Continual Learning

Introduction to Deep Continual Learning

Antonio Carta

antonio.carta@unipi.it

Outline

- Module Structure
- Definition of Deep Continual Learning
- Catastrophic Forgetting

Offline Learning

Limitations of offline learning

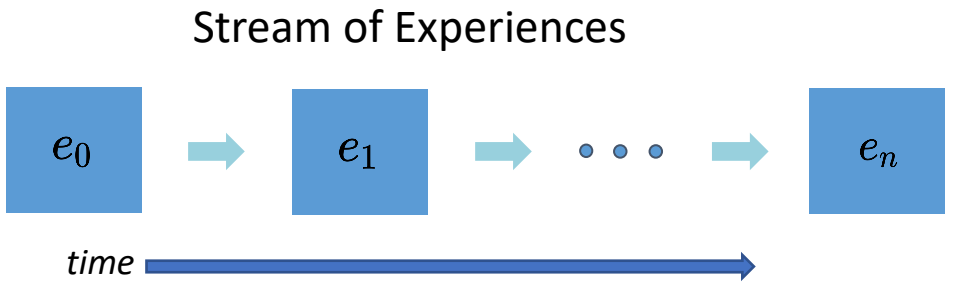
- sustainability, efficiency
- Learning in low data scenarios
- Privacy: learning when we cannot store the data
- Distributed learning: multiple devices with privacy/efficiency constraints

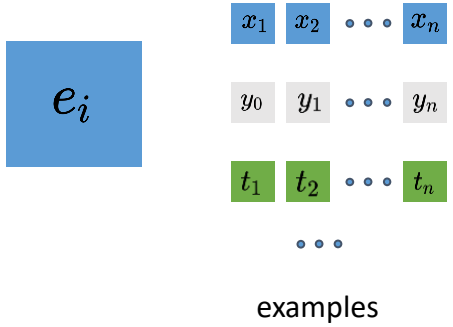
Deep Continual Learning

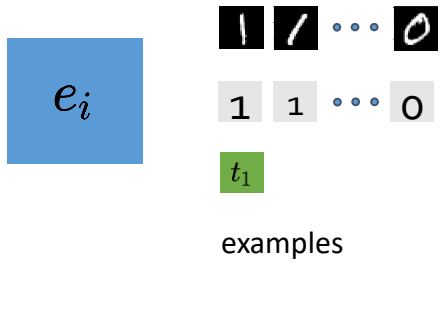
Learning from a non-stationary stream with Deep Neural Networks

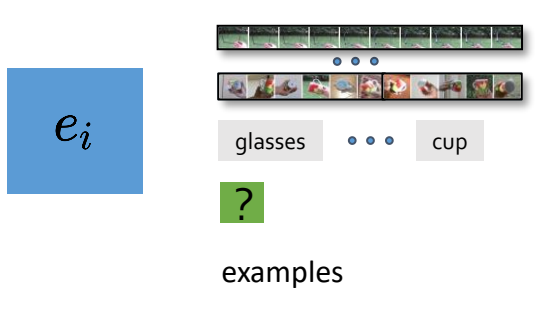
- How to learn incrementally with DNN
- Without forgetting old tasks
- We revisit the previous problems in a more challenging setting:
  - Multi-Task Learning -> Task-Incremental Learning
  - Online with prequential evaluation -> evaluation on the entire stream (i.e. never forget)

Continual Learning









Continual Learning - Definition

In continual learning (CL) data arrives in a streaming fashion as a (possibly infinite) sequence of learning experiences $S = e_1, \ldots, e_n$. For a supervised classification problem, each experience e_i consists of a batch of samples $\mathcal{D}^i$, where each sample is a tuple $\langle x_k^i, y_k^i \rangle$ of input and target, respectively, and the labels y_k^i are from the set $\mathcal{Y}^i$, which is a subset of the entire universe of classes $\mathcal{Y}$. Usually $\mathcal{D}^i$ is split into a separate train set $\mathcal{D}_{train}^i$ and test set $\mathcal{D}_{test}^i$.

