Rochester Institute of Technology

CVPR SEATTLE WASHINGTON JUNE 16-18 2020

Lifelong Machine Learning with Deep Streaming Linear Discriminant Analysis

Tyler L. Hayes[1] and Christopher Kanan[1,2,3]

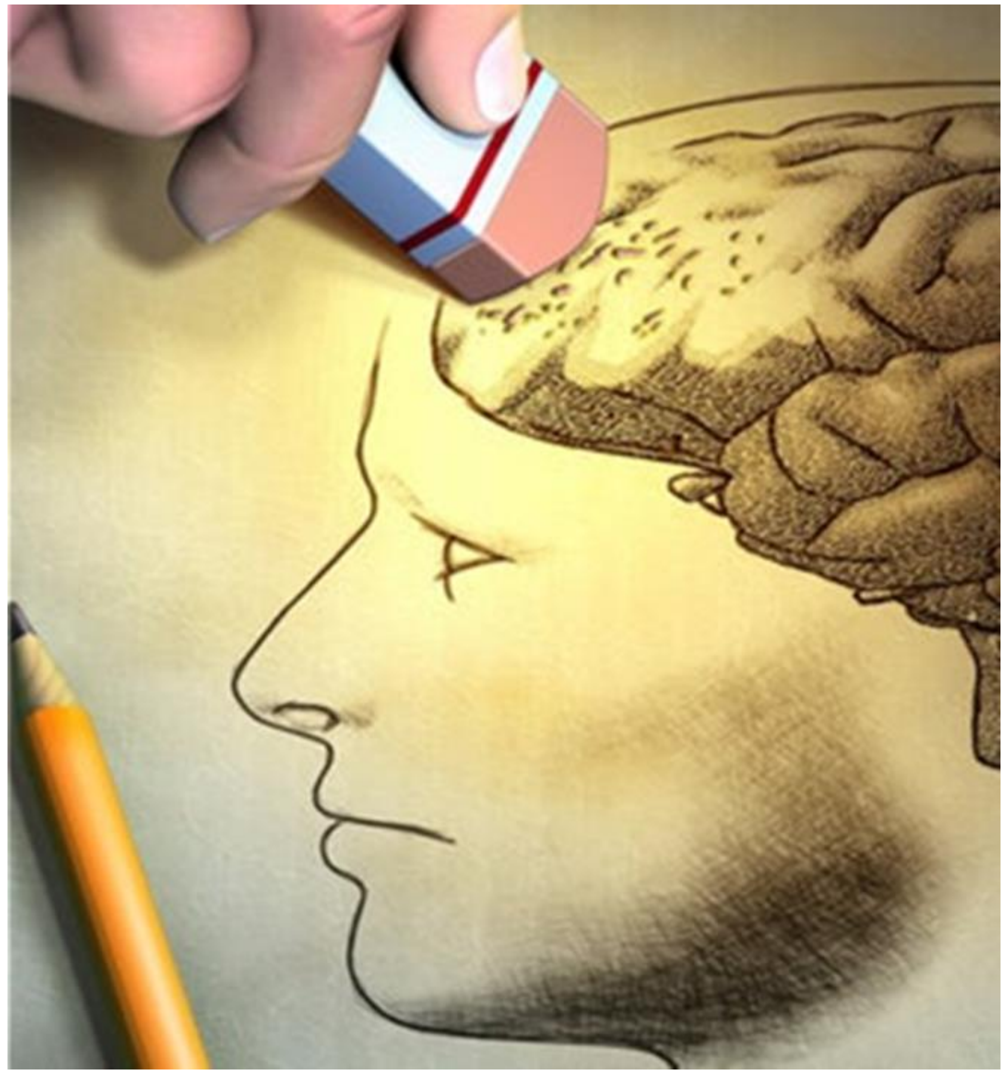
1. Rochester Institute of Technology, 2. Paige, 3. Cornell Tech

Paper available on arXiv: https://arxiv.org/abs/1909.01520
Code available on GitHub: https://github.com/tyler-hayes/Deep_SLDA
Contact: tlh6792@rit.edu; https://tyler-hayes.github.io

CVPR-2020 Workshop on Continual Learning in Computer Vision (CLVision)

