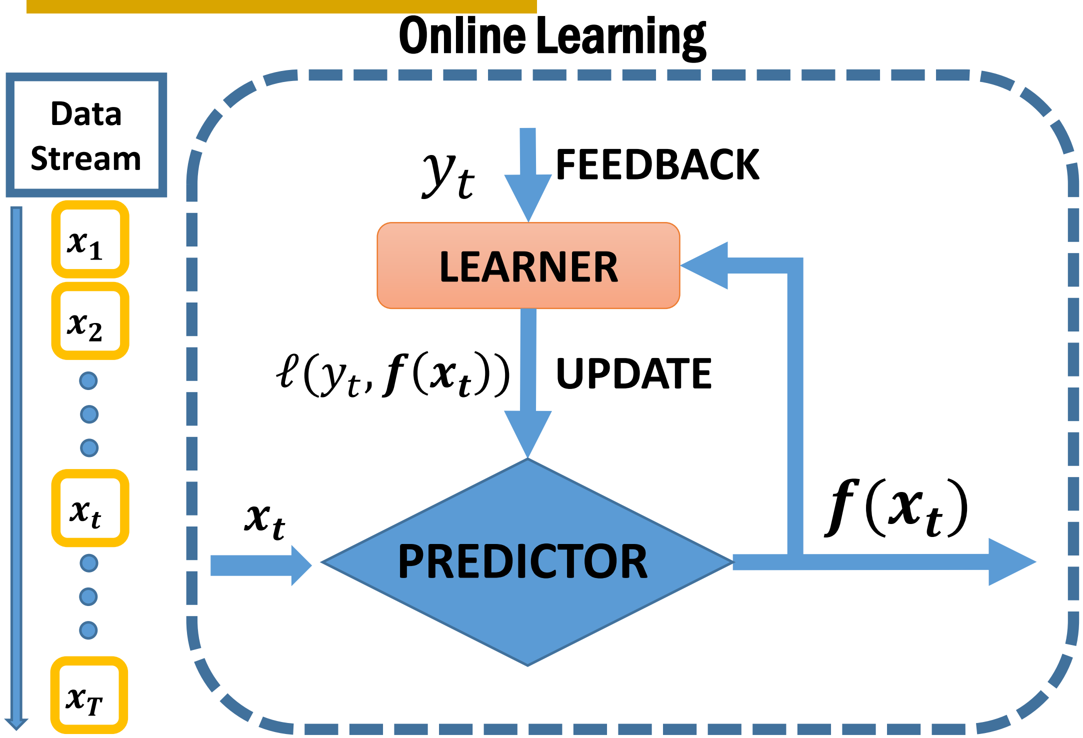
Introduction & Motivation

Online Learning and Deep Learning



Introduction & Motivation

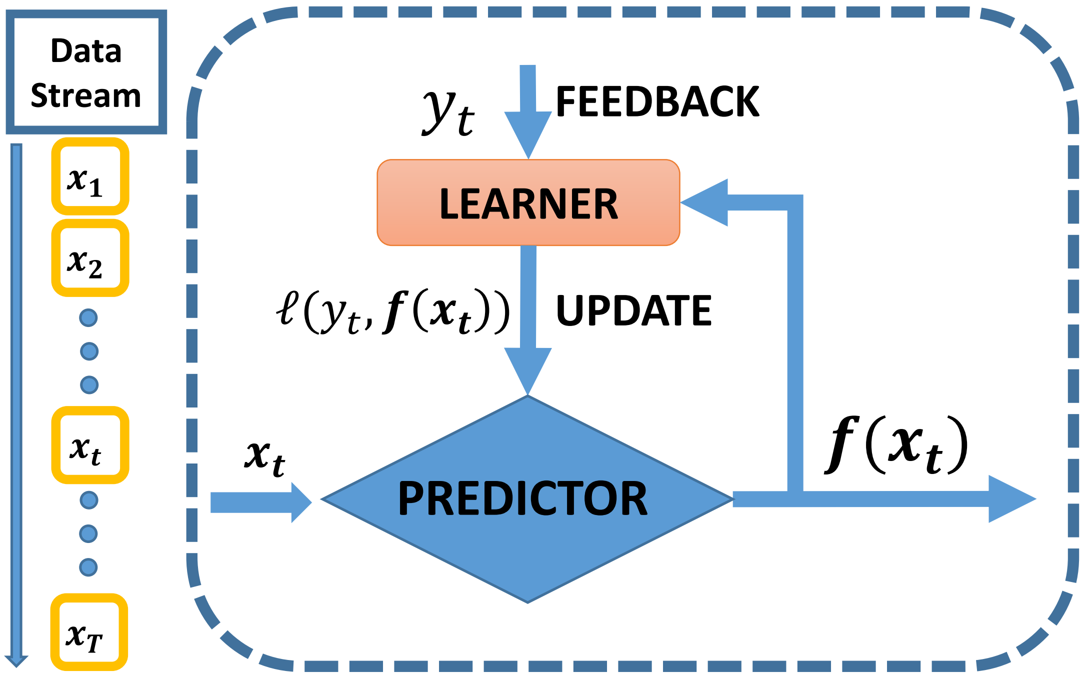
Online Learning and Deep Learning



Introduction & Motivation

Online Learning and Deep Learning

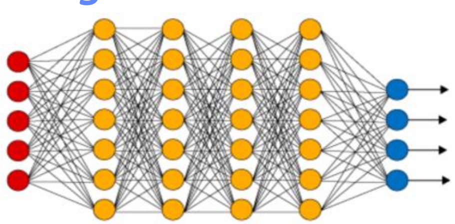
Online Learning



Deep Learning

State of the art in many applications

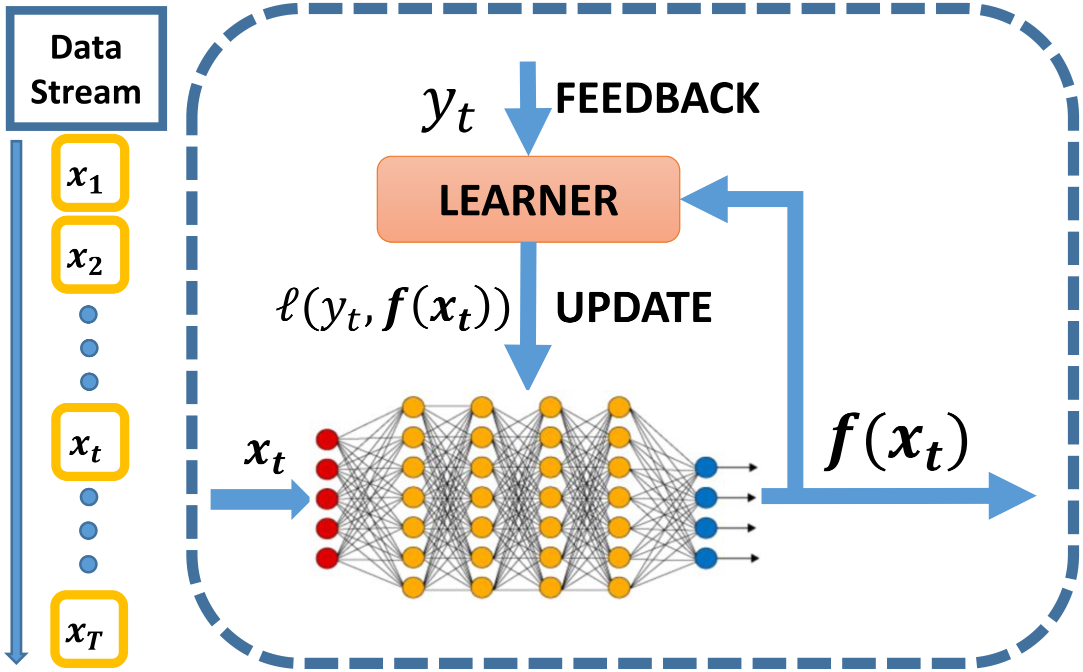
- Easily beats kernel methods
- *Krizhevsky et al. 2012*
- *Simonyan & Zisserman 2014*
- *He et al. 2016*
- *Huang et. al 2017*



Introduction & Motivation

Online Learning and Deep Learning

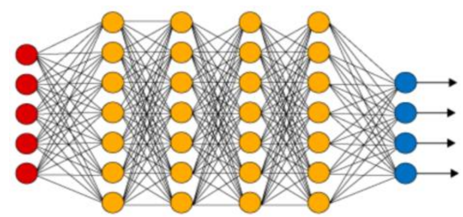
Online Learning



Deep Learning

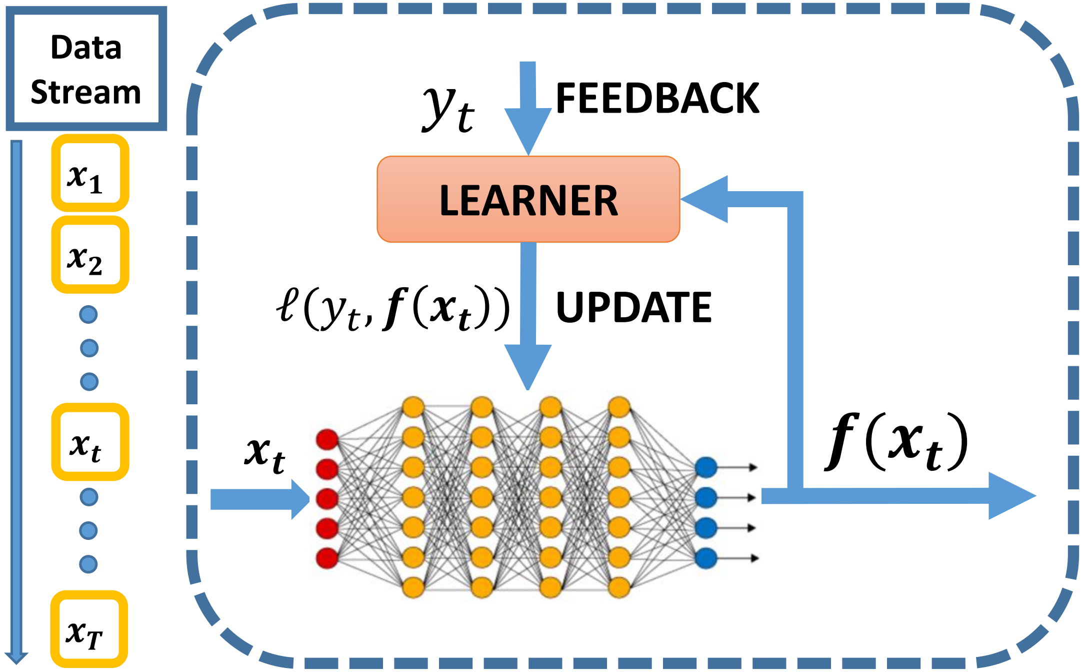
State of the art in many applications

- Easily beats kernel methods
- *Krizhevsky et al. 2012*
- *Simonyan & Zisserman 2014*
- *He et al. 2016*
- *Huang et. al 2017*