A continual learning algorithm $\mathcal{A}^{CL}$ is a function with the following signature:

$$\mathcal{A}^{CL} : \langle f_{i-1}^{CL}, \mathcal{D}_{train}^i, \mathcal{M}_{i-1}, t_i \rangle \rightarrow \langle f_i^{CL}, \mathcal{M}_i \rangle \quad (1)$$

where f_i^{CL} is the model learned after training on experience e_i, $\mathcal{M}_i$ a buffer of past knowledge, such as previous samples or activations, stored from the previous experiences and usually of fixed size. The term t_i is a task label which may be used to identify the correct data distribution.

The objective of a CL algorithm is to minimize the loss $\mathcal{L}_S$ over the entire stream of data S:

$$\mathcal{L}_S(f_n^{CL}, n) = \frac{1}{\sum_{i=1}^{n} |\mathcal{D}_{test}^i|} \sum_{i=1}^{n} \mathcal{L}_{exp}(f_n^{CL}, \mathcal{D}_{test}^i) \quad (2)$$

$$\mathcal{L}_{exp}(f_n^{CL}, \mathcal{D}_{test}^i) = \sum_{j=1}^{|\mathcal{D}_{test}^i|} \mathcal{L}(f_n^{CL}(x_j^i), y_j^i), \quad (3)$$

where the loss $\mathcal{L}(f_n^{CL}(x), y)$ is computed on a single sample $\langle x, y \rangle$, such as cross-entropy in classification problems.

Continual Learning for Robotics: Definition, Framework, Learning Strategies, Opportunities and Challenges, Lesort et al. Information Fusion, 2020.

CL and Neuroscience

“We are not looking for incremental improvements in state-of-the-art AI and neural networks, but rather paradigm-changing approaches to machine learning that will enable systems to continuously improve based on experience.”

— Hava Siegelmann, 2018

CL Desiderata

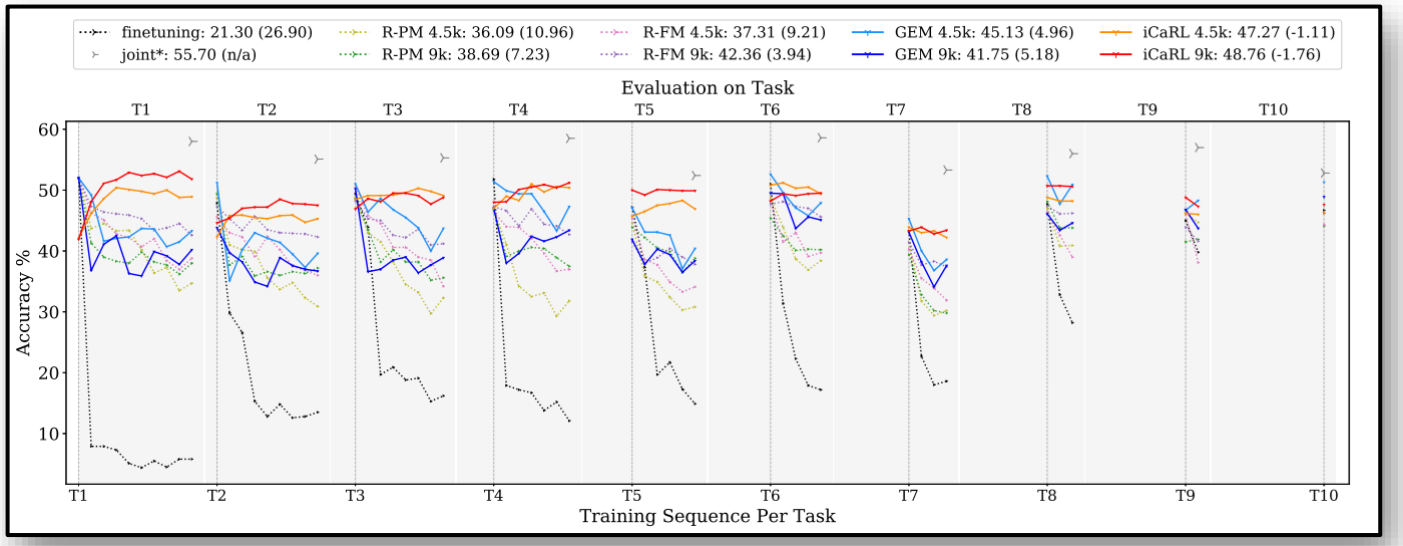
- **Lifelong timescales**: learning incrementally on long (lifelong) timescales is a fundamental property of intelligent systems.
- **Limited memory**: No (or limited) access to previously encountered data.
- **Limited computation**: Computational cost is limited over time and it doesn't grow too much.
- **Incremental learning**: the model continually improves over time.