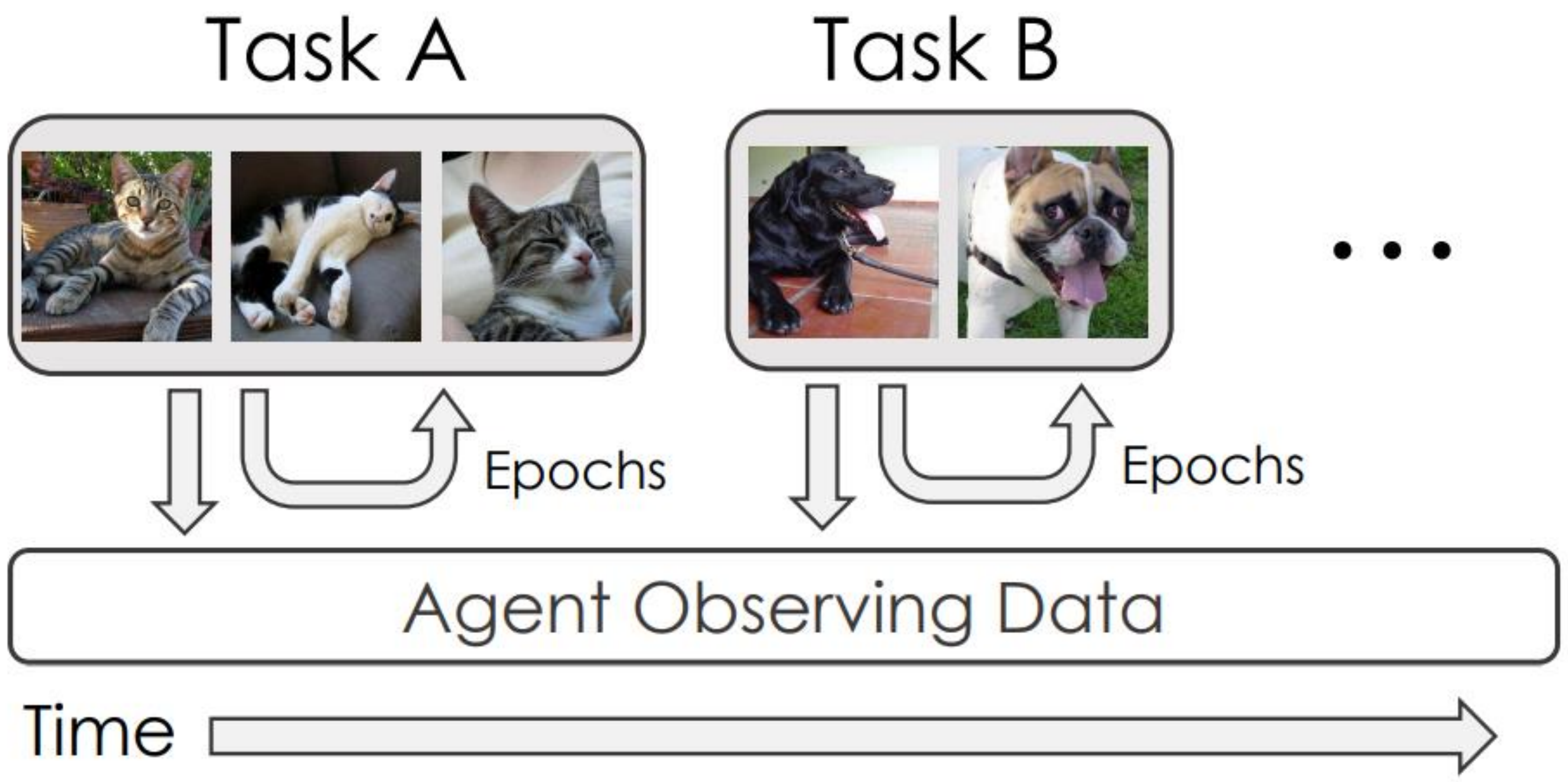
Lifelong Incremental Learning

- People continually learn new information throughout life
- Deep Neural Networks are the dominant approach to machine perception, but:
  - They cannot learn new instances immediately
  - Learning requires multiple loops over a dataset
  - They are susceptible to ***catastrophic forgetting*** if data is not independent and identically distributed
- A lifelong incremental learner overcomes these limitations and should be subject to both memory and time constraints



