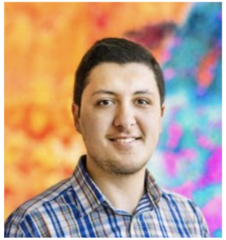
WED-AM-349

Real-Time Evaluation in Online Continual Learning: A New Hope

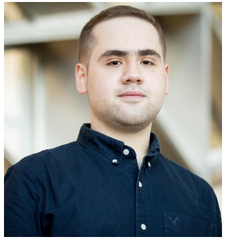
Yasir Ghunaim*
(KAUST)

Adel Bibi*
(Oxford)

Kumail Hammoud
(KAUST)



Motasem Alfarra
(KAUST)



Hasan Hammoud
(KAUST)

Ameya Prabhu
(Oxford)

Philip H. S. Torr
(Oxford)

Bernard Ghanem
(KAUST)

* indicates equal contribution

Introduction

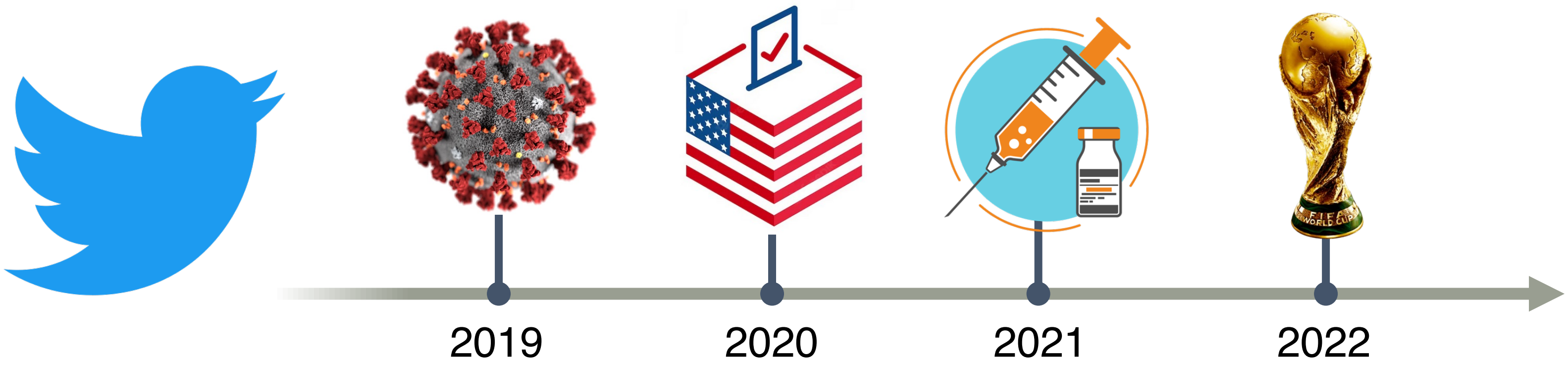
Online Continual Learning (OCL)

Train a model through a single pass over a stream of time-**varying distribution.**

Introduction

Why is OCL important?

Predict misinformation on Twitter's data



Need to continually learn over time!

Image credits: https://health.wyo.gov/publichealth/infectious-disease-epidemiology-unit/disease/novel-coronavirus/, https://www.freepik.com/premium-vector/election-day-usa-voting-vector-logo-icon-template_20456439.htm, https://www.fayettecountypa.org/CivicAlerts.aspx?AID=120&ARC=314, https://skybluetavern.com/2022-fifa-world-cup/

Introduction

Real-time streams are fast!