Introduction & Motivation

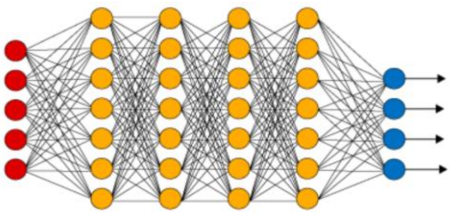
Online Learning and Deep Learning



Deep Learning

State of the art in many applications

- Easily beats kernel methods
- *Krizhevsky et al. 2012*
- *Simonyan & Zisserman 2014*
- *He et al. 2016*
- *Huang et. al 2017*



Existing Online Deep Learning

- Very limited work that addresses Deep Learning in Online Setting
- Few Attempts (*Zhou et al. 2012, Lee et. al. 2016*): Consider Mini-batch Optimization

Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

Choose a (very) deep network –

Choosing a sufficiently complex network ensures that the pattern in data **CAN** be learnt

... however ... particularly for online settings ...

Introduction & Motivation

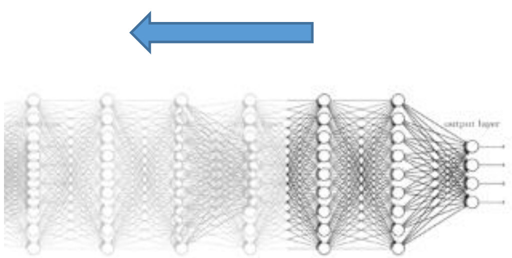
Challenges in using Deep Networks for Online Learning (1/2)

Choose a (very) deep network –

Choosing a sufficiently complex network ensures that the pattern in data **CAN** be learnt

... however ... particularly for online settings ...

Vanishing Gradient



Bengio et al. 1994
Hochreiter 1998, etc.

Introduction & Motivation

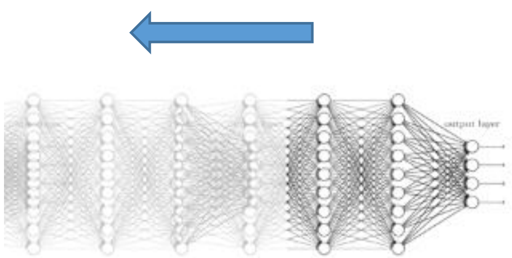
Challenges in using Deep Networks for Online Learning (1/2)

Choose a (very) deep network –

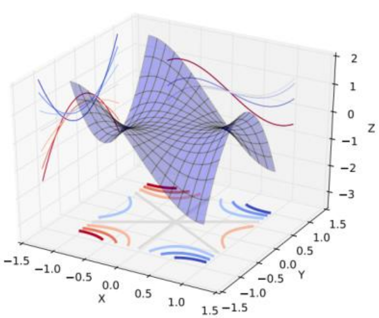
Choosing a sufficiently complex network ensures that the pattern in data **CAN** be learnt

... however ... particularly for online settings ...

Vanishing Gradient	Saddle Points (& Local Minima)



Bengio et al. 1994
Hochreiter 1998, etc.



Dauphin et al. 2014

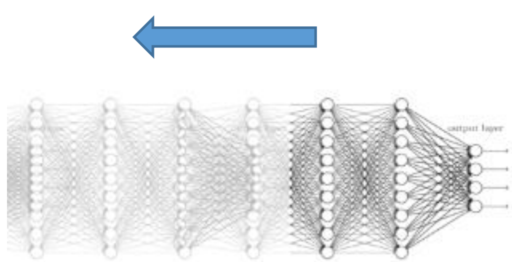
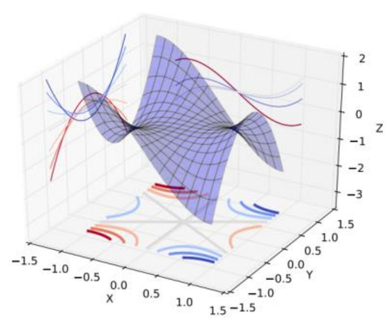
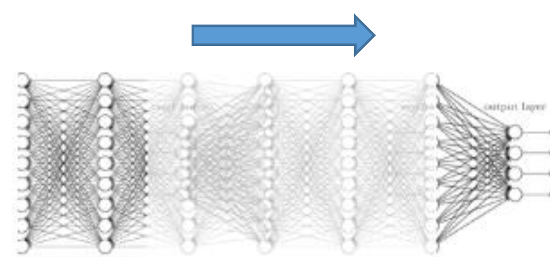
Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

Choose a (very) deep network –

Choosing a sufficiently complex network ensures that the pattern in data **CAN** be learnt

... however ... particularly for online settings ...

Vanishing Gradient	Saddle Points (& Local Minima)	Diminishing Feature Reuse
		
Bengio et al. 1994 *Hochreiter 1998, etc.*	*Dauphin et al. 2014*	*Srivastava et al. 2015*

Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

Unique Problem – prefer different depths at different stages of training

Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

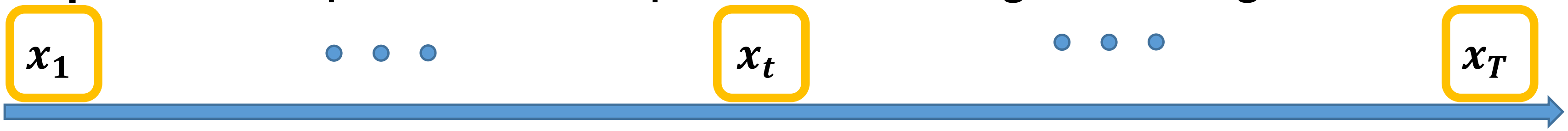
Unique Problem – prefer different depths at different stages of training

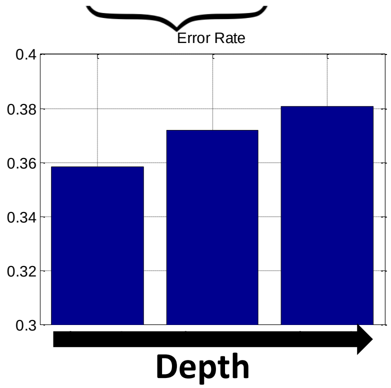
x_1 • • • x_t • • • x_T

Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

Unique Problem – prefer different depths at different stages of training

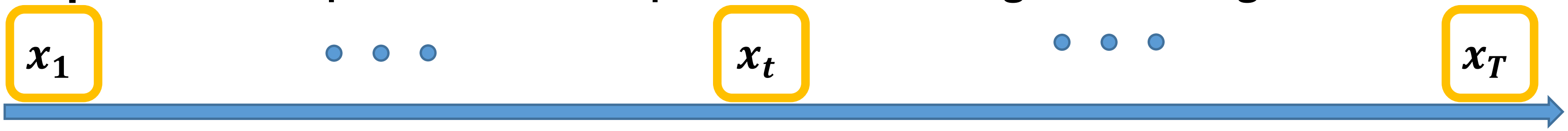


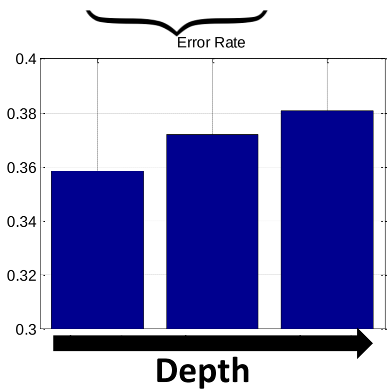


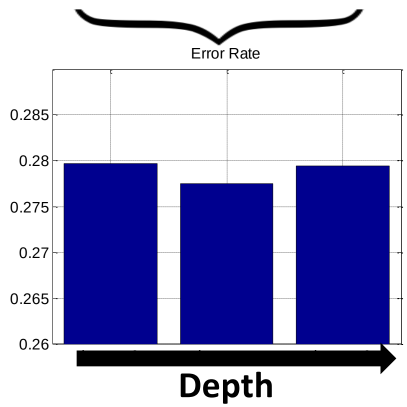
Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