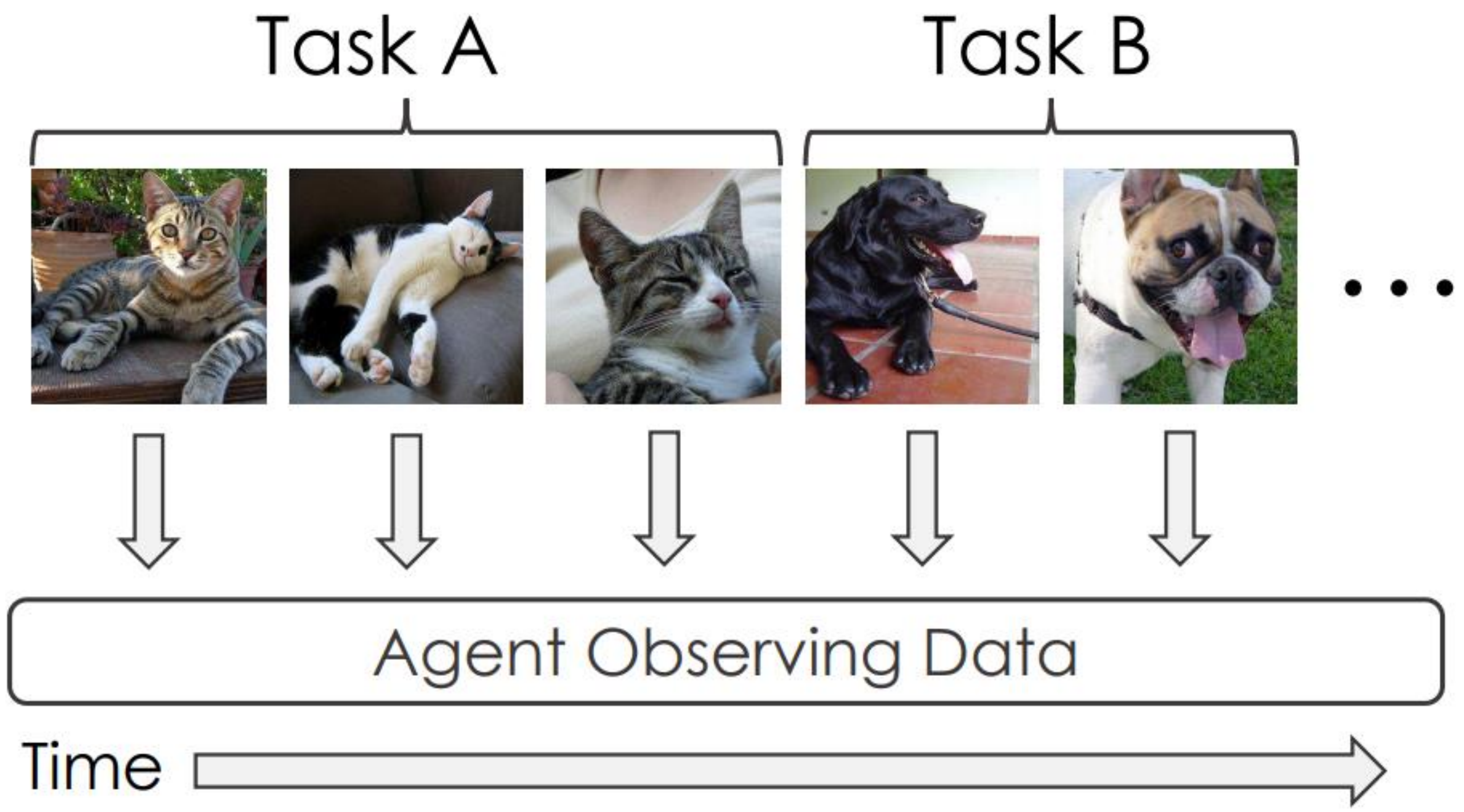
Incremental Batch Learning

- Learner receives a batch of data from one or more classes, may **loop** over the batch until learned, and can only be evaluated at the end of training a batch
- **Caveats:**
    - Must wait for data batch to accumulate before learning
    - Looping makes learning time consuming
    - Must wait until after batch has been learned to evaluate