10x slower than the stream

OCL method

Predict misinformation

Generates

350K tweets/minute

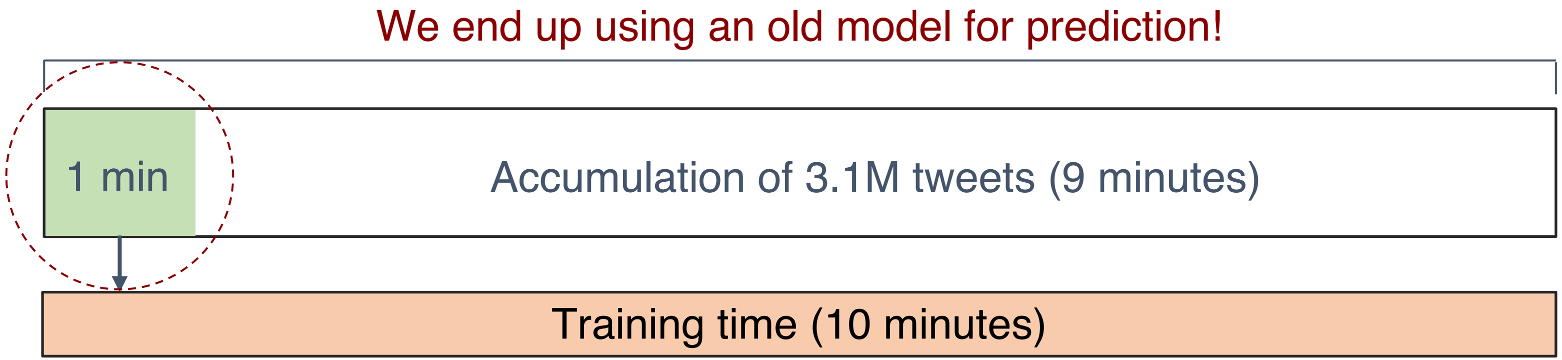
Can train on

35K tweets/minute

David Sayce. The number of tweets per day in 2022, Aug 2022.

Introduction

Real-time streams are fast!



Introduction

Real-time streams are fast!

We end up using an old model for prediction!

Efficient learning is key in real-time OCL applications

1 min

Accumulation of 3.1M tweets (9 minutes)

Training time (10 minutes)

Introduction: Proposal

Current OCL evaluation:

- Allows **unlimited** training computational budget.
- Unfairly compares methods with different training complexities.

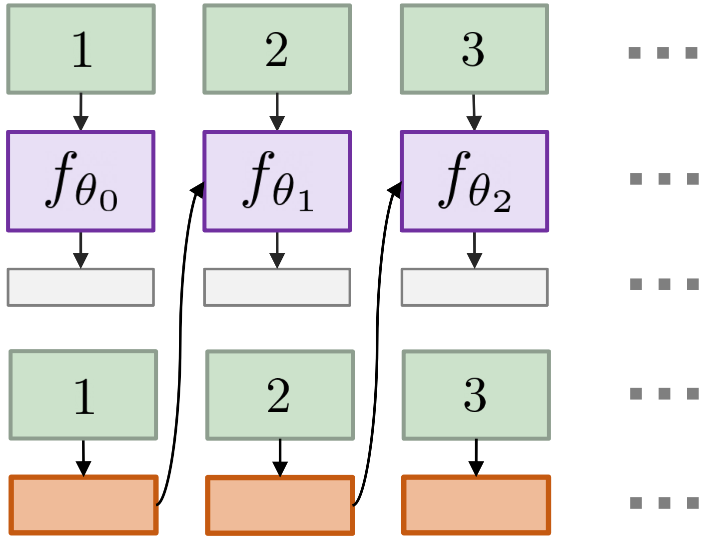
How can we evaluate OCL in a fair manner?

A real-time evaluation protocol for OCL that factors in training computational complexity.

