Cost-effective On-device Continual Learning over Memory Hierarchy with Miro

Xinyue Ma¹, Suyeon Jeong¹, Minjia Zhang², Di Wang, Jonghyun Choi³, Myeongjae Jeon¹

¹UNIST ²Microsoft ³Yonsei University
On-device Continual Learning

*data drifts*: data distribution changes over time, creating unseen tasks

**Human activity recognition**

- New real-life activities and gestures

**Video analytics**

- Unseen objects, scenes, and lighting conditions

**On-device Learning is essential:**

- Protect privacy-sensitive data
- Promptly adapt to new data for customization ➔ Through Continual Learning!
On-device Continual Learning

*Learning Incrementally* as new data becomes available

- **Stream (A)**
  - Model for A
  - Accuracy(A): 90%

- **Stream (B)**
  - Model for A, B
  - Accuracy(A): 60%

- C, D, E, ...
  - Accuracy(A): 20%

*Forgetting*: Previously learned knowledge gradually fades away
Remembering through Episodic Memory (EM)

Training on both new and old data

- Designed for server computing
- No sufficient consideration on energy-efficiency
A System Approach: Hierarchical Episodic Memory (HEM)

Old data stored in RAM and **storage**

**Training Step 1**

- Slow, Economic Secondary Storage (e.g., $20 for a 256 GB SD Card)
- Fast, Expensive RAM

**Training Step 2**

**Random Swapping**

Higher **data diversity** ➔ Higher accuracy

![Diagram showing data streams and episodic memory modules](image)
A System Approach: Hierarchical Episodic Memory (HEM)

Old data stored in RAM and storage

Training Step 1
Training Step 2

Can we improve HEM at runtime, considering more system resources to make it more cost-effective?

Higher data diversity ➔ Higher accuracy
Cost-effectiveness in HEM

More resources used ➔ More energy spent
➔ Higher accuracy

Config C > Config B > Config A
The Memory Buffers

Memory Buffers (Stream + EM) is related to GPU and Memory

Energy $\propto$ Memory Buffer Sizes

*Without modifying the model*
Better Allocation for a Fixed Budget

- Best allocation size differs across memory budgets.
- Larger memory budget does not guarantee better accuracy.
• Some configurations (▲) are better than the others.
• The best configuration changes over time.
Swapping has minimum impact on energy consumption and a low knee point.
Turning Observations to Design Principles

P1. Identifies the best configuration in terms of cost-effectiveness.

P2. Updates the configuration when new tasks arrive.

P3. Configures the swap ratio as high as possible.

+P4. Adapts to changes in resource states in the system.
Overall Architecture of Miro

1. Profiling at Low Overhead (P4)
   - For every task (P2) 7-10% Overhead

2. Find the best configuration (P1)

3. Configuring Swap Ratio (P3)

4. Adapting to changing resources states (P4)
2. Find the Best Configuration

Cost-effectiveness = \frac{\text{Accuracy Gain}}{\text{Energy Spent}} + \text{Cutline: Filtering out unpromising configurations}
3 & 4. Adapting to Changes in Resource States

**Swap Ratio**
- I/O Congestion → Halving the swap ratio

**Memory Buffers (SB+EM)**
- Lower Memory Budget → Blacklist the over-budget configurations

Graphs showing:
- Accuracy (%) vs. Swap Ratio (%)
- Energy (kJ) vs. Accuracy (%) with a cutline indicating new best and cost-efficiency.
Evaluation - Setup

**DATASETS**

10 Tasks
- Image Classification
  - CIFAR100 (50k / 197MB)
- Audio Classification
  - UrbanSound8K (8.7k / 184MB)
- HAR
  - DSADS (9k / 163MB)

20 Tasks
- Image Classification
  - Tiny-ImageNet (100k / 1.1GB)

50 Tasks
- Image Classification
  - ImageNet-1K (1.28m / 146GB)

**SYSTEM PLATFORMS**

- NVIDIA Jetson TX2
- NVIDIA Jetson Xavier NX
Miro is more *cost-effective* across the task domains.
Evaluation – Scalability

ImageNet1k (146GB) split into 50 Tasks

- Consistently higher cost-effectiveness
- Scalability for longer and larger workloads
Conclusion

• We explored the design parameters of HEM and analyzed their impacts on cost-effectiveness.

• We built Miro, a system runtime that dynamically re-configures CL tasks, to achieve cost-effective on-device continual learning.

• Miro shows higher cost-effectiveness on 5 datasets over 3 task domains, and great scalability for longer and larger workloads.
Q&A

Thank you for listening!