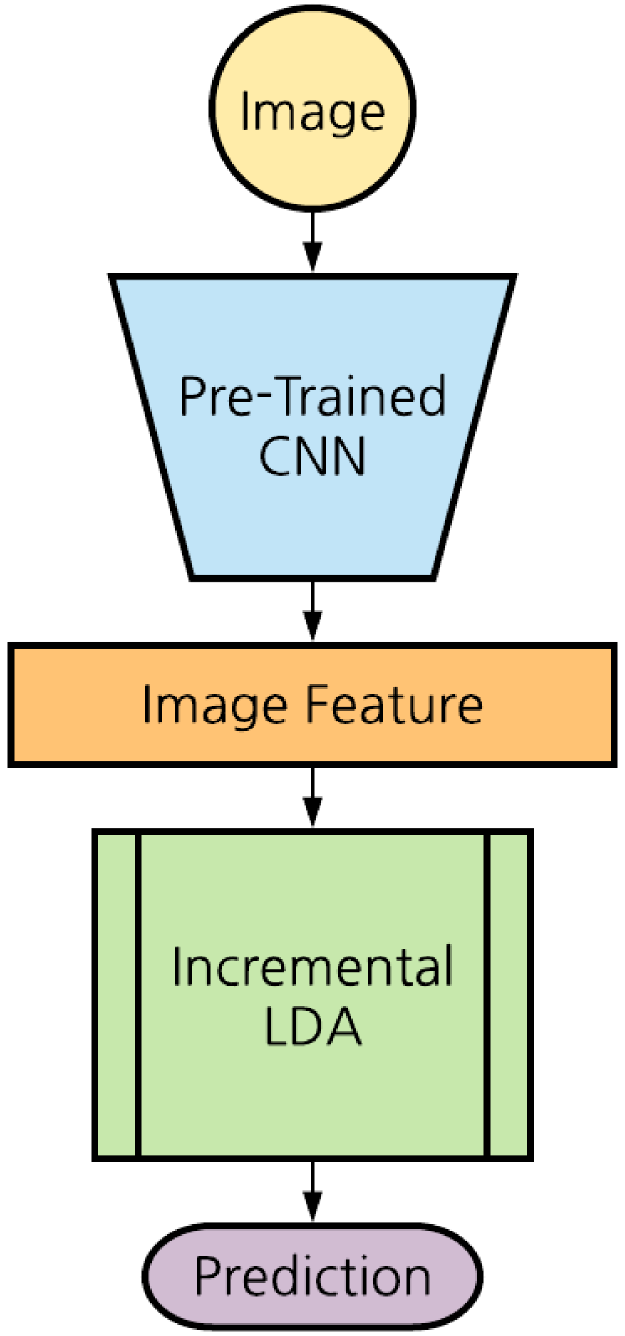
Online Streaming Learning

- Learner receives training instances one at a time, is only allowed one loop through the entire dataset, and can be evaluated at any time during training
- **Advantages:**
  - Closer to how humans/animals learn
  - New instances are learned immediately, meaning the agent can be evaluated immediately
  - Better suited for real-time applications



Deep Streaming Linear Discriminant Analysis

1. Extract image feature from **pre-trained deep CNN**
2. Update **class-specific running mean vector** and running **shared covariance matrix** among classes
3. During inference, a prediction is made by assigning the label of the **closest Gaussian in feature space** defined by the class mean vectors and covariance matrix