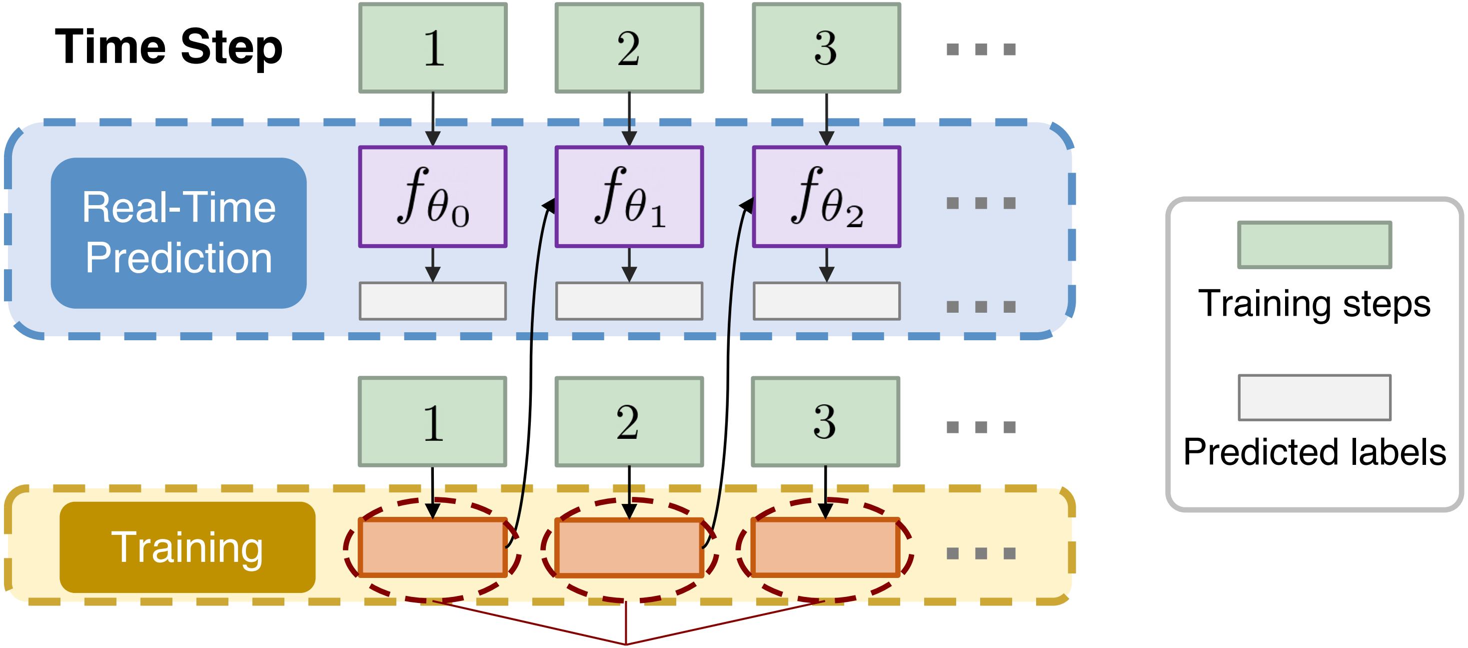
Methodology: Conventional Setup

Learn $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

1. $\mathcal{S}$ reveals $\{x_i^t\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$
2. f_{θ_t} generates $\{\tilde{y}_i^t\}_{i=1}^{n_t}$
3. $\mathcal{S}$ reveals $\{y_i^t\}_{i=1}^{t}$
4. Evaluate $\{\tilde{y}_i^t\}_{i=1}^{n_t}$ with $\{y_i^t\}_{i=1}^{t}$
5. Train f_{θ_t} and update to θ_{t+1}



Methodology: Conventional Setup



Assumes stream will wait for the model to complete training

Methodology: Proposed Setup

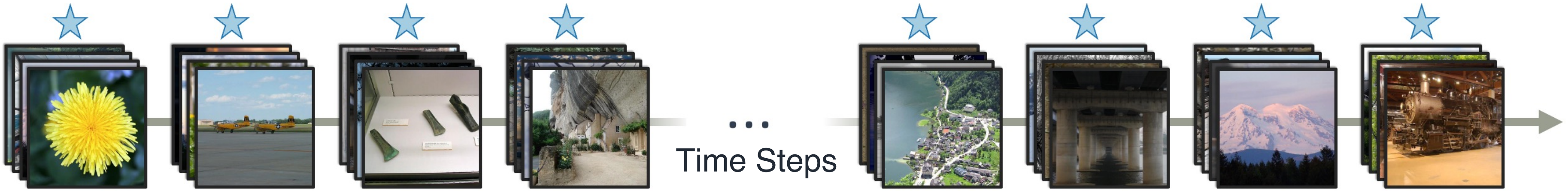
1. **Fast streams** (social media, real-time sensors)

2. **Slow streams** (medical and agriculture applications)

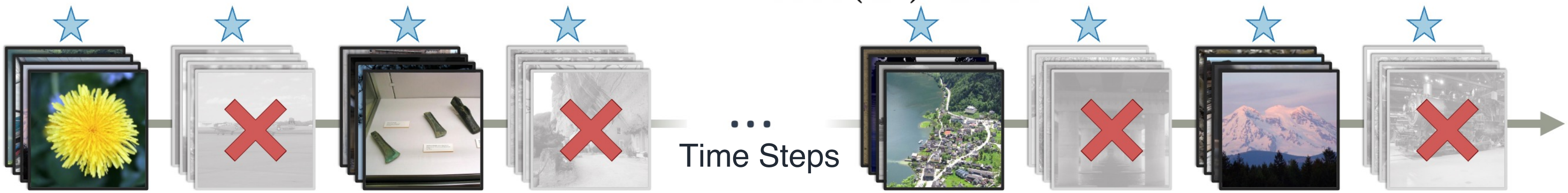
Definition. Given a stream $\mathcal{S}$ and an OCL algorithm $\mathcal{A}$, we define a notion of $\boxed{\text{stream-model relative complexity } \mathcal{C}_{\mathcal{S}}(\mathcal{A})} \in \mathbb{R}^+$, where $\boxed{\mathcal{C}_{\mathcal{S}}(\mathcal{A}) = 1}$ means that a continual learner $\mathcal{A}$ can update the model parameters θ before the stream reveals the data of the next step.