Unique Problem – prefer different depths at different stages of training



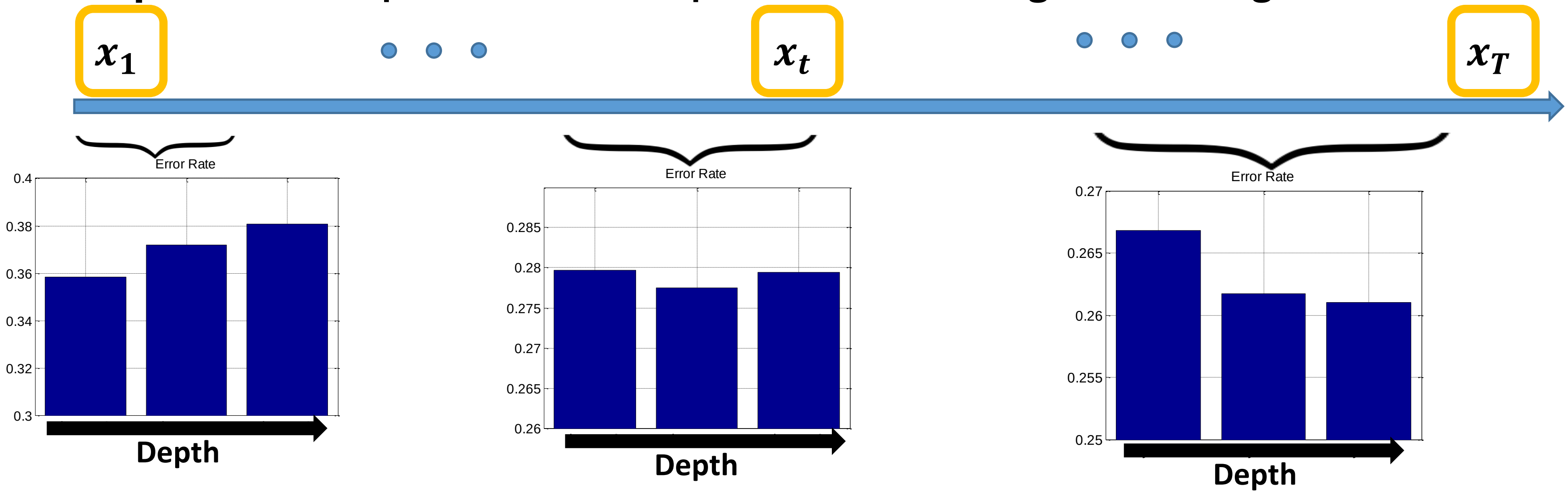




Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

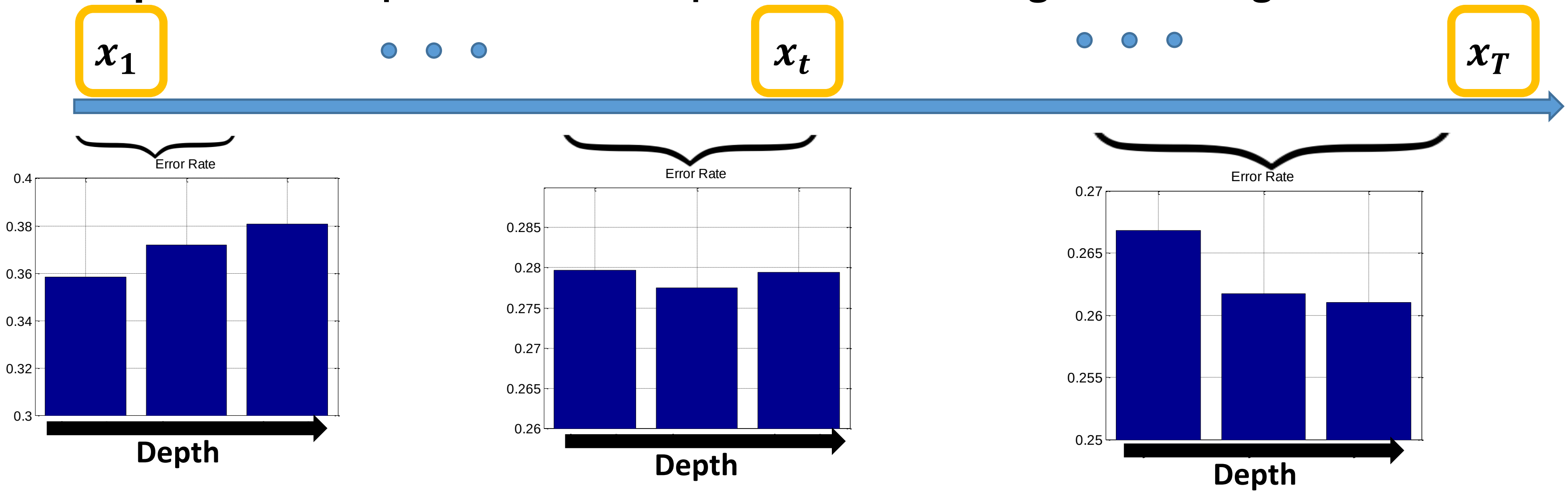
Unique Problem – prefer different depths at different stages of training



Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

Unique Problem – prefer different depths at different stages of training

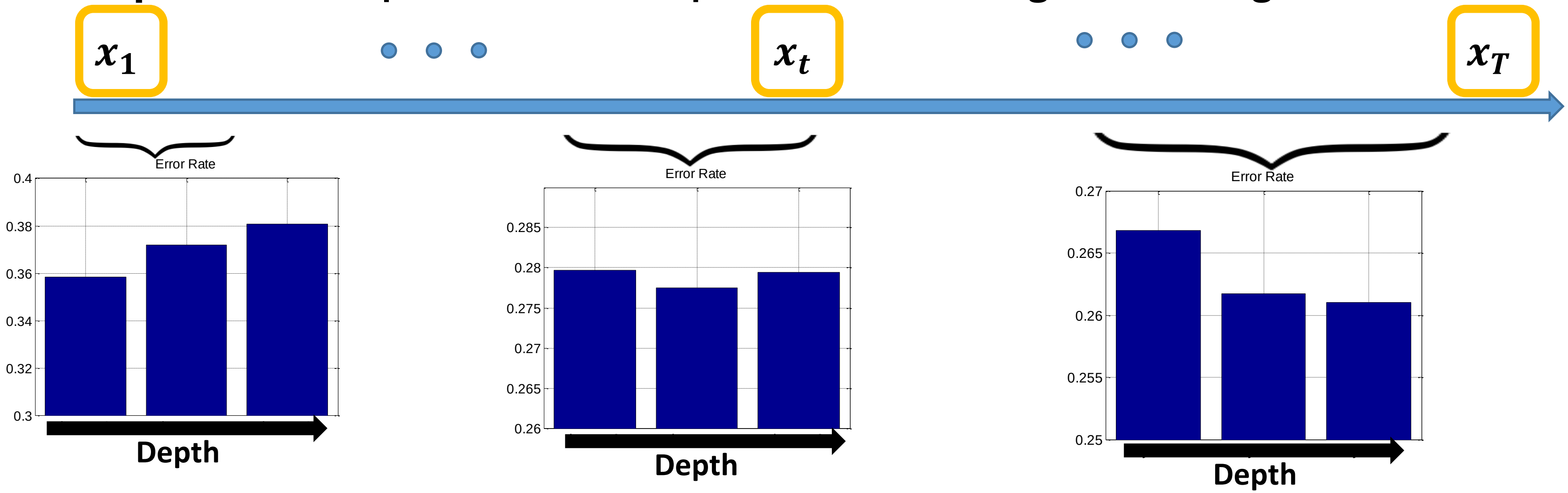


- **Final Error** – anyone could be best depending on how much of the data has been processed.
- Problem is magnified for concept-drift scenarios
- Best of both worlds?

Introduction & Motivation

Challenges in using Deep Networks for Online Learning (1/2)

Unique Problem – prefer different depths at different stages of training



- **Final Error** – anyone could be best depending on how much of the data has been processed.
- Problem is magnified for concept-drift scenarios
- Best of both worlds?

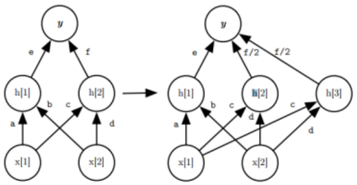
Fast Learning + Power of Deep Representation

Online Deep Learning

Shallow to Deep Principle

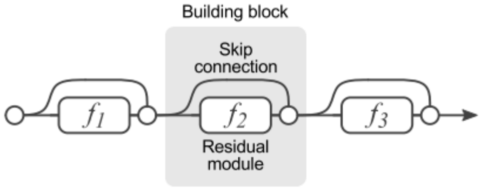
Explicitly Shallow to Deep (Function Preservation Principle)

- Net2Net — *Chen et al. 2016*
- NetMorph — *Wei et al. 2016*



Implicitly Shallow to Deep

- ResNet — *He et al. 2016*
- Highway Net — *Srivastava et al. 2015*
- DenseNet — *Huang et al. 2017*
- also Fractal Net — *Larsson et al. 2017*