Rochester Institute of Technology

CVPR SEATTLE WASHINGTON JUNE 16-18 2020

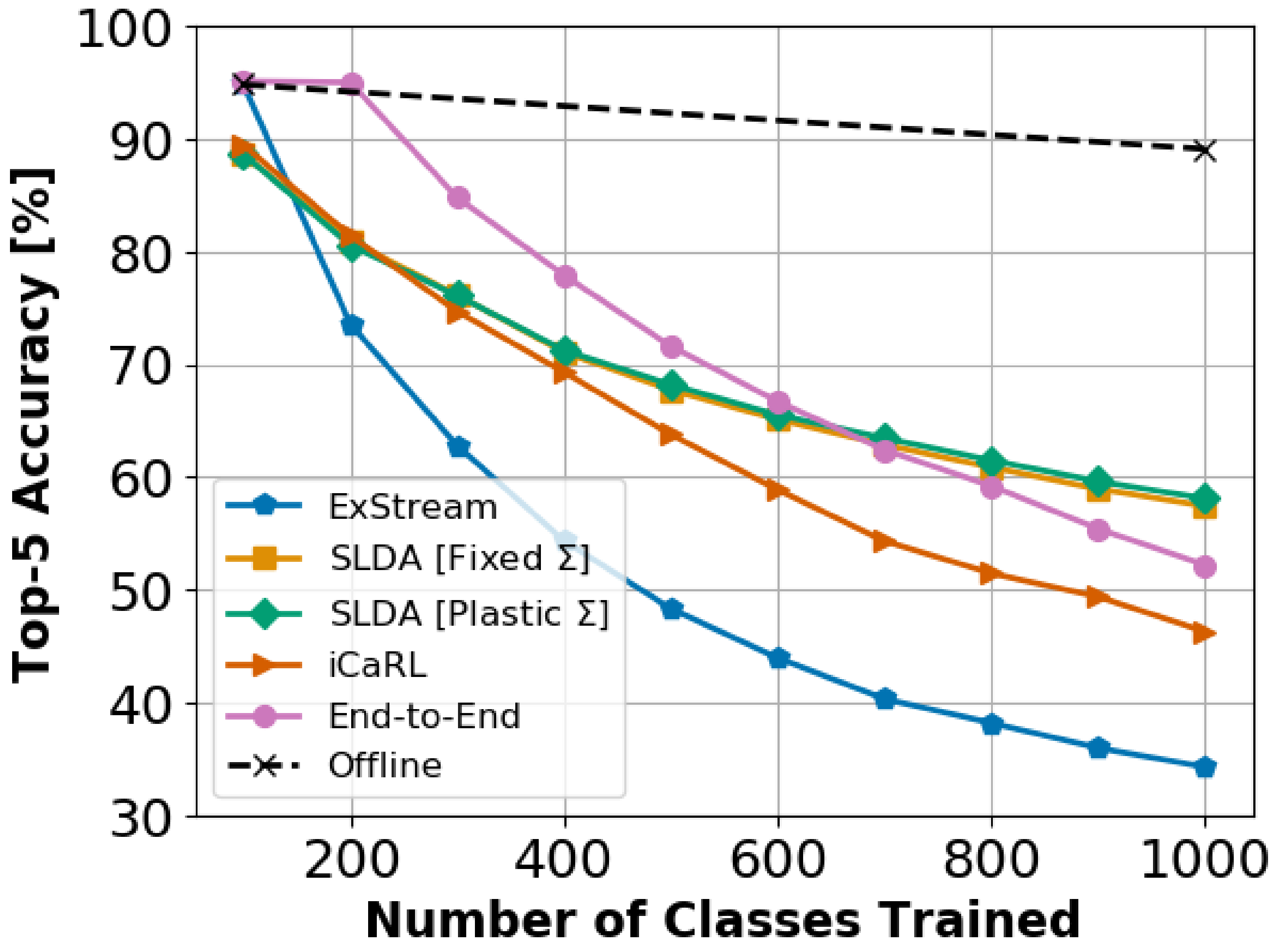
Example CORe50 Videos

Experimental Setup

- We evaluate performance on:
  - **ImageNet ILSVRC-2012**: 1,000 classes with ~1.28 million images
  - **CORe50**: 10 classes with 6,000 images
- We compare several **single-headed** baselines (i.e., not task aware):
  - **SLDA** (streaming): two versions - plastic covariance and fixed covariance
  - **ExStream** (streaming): uses stream clustering and replay
  - **iCaRL** (incremental batch): popular baseline that combines replay with a distillation loss and nearest class mean classifier
  - **End-to-End** (incremental batch): state-of-the-art model that extends iCaRL by using several data augmentation schemes and a neural network classifier
  - **Offline** (upper bound): an optimized offline learner