Methodology: Proposed Setup

Method $\mathcal{A}$, $\mathcal{C}_{\mathcal{S}}(\mathcal{A}) = 1$



Method $\mathcal{B}$, $\mathcal{C}_{\mathcal{S}}(\mathcal{B}) = 2$



Used for evaluation   Skipped during training

Methodology: Proposed Setup

Learn $f_\theta : \mathcal{X} \to \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

1. $\mathcal{S}$ reveals $\{x_i^t\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$
2. f_{θ_t} generates $\{\tilde{y}_i^t\}_{i=1}^{n_t}$
3. $\mathcal{S}$ reveals $\{y_i^t\}_{i=1}^{t}$
4. Evaluate $\{\tilde{y}_i^t\}_{i=1}^{n_t}$ with $\{y_i^t\}_{i=1}^{t}$
5. Train f_{θ_t} and update to θ_{t+1}

Need to modify this step to incorporate $\mathcal{C}_\mathcal{S}(\mathcal{A}) = k$

Methodology: Proposed Setup

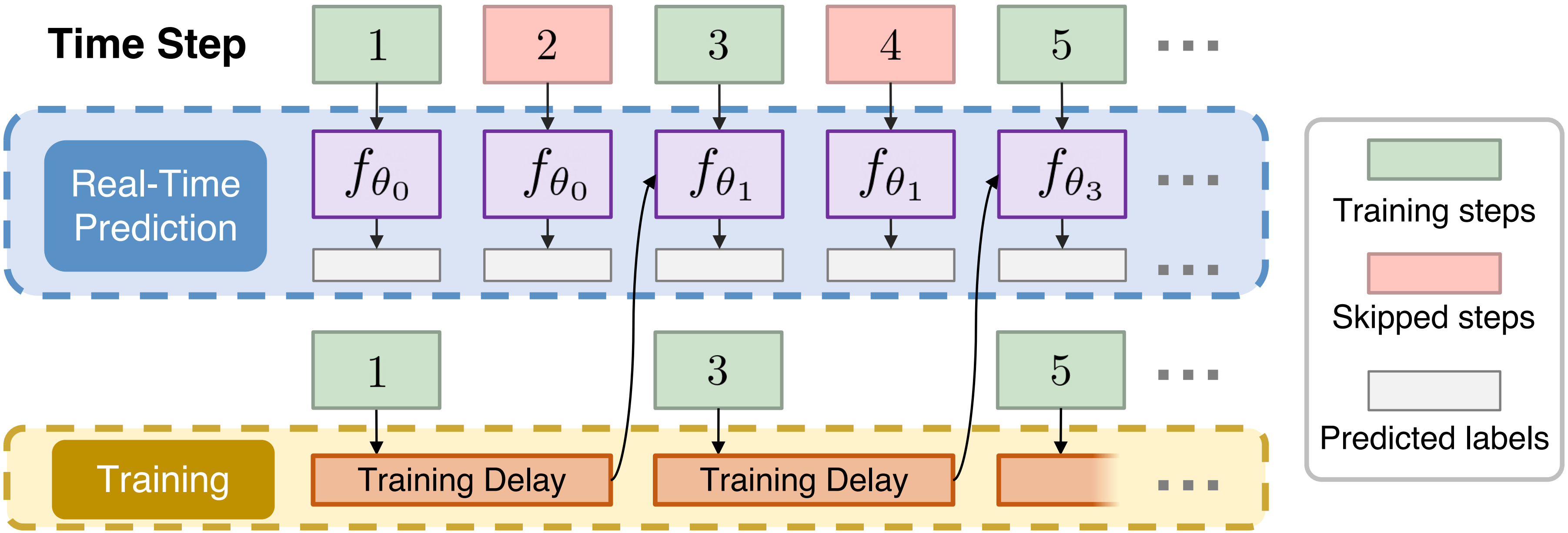
Learn $f_\theta : \mathcal{X} \to \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