Online Deep Learning

Architecture for ODL

Shallow to Deep principle is suited for Online Deep Learning

- Start Shallow → Faster Convergence
- Become Deeper → Deep Representation

Proposed Architecture

Attach intermediate classifier to every hidden layer

Online Deep Learning

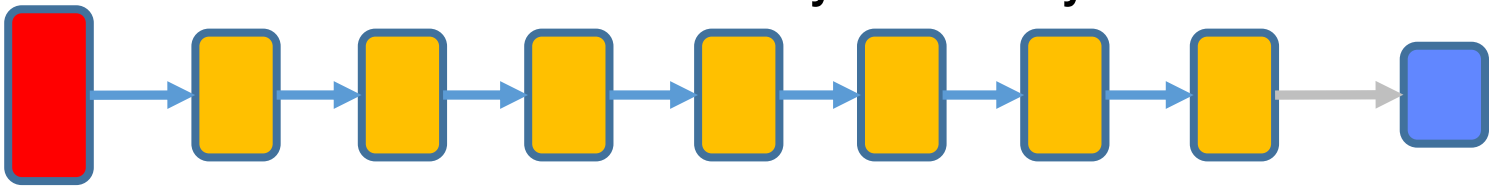
Architecture for ODL

Shallow to Deep principle is suited for Online Deep Learning

- Start Shallow → Faster Convergence
- Become Deeper → Deep Representation

Proposed Architecture

Attach intermediate classifier to every hidden layer



Online Deep Learning

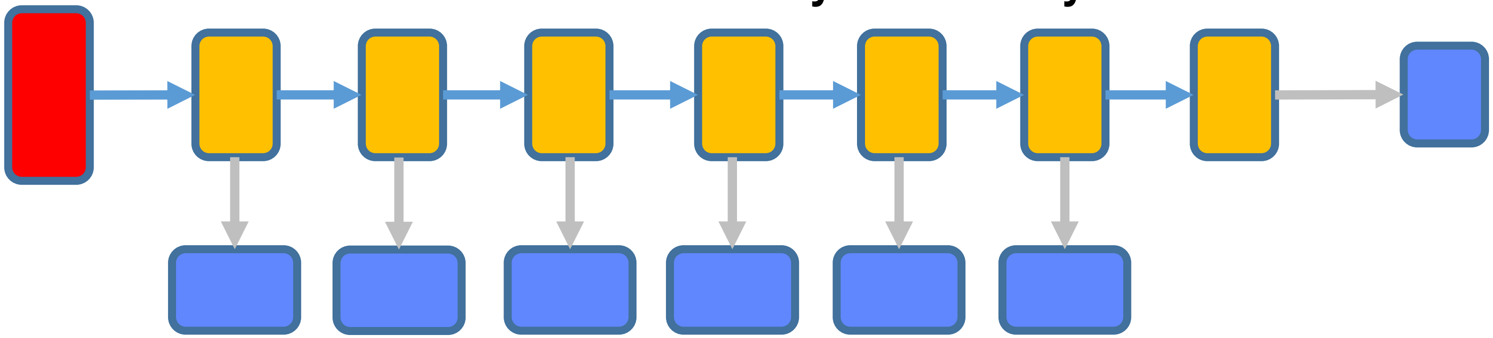
Architecture for ODL

Shallow to Deep principle is suited for Online Deep Learning

- Start Shallow → Faster Convergence
- Become Deeper → Deep Representation

Proposed Architecture

Attach intermediate classifier to every hidden layer



Online Deep Learning

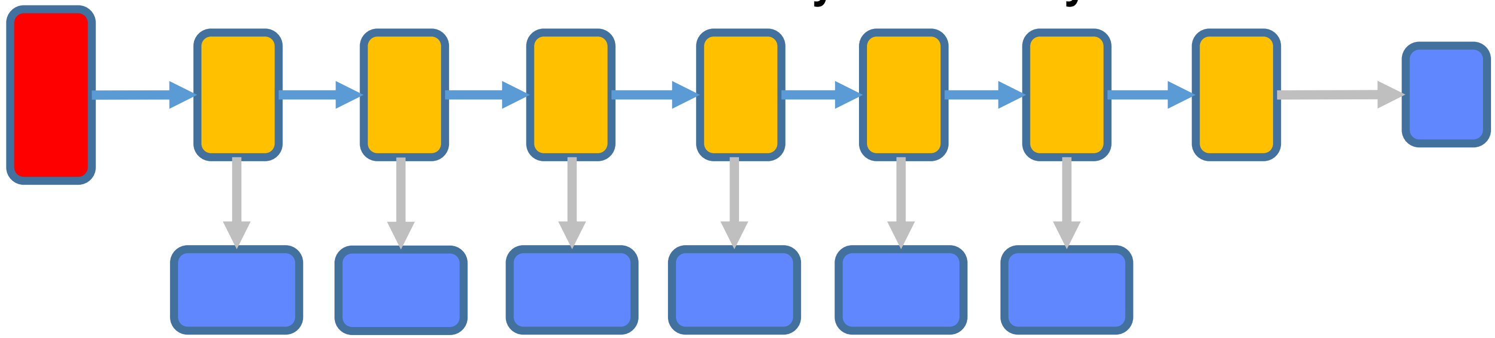
Architecture for ODL

Shallow to Deep principle is suited for Online Deep Learning

- Start Shallow → Faster Convergence
- Become Deeper → Deep Representation

Proposed Architecture

Attach intermediate classifier to every hidden layer



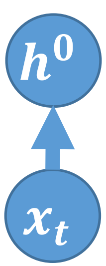
Dynamically vary the Effective Depth based on the data

Online Deep Learning

Hedge Backpropagation (1/3)

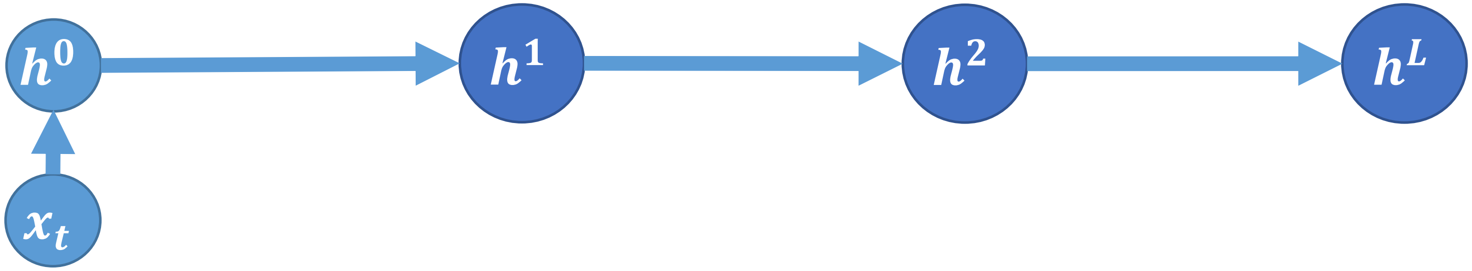
Online Deep Learning

Hedge Backpropagation (1/3)



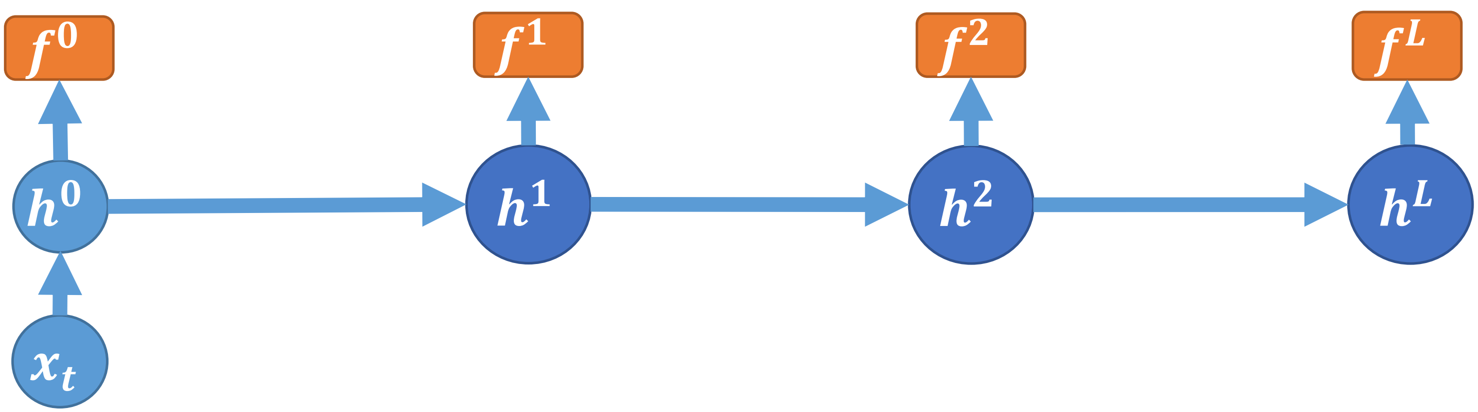
Online Deep Learning

Hedge Backpropagation (1/3)



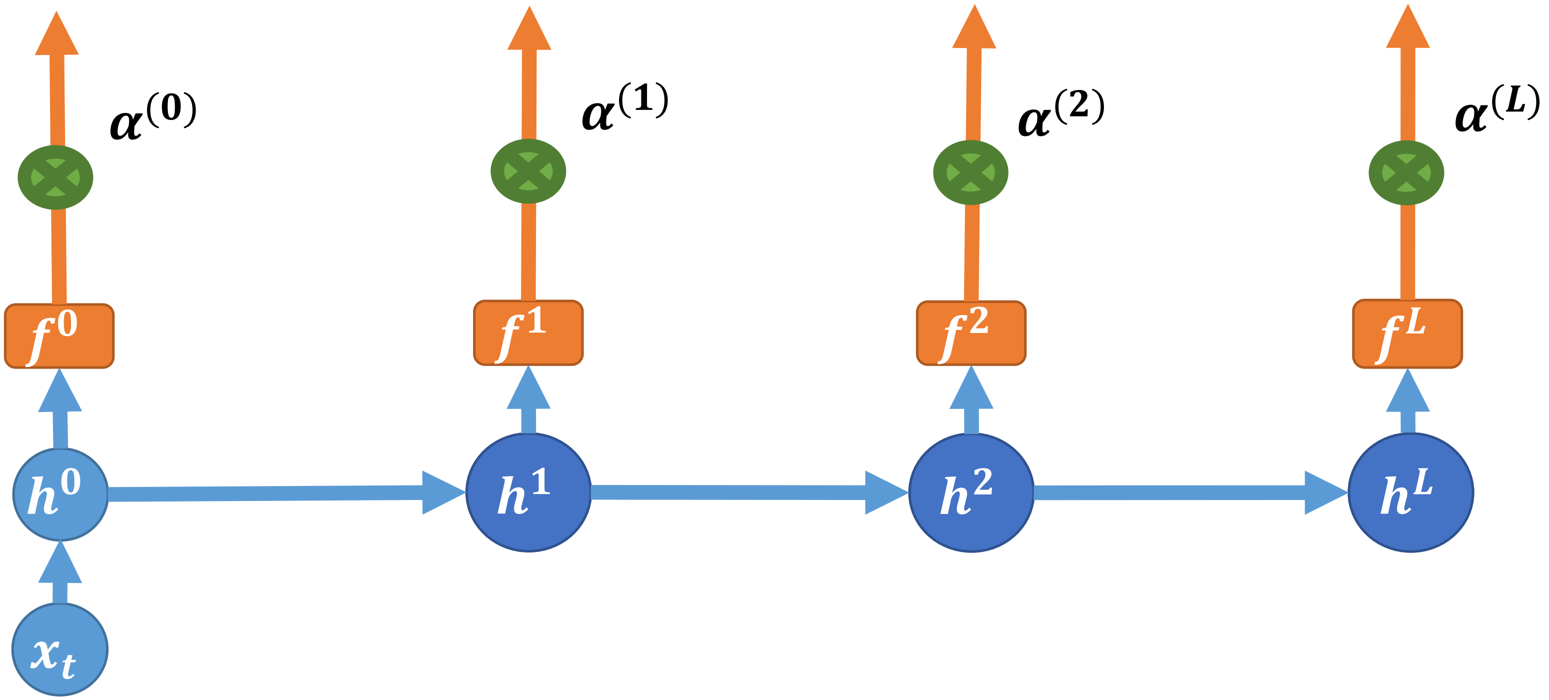
Online Deep Learning

Hedge Backpropagation (1/3)