Results on ImageNet ILSVRC-2012

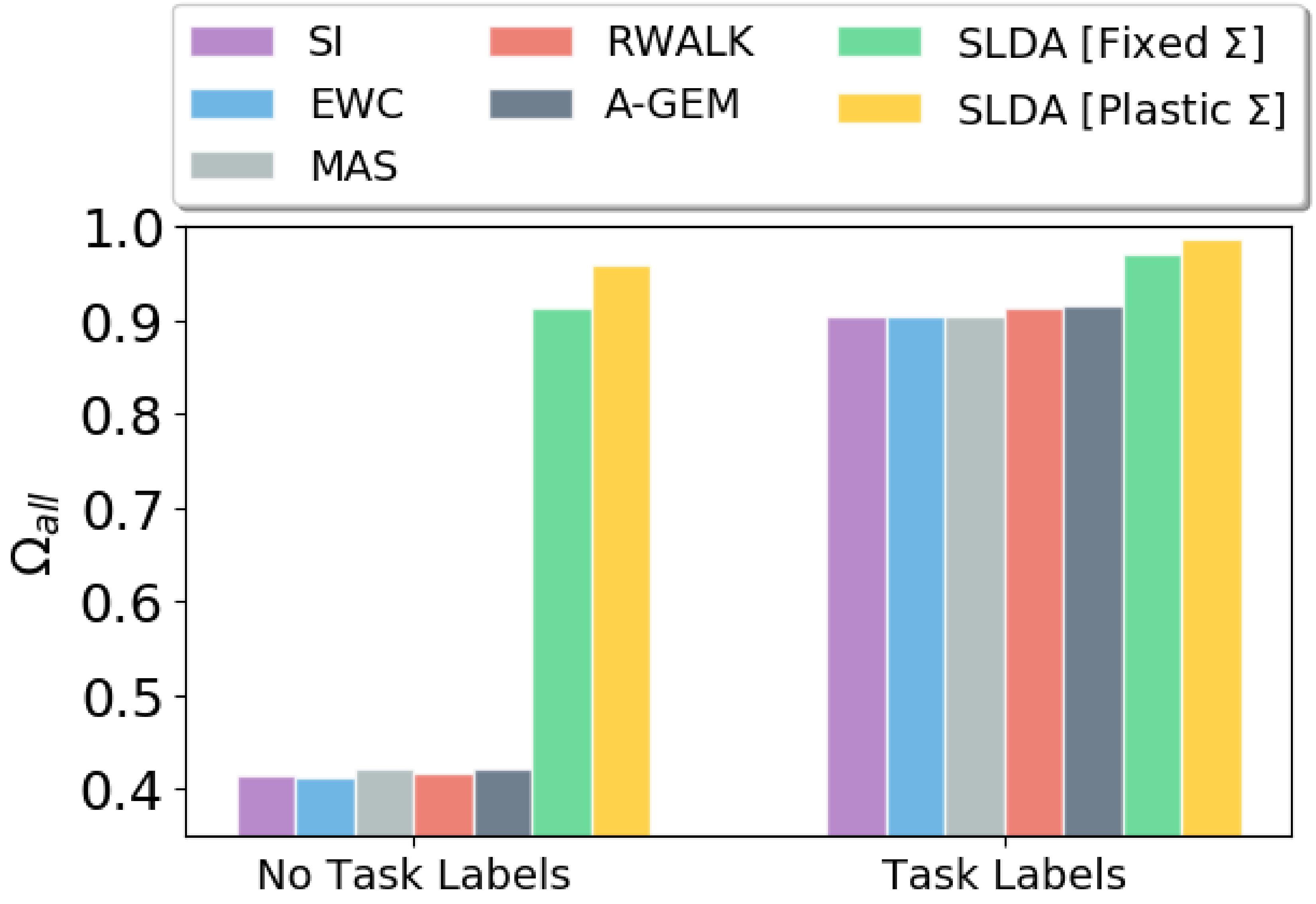
- Models use ResNet-18 and same initialization
- SLDA achieves the **best final top-5 accuracy on ImageNet**, while running over **100 times faster** and using **1,000 times less memory** than the iCaRL and End-to-End models
  - Requires only 30 minutes to train classes 100-1,000
  - Requires only 0.001 GB of storage beyond the CNN



Results on CORe50

- We compare **multi-headed** regularization baselines that require the task label during inference
- We show task-aware methods perform poorly when task labels are withheld during testing
- **SLDA outperforms task-aware methods** both with and without task labels