1. $\mathcal{S}$ reveals $\{x_i^t\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$
2. f_{θ_t} generates $\{\tilde{y}_i^t\}_{i=1}^{n_t}$
3. $\mathcal{S}$ reveals $\{y_i^t\}_{i=1}^{t}$
4. Evaluate $\{\tilde{y}_i^t\}_{i=1}^{n_t}$ with $\{y_i^t\}_{i=1}^{t}$
5. $\boxed{\text{If } \mathrm{mod}(t-1, k) = 0,}$ train f_{θ_t} and update to θ_{t+1}

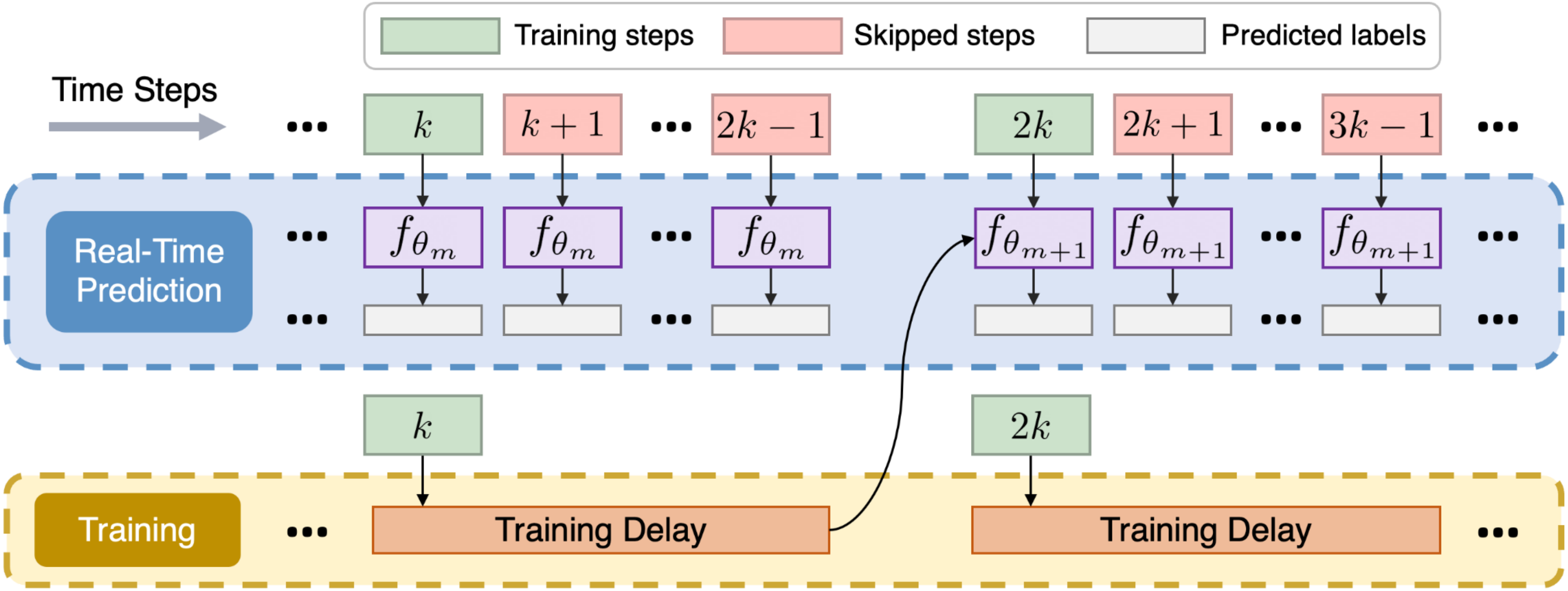
$\mathcal{C}_\mathcal{S}(\mathcal{A}) = k$

Methodology: Proposed Setup

Let's consider a method with: $\mathcal{C}_S(\mathcal{A}) = 2$



Methodology: Proposed Setup



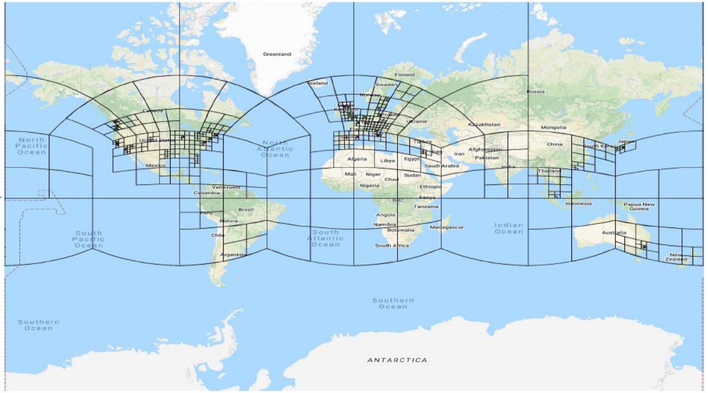
Experiments: The Task

CLOC [1]