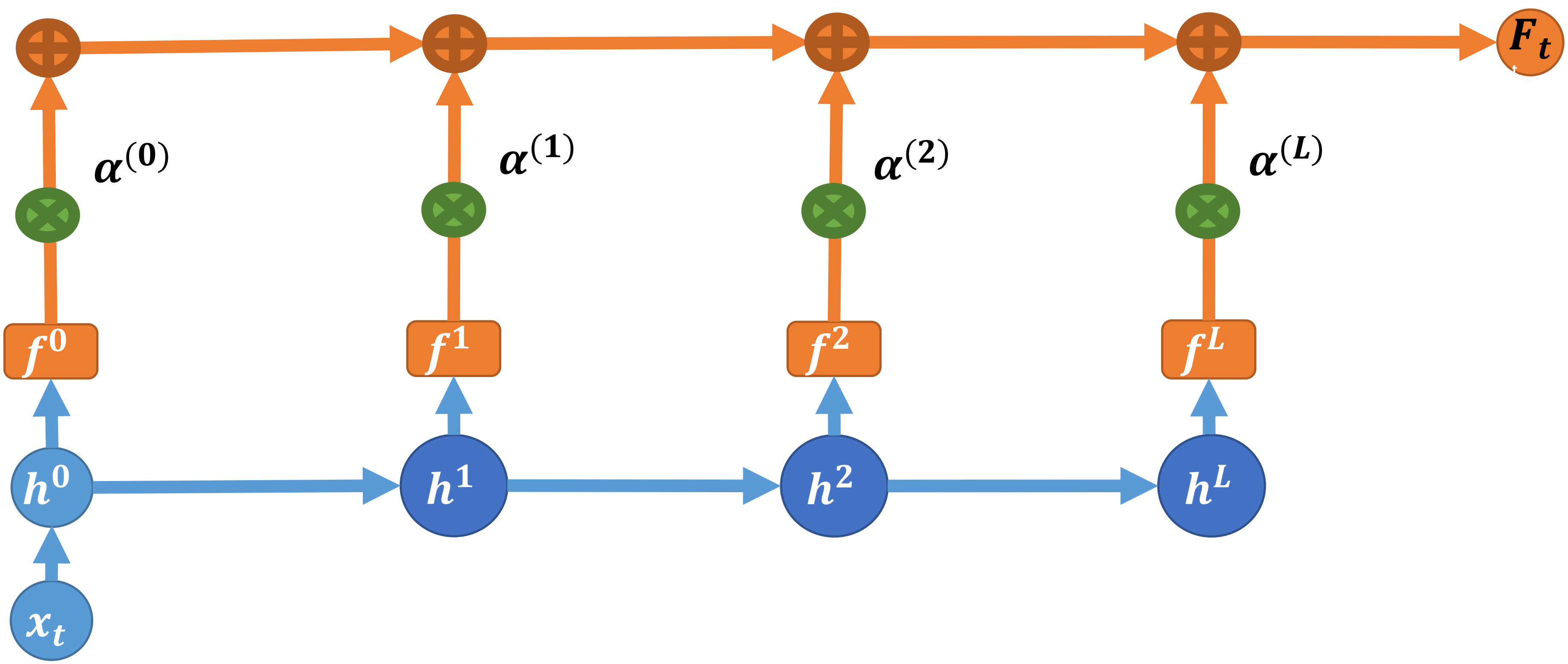
Online Deep Learning

Hedge Backpropagation (1/3)



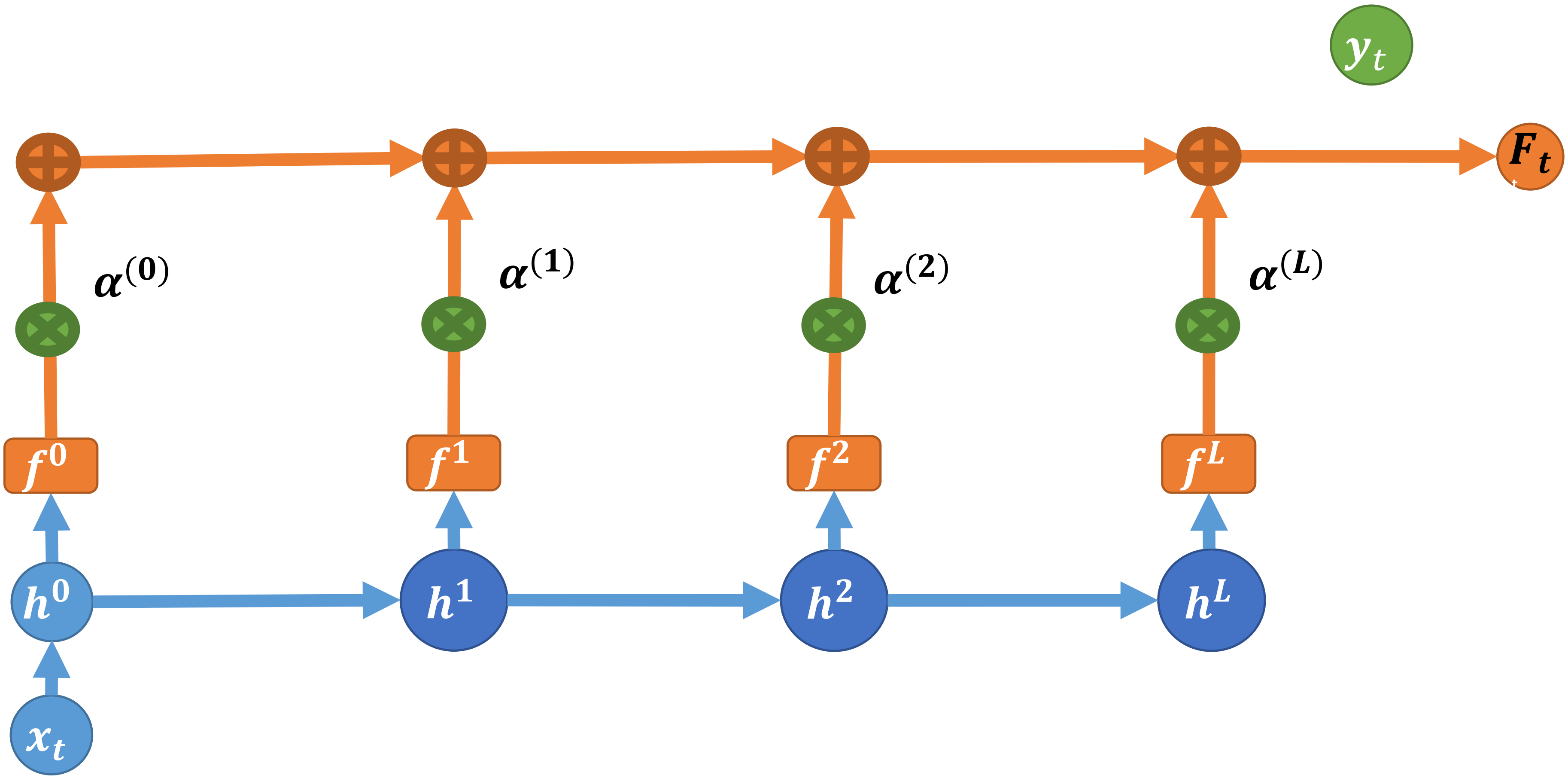
Online Deep Learning

Hedge Backpropagation (1/3)