Efficiency + Scalability = Sustainability

Challenges

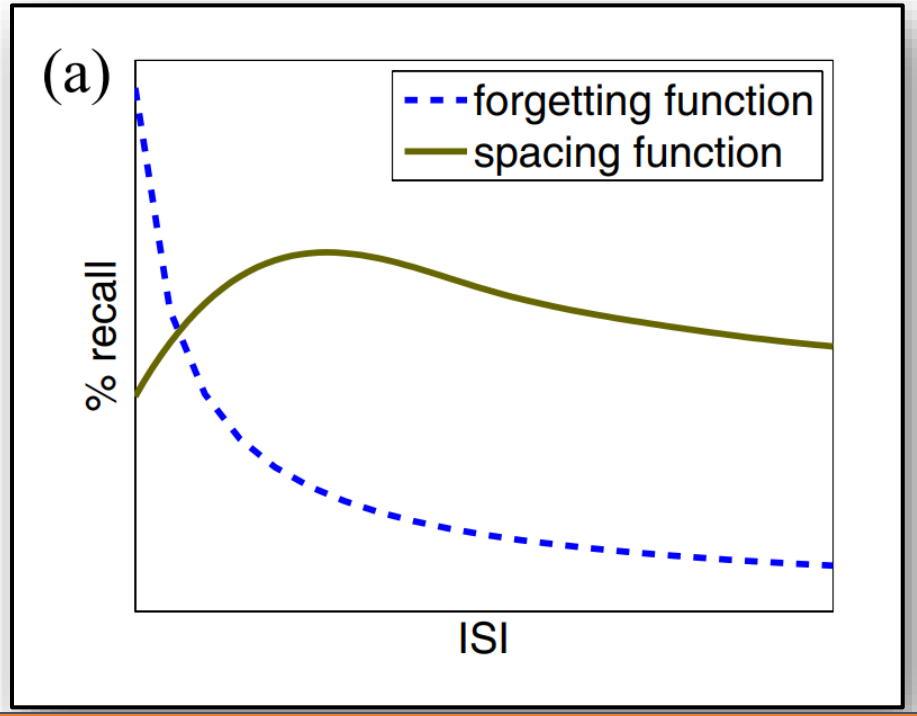
Catastrophic Forgetting

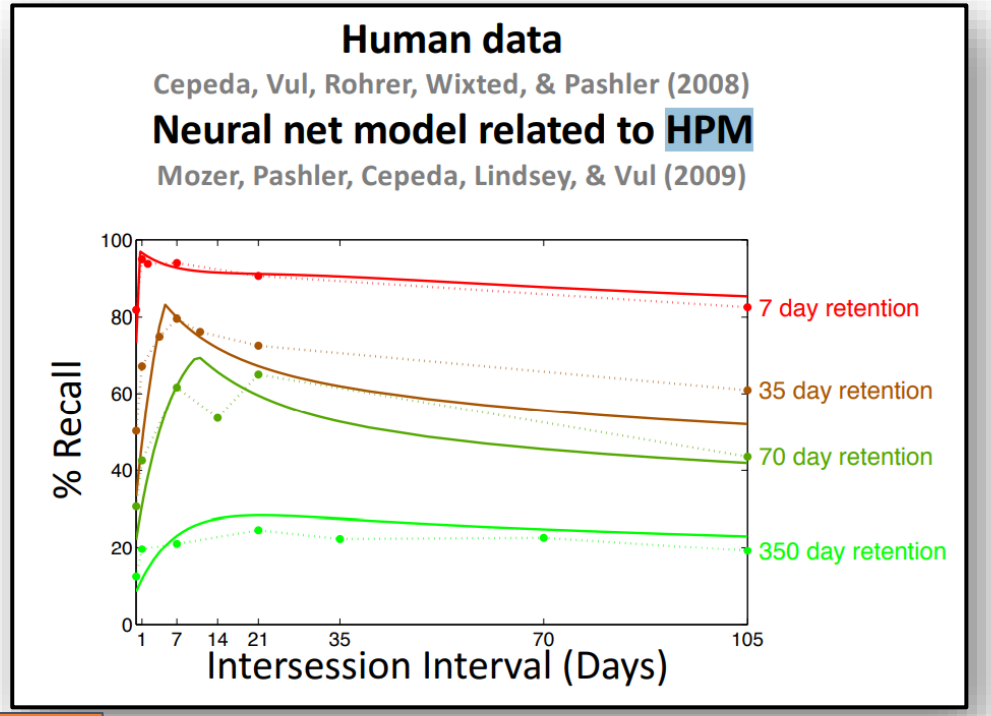
***Catastrophic interference**, a.k.a. catastrophic forgetting, is the tendency of deep neural networks to completely and abruptly forget previous tasks/domains when learning new ones*