Input Image

Predict the Geolocation

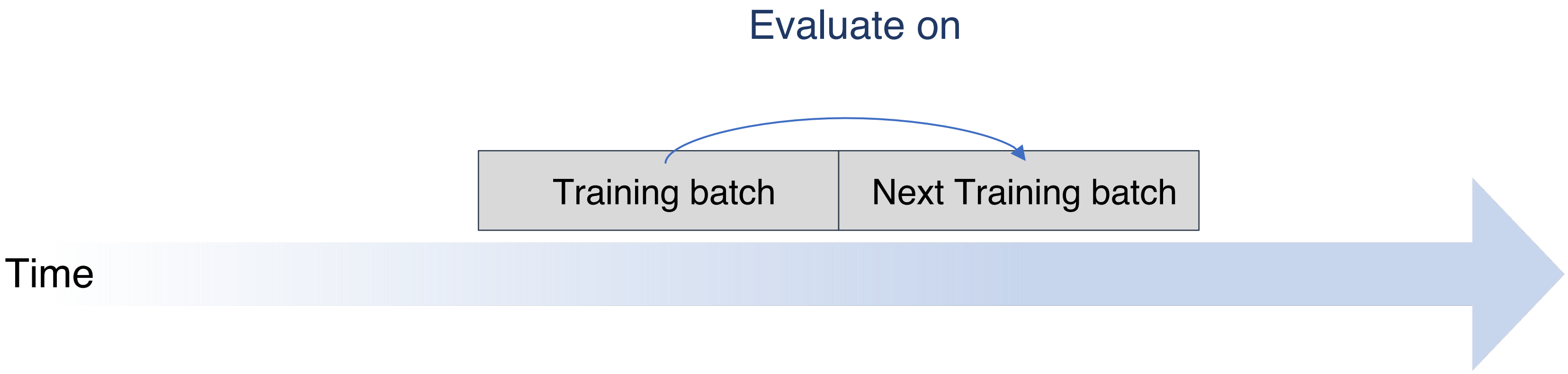
712 Classes (Locations)



[1] Cai, Zhipeng, et al. "Online continual learning with natural distribution shifts: An empirical study with visual data." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021.

Experiments: Metric

Average Online Accuracy: measures the ability of models to adapt to incoming stream samples



Experiments: Methods

CL Strategy	Method($\mathcal{A}$)	$\mathcal{C}_S(\mathcal{A})$	Delay
Experience Replay	ER [1]	1	0
Regularizations	ACE [2]	1	0
	LwF [3]	4/3	1/3
	RWalk [4]	2	1
LR Scheduler	PoLRS [5]	3	2
Sampling Strategies	MIR [6]	5/2	3/2
	GSS [7]	6	5

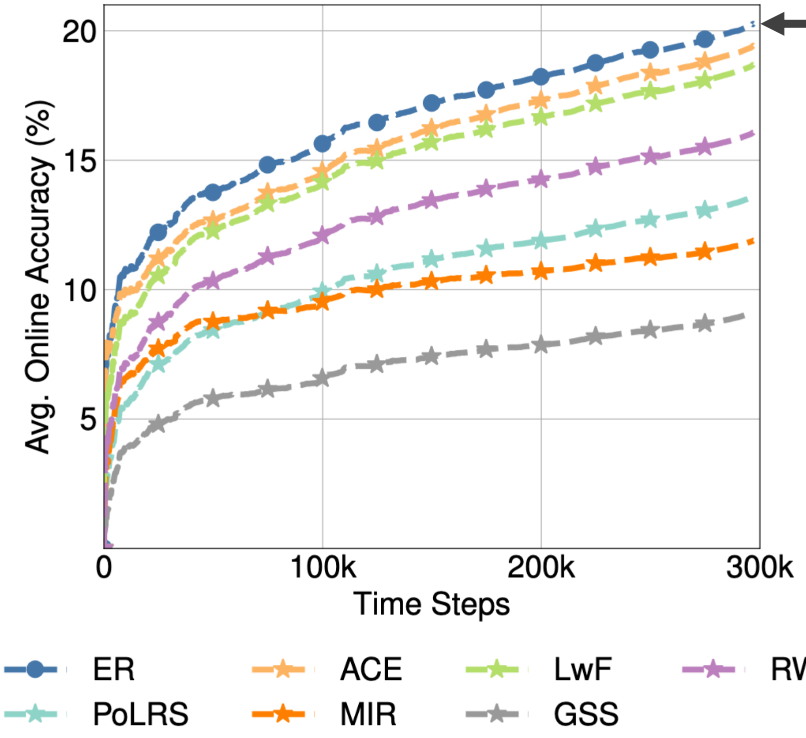
[1] Chaudhry, Arslan, et al. Continual learning with tiny episodic memories. In International Conference on Machine Learning (ICML), 2019
[2] Lucas Caccia, et al. New insights on reducing abrupt representation change in online continual learning. In International Conference on Learning Representations (ICLR), 2022.
[3] Zhizhong Li, et al. Learning without forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE TPAMI), 2017.
[4] Arslan Chaudhry, et al. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In European Conference on Computer Vision (ECCV), 2018.
[5] Zhipeng Cai, et al. Online continual learning with natural distribution shifts: An empirical study with visual data. In International Conference on Computer Vision (ICCV), 2021.
[6] Rahaf Aljundi, et al. Online continual learning with maximally interfered retrieval. In Conference on Neural Information Processing Systems (NeurIPS), 2019.
[7] Rahaf Aljundi, et al. Gradient based sample selection for online continual learning. In Conference on Neural Information Processing Systems (NeurIPS), 2019.