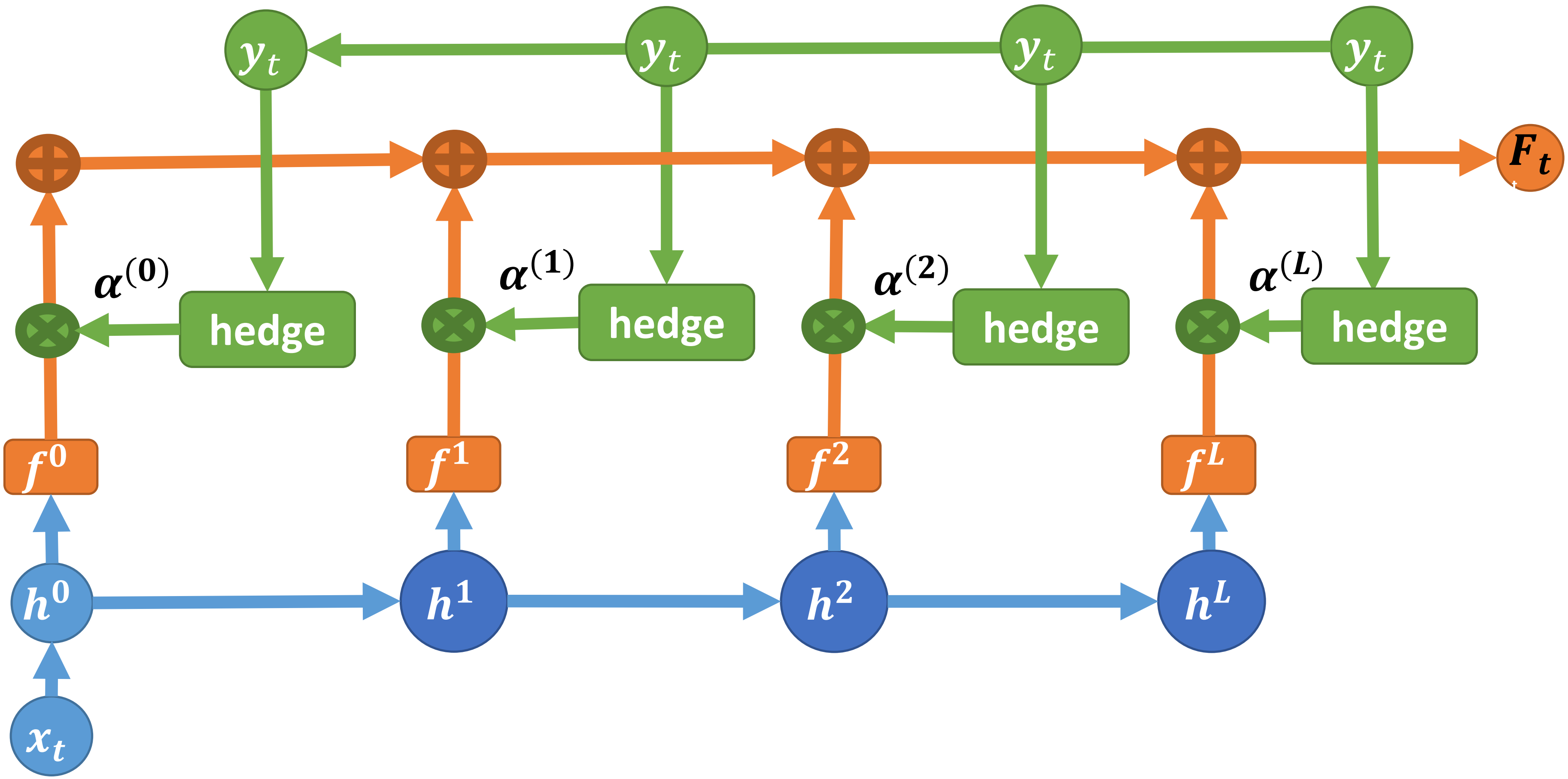
Online Deep Learning

Hedge Backpropagation (1/3)



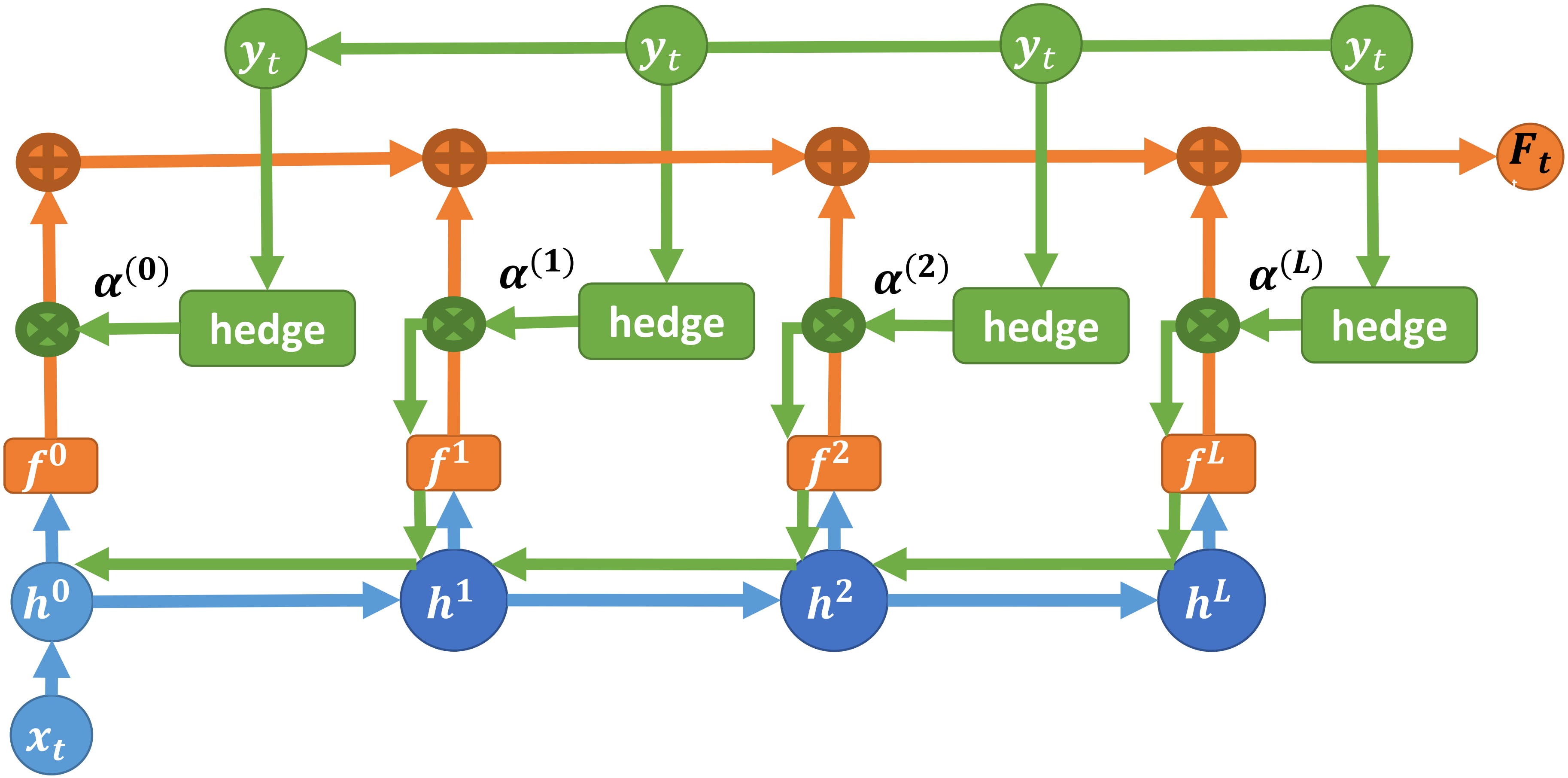
Online Deep Learning

Hedge Backpropagation (1/3)



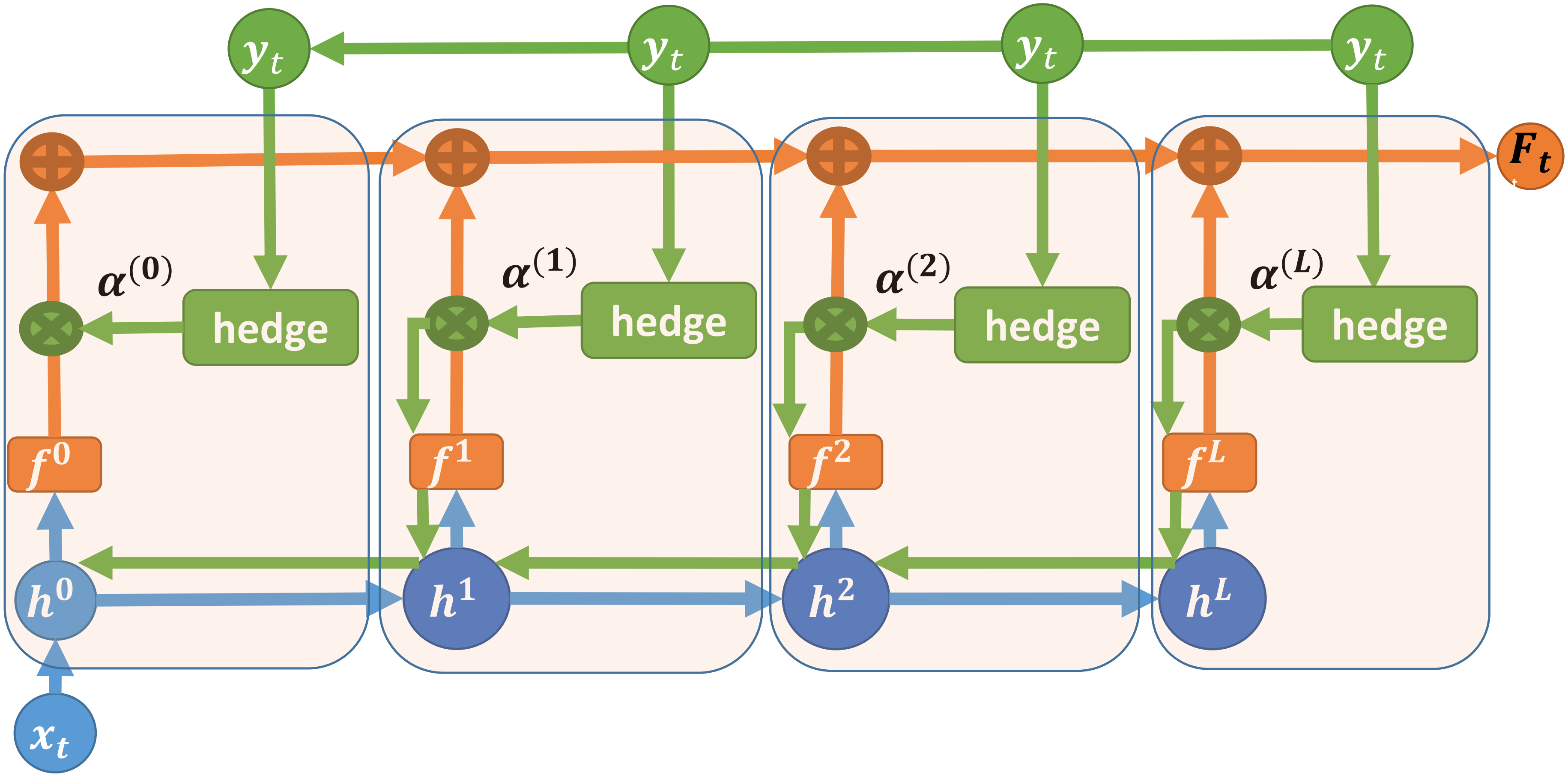
Online Deep Learning

Hedge Backpropagation (1/3)



Online Deep Learning

Hedge Backpropagation (1/3)



Online Deep Learning

Hedge Backpropagation (2/3)

Online Deep Learning

Hedge Backpropagation (2/3)

A **Dynamic** Objective Function

$$\mathcal{L}(\mathbf{F}(\mathbf{x}), y) = \sum_{l=0}^{L} \alpha^{(l)} \mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)$$

Online Deep Learning

Hedge Backpropagation (2/3)