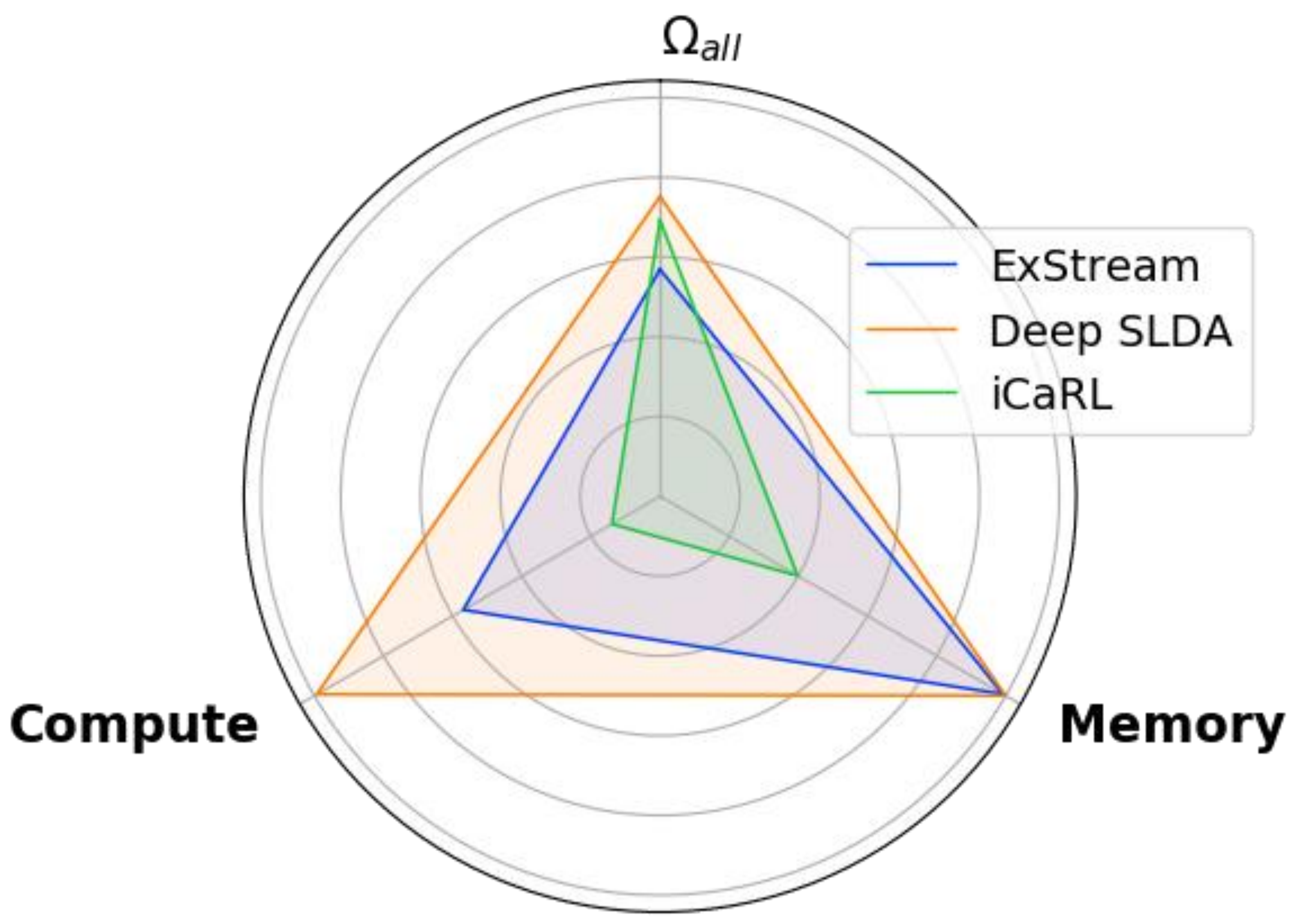
Percentage of performance with respect to offline baseline



Conclusions

- SLDA is popular in the data mining community, but has not been used recently for incremental learning from large image classification datasets
- Using features from a **pre-trained deep CNN**, we show that SLDA outperforms incremental batch learning models, while being more **lightweight**
- While incremental representation learning may improve performance further, we urge future developers to compute performance of **only training the output layer** to ensure gains are being realized

Comparison of overall performance, memory, and compute on ImageNet



Sponsors

This work was supported in part by NSF award #1909696, the DARPA/MTO Lifelong Learning Machines program [W911NF-18-2-0263], and AFOSR grant [FA9550-18-1-0121]. The views and conclusions contained herein are those of the authors and should not be interpreted as representing the official policies or endorsements of any sponsor.