Experiments

What happens when OCL methods are evaluated under our **delayed real-time** evaluation setup?

Experiments: Fast Stream

Method($\mathcal{A}$)	$\mathcal{C}_S(\mathcal{A})$	**Delay**
ER	1	0
ACE	1	0
LwF	4/3	1/3
RWalk	2	1
PoLRS	3	2
MIR	5/2	3/2
GSS	6	5

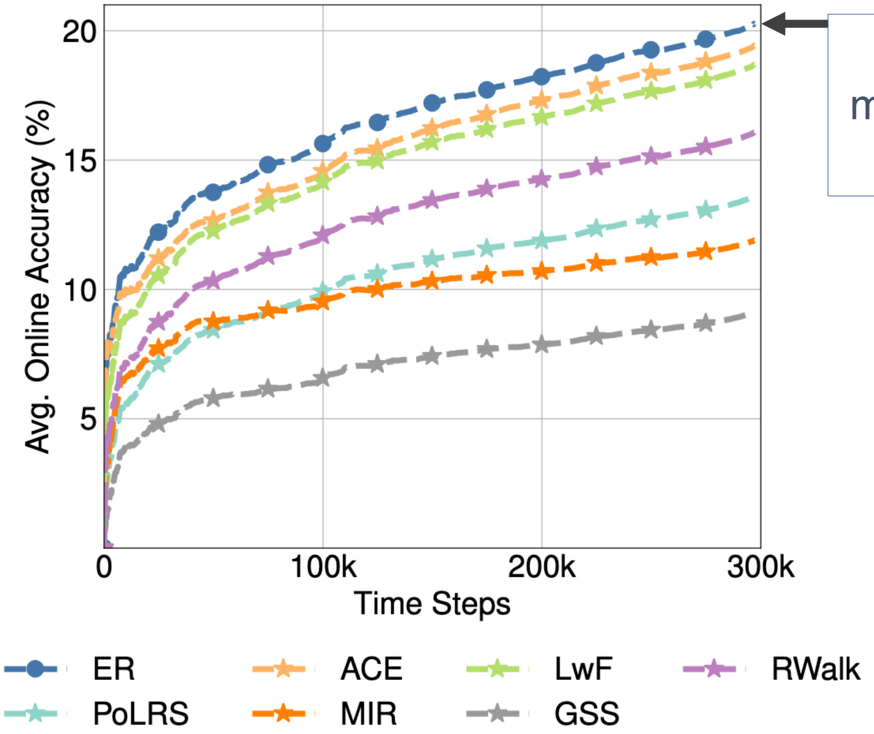


ER, the simplest method, outperforms all considered methods

Experiments: Fast Stream

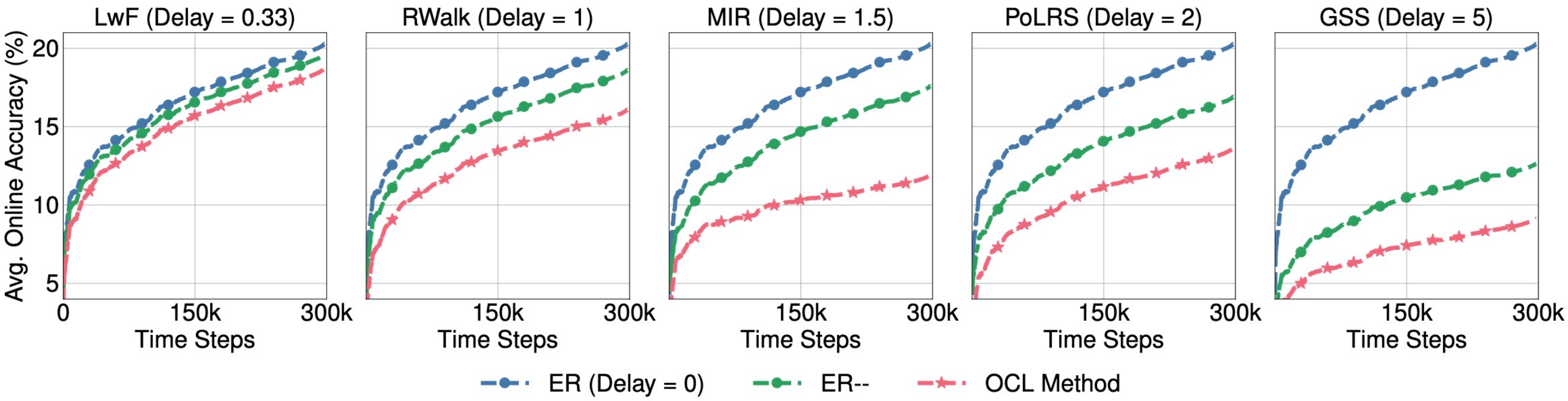
Method($\mathcal{A}$)	$\mathcal{C}_{\mathcal{S}}(\mathcal{A})$	Delay
ER	1	0
ACE	1	0
LwF	4/3	1/3
RWalk	2	1