A **Dynamic** Objective Function

$$\mathcal{L}(\mathbf{F}(\mathbf{x}), y) = \sum_{l=0}^{L} \alpha^{(l)} \mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)$$

3 Main Updates

Loss / Classifier Weight update (Hedge)

Classifier Update

DNN Update

Online Deep Learning

Hedge Backpropagation (2/3)

A **Dynamic** Objective Function

$$\mathcal{L}(\mathbf{F}(\mathbf{x}), y) = \sum_{l=0}^{L} \alpha^{(l)} \mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)$$

3 Main Updates

Loss / Classifier Weight update (Hedge) $\alpha_{t+1}^{(l)} \leftarrow \alpha_t^{(l)} \beta^{\mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)}$

Classifier Update

DNN Update

Online Deep Learning

Hedge Backpropagation (2/3)

A Dynamic Objective Function

$$\mathcal{L}(\mathbf{F}(\mathbf{x}), y) = \sum_{l=0}^{L} \alpha^{(l)} \mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)$$

3 Main Updates

Loss / Classifier Weight update (Hedge)

$$\alpha_{t+1}^{(l)} \leftarrow \alpha_t^{(l)} \beta^{\mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)}$$

Classifier Update

$$\begin{aligned} \Theta_{t+1}^{(l)} &\leftarrow \Theta_t^{(l)} - \eta \nabla_{\Theta_t^{(l)}} \mathcal{L}(\mathbf{F}(\mathbf{x}_t, y_t)) \\ &= \Theta_t^{(l)} - \eta \alpha^{(l)} \nabla_{\Theta_t^{(l)}} \mathcal{L}(\mathbf{f}^{(l)}, y_t) \end{aligned}$$

DNN Update

Online Deep Learning

Hedge Backpropagation (2/3)

A Dynamic Objective Function

$$\mathcal{L}(\mathbf{F}(\mathbf{x}), y) = \sum_{l=0}^{L} \alpha^{(l)} \mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)$$

3 Main Updates

Loss / Classifier Weight update (Hedge)	$\alpha_{t+1}^{(l)} \leftarrow \alpha_t^{(l)} \beta^{\mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)}$
Classifier Update	$\Theta_{t+1}^{(l)} \leftarrow \Theta_t^{(l)} - \eta \nabla_{\Theta_t^{(l)}} \mathcal{L}(\mathbf{F}(\mathbf{x}_t, y_t))$ $= \Theta_t^{(l)} - \eta \alpha^{(l)} \nabla_{\Theta_t^{(l)}} \mathcal{L}(\mathbf{f}^{(l)}, y_t)$
DNN Update	$W_{t+1}^{(l)} \leftarrow W_t^{(l)} - \eta \sum_{j=l}^{L} \alpha^{(j)} \nabla_{W^{(l)}} \mathcal{L}(\mathbf{f}^{(j)}, y_t)$

Online Deep Learning

Hedge Backpropagation (2/3)

A **Dynamic** Objective Function

$$\mathcal{L}(\mathbf{F}(\mathbf{x}), y) = \sum_{l=0}^{L} \alpha^{(l)} \mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)$$

3 Main Updates

Loss / Classifier Weight update (Hedge)

$$\alpha_{t+1}^{(l)} \leftarrow \alpha_t^{(l)} \beta^{\mathcal{L}(\mathbf{f}^{(l)}(\mathbf{x}), y)}$$

Classifier Update

$$\begin{aligned} \Theta_{t+1}^{(l)} &\leftarrow \Theta_t^{(l)} - \eta \nabla_{\Theta_t^{(l)}} \mathcal{L}(\mathbf{F}(\mathbf{x}_t, y_t)) \\ &= \Theta_t^{(l)} - \eta \alpha^{(l)} \nabla_{\Theta_t^{(l)}} \mathcal{L}(\mathbf{f}^{(l)}, y_t) \end{aligned}$$

DNN Update

$$W_{t+1}^{(l)} \leftarrow W_t^{(l)} - \eta \sum_{j=l}^{L} \alpha^{(j)} \nabla_{W^{(l)}} \mathcal{L}(\mathbf{f}^{(j)}, y_t)$$

Smoothing Parameter

$$\alpha^{(l)} \leftarrow \max\left(\alpha^{(l)}, \frac{s}{L}\right)$$

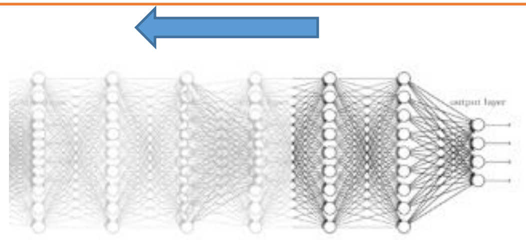
Online Deep Learning

Hedge Backpropagation (3/3)

Online Deep Learning

Hedge Backpropagation (3/3)

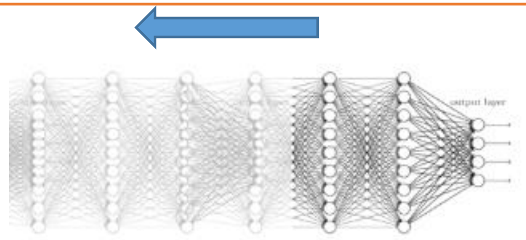
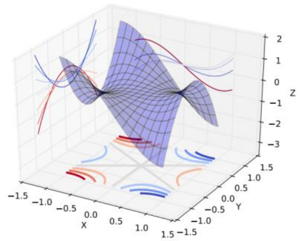
Vanishing Gradient



Intermediate classifiers reduce initial susceptibility to Vanishing Gradient

Online Deep Learning

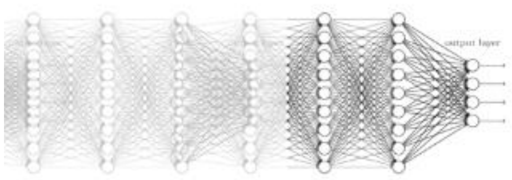
Hedge Backpropagation (3/3)

Vanishing Gradient	Saddle Points (& Local Minima)
	
Intermediate classifiers reduce initial susceptibility to Vanishing Gradient	Multiple Loss functions allow easier exits from saddle points

Online Deep Learning

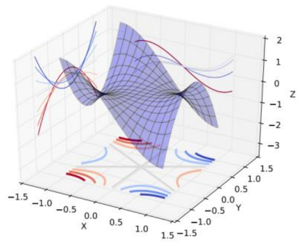
Hedge Backpropagation (3/3)

Vanishing Gradient



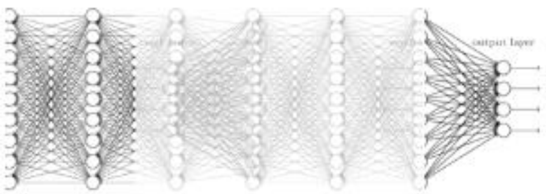
Intermediate classifiers reduce initial susceptibility to Vanishing Gradient

Saddle Points (& Local Minima)



Multiple Loss functions allow easier exits from saddle points

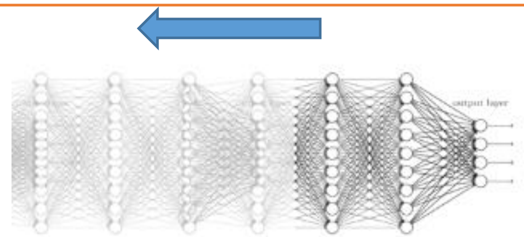
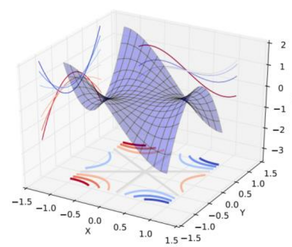
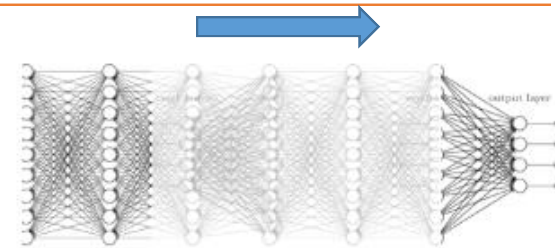
Diminishing Feature Reuse



Intermediate features are directly used for classification

Online Deep Learning

Hedge Backpropagation (3/3)

Vanishing Gradient	Saddle Points (& Local Minima)	Diminishing Feature Reuse
		
Intermediate classifiers reduce initial susceptibility to Vanishing Gradient	Multiple Loss functions allow easier exits from saddle points	Intermediate features are directly used for classification

Parallel Interpretations

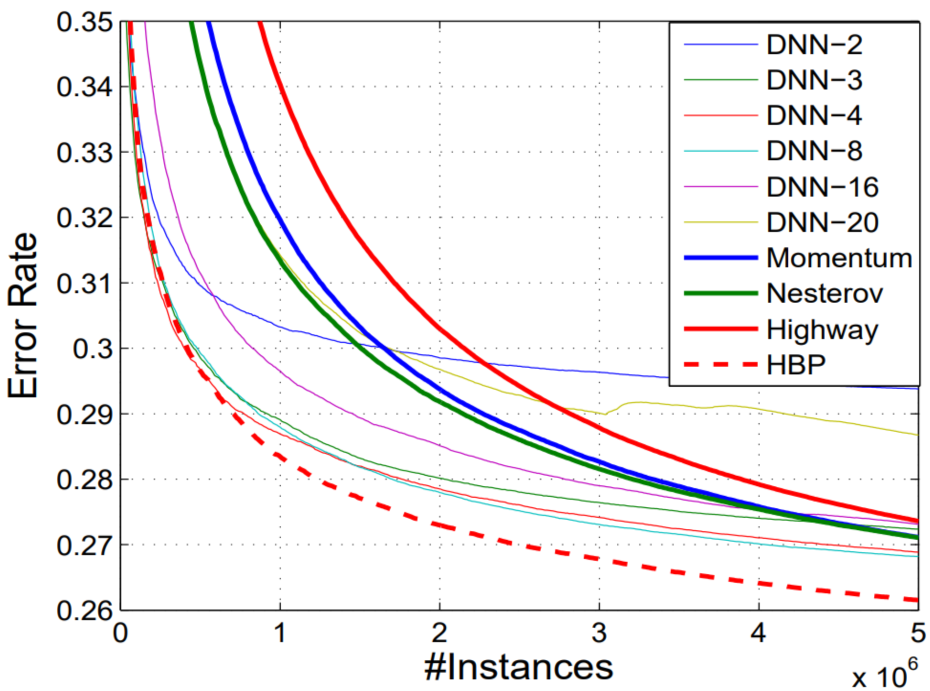
Student-Teacher Learning | Lifelong Learning | Concept-Drift Adaptation, etc.