PoLRS	3	2
MIR	5/2	3/2
GSS	6	5

What if we increase the complexity of ER (baseline) such that it has the same delay as each method?



ER, the simplest method, outperforms all considered methods

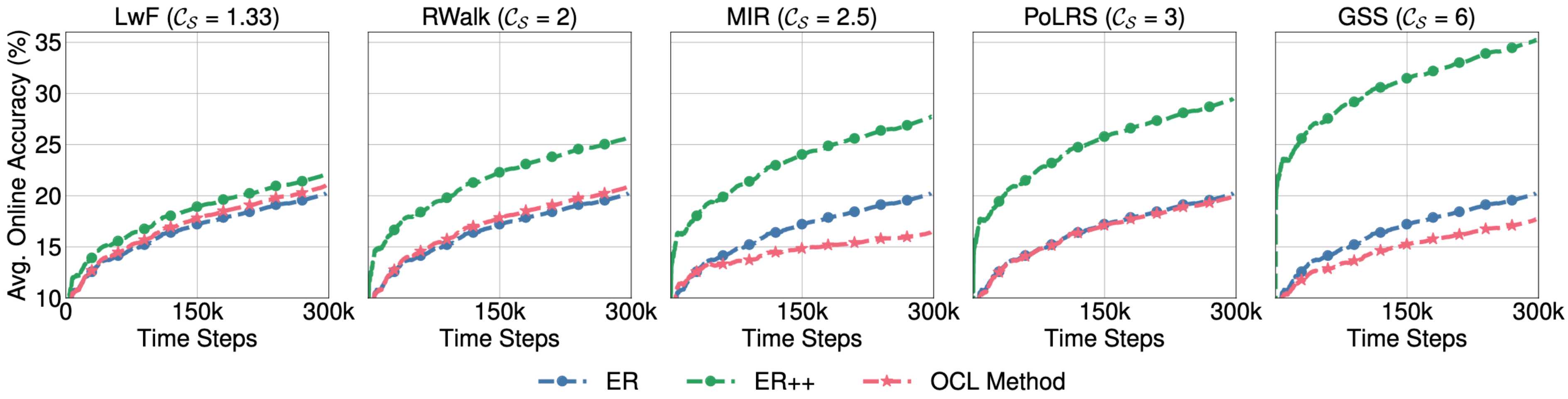
Experiments: Fast Stream – Data Normalization



Experiments

What if the stream is as slow as each OCL method?

Experiments: Slow Stream



Conclusion

- All evaluated methods underperformed the ER baseline in our realistic setup.
- OCL research should consider training efficiency in evaluations.