Experiments

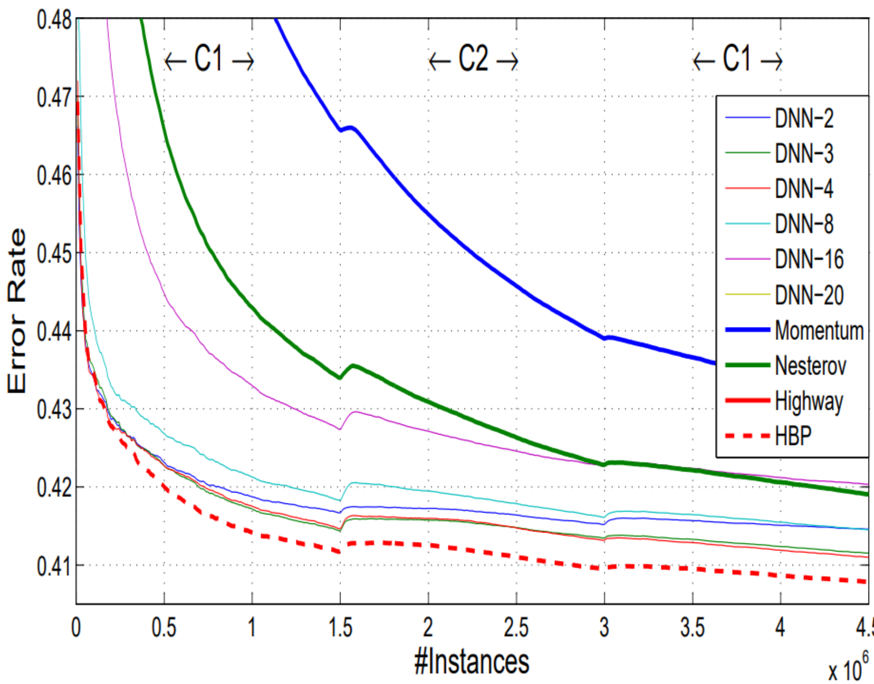
Convergence behavior on stationary and concept drift datasets

Baselines: Linear and Kernel OL | DNNs with varying depth | DNN-20 – with momentum, Highway

Proposed: Online Deep Learning by Hedge Backpropagation (DNN-20)



(a) HIGGS



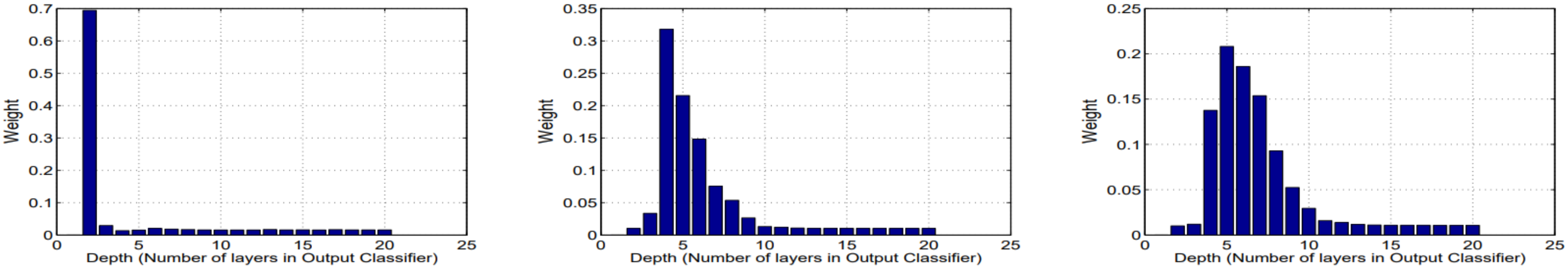
(e) Concept Drift 1 (CD1)

Experiments

Other Insights

Experiments

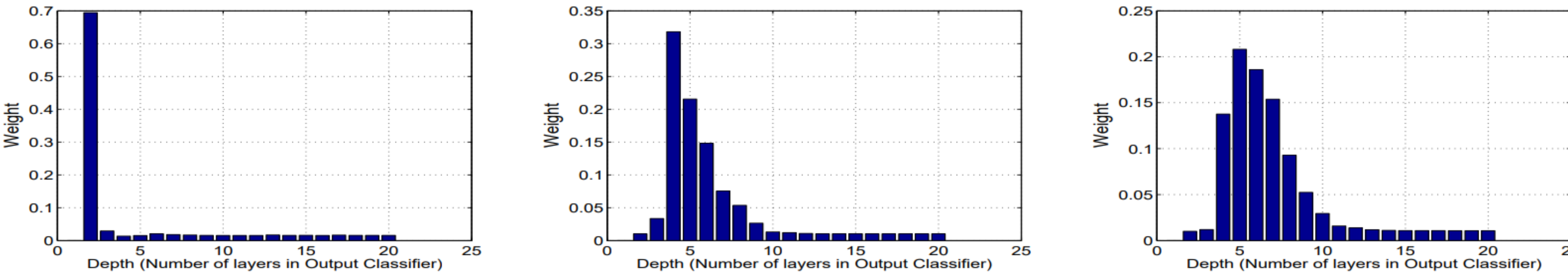
Other Insights



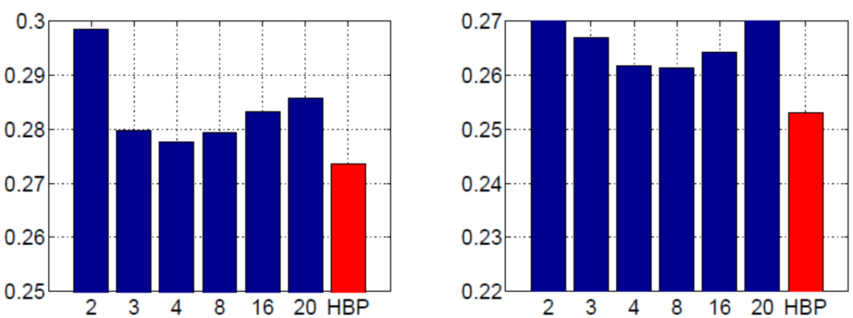
(a) First 0.5% of Data   (b) 10-15% of Data   (c) 60-80% of Data

Experiments

Other Insights



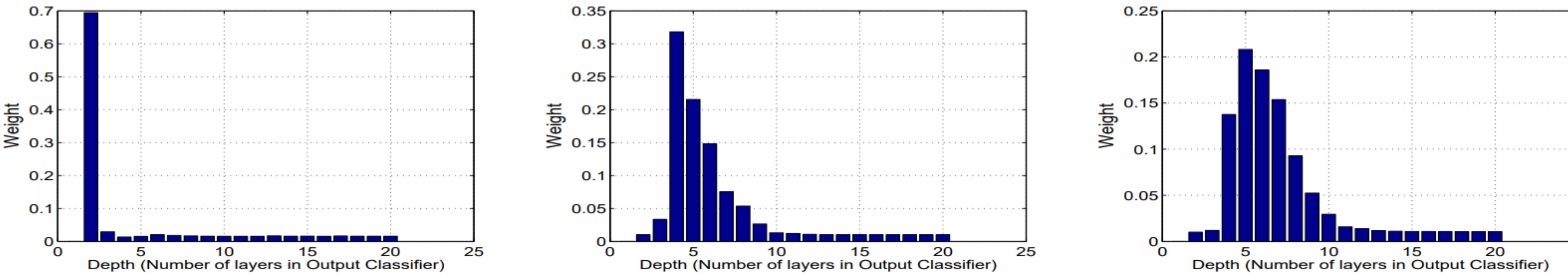
(a) First 0.5% of Data (b) 10-15% of Data (c) 60-80% of Data



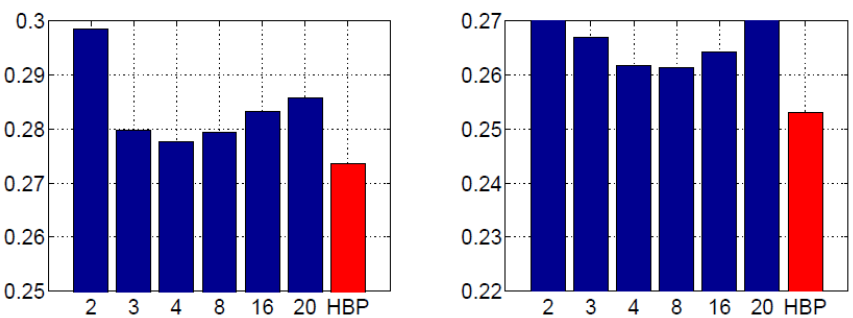
(a) Error in 10-15% of data (b) Error in 60-80% of data

Experiments

Other Insights



(a) First 0.5% of Data (b) 10-15% of Data (c) 60-80% of Data



(a) Error in 10-15% of data (b) Error in 60-80% of data

Error Variation with Depth

Depth	12	16	20	30
Online BP	0.2692	0.2731	0.2868	0.4770
HBP	0.2609	0.2613	0.2615	0.2620

Acknowledgements

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative.

Thank you

LARC
LIVING ANALYTICS
RESEARCH CENTRE

Personalized Participatory Nation

School of Information Systems
Singapore Management University
80 Stamford Road
Singapore 178902
Tel: 65 6808 5227

larc.smu.edu.sg facebook.com/larc.cmu.smu twitter.com/larc_cmu_smu