A Continual Learning Survey: Defying Forgetting in Classification Tasks by M. De Lange, TPAMI 2021.

Forgetting in Humans





The spacing function (solid line) depicts recall at test following two study sessions separated by a given intersession interval (ISI); the forgetting function (dashed line) depicts recall as a function of the lag between study and test.

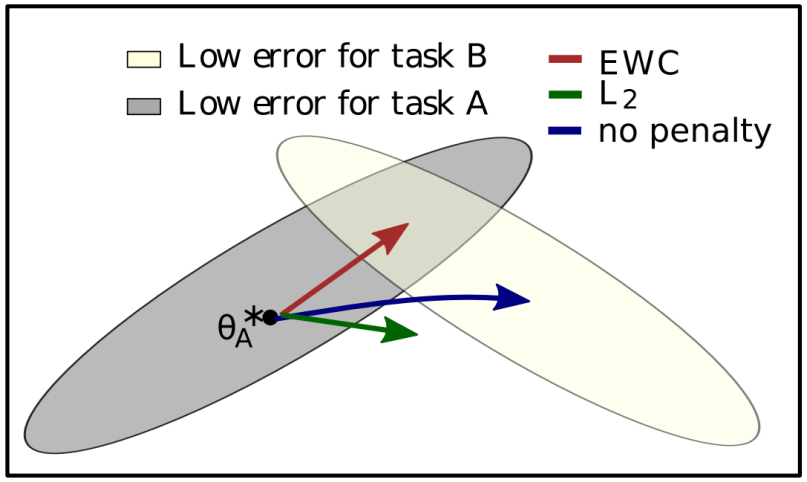
Predicting the optimal spacing of study: A multiscale context model of memory. M. Mozer et al. NeurIPS, 2009.

Forgetting – EWC

How can we approximate the loss on the previous task/domain without access to the data?

L_s is made of two parts:

- The loss on current data, which we can compute
- The loss on previous data, which we CANNOT compute because we don't have the data



The objective of a CL algorithm is to minimize the loss $\mathcal{L}_S$ over the entire stream of data S:

$$\mathcal{L}_S(f_n^{CL}, n) = \frac{1}{\sum_{i=1}^{n} |\mathcal{D}_{test}^i|} \sum_{i=1}^{n} \mathcal{L}_{exp}(f_n^{CL}, \mathcal{D}_{test}^i) \quad (2)$$

$$\mathcal{L}_{exp}(f_n^{CL}, \mathcal{D}_{test}^i) = \sum_{j=1}^{|\mathcal{D}_{test}^i|} \mathcal{L}(f_n^{CL}(\boldsymbol{x}_j^i), y_j^i), \quad (3)$$

where the loss $\mathcal{L}(f_n^{CL}(\boldsymbol{x}), y)$ is computed on a single sample $\langle \boldsymbol{x}, y \rangle$, such as cross-entropy in classification problems.

Overcoming catastrophic forgetting in neural networks. James Kirkpatrick et al. Proceedings of the national academy of sciences, 2017.

Stability-Plasticity Dilemma

- **Stability**: the ability to avoid forgetting of old task/domains
- **Plasticity**: the ability to learn new tasks
- In general, there is a tradeoff between the two properties
  - I can freeze the network to prevent forgetting
  - I can do a naive finetuning (or even randomly initialize) to have optimal plasticity

Q: How can we limit forgetting?

Module Organization

Lectures

- Deep Continual Learning and Catastrophic Forgetting (today)
- Scenarios, Evaluation, Metrics
- Methodologies – Baselines and Replay
- Methodologies – Regularization
- Methodologies – Architectural methods and incremental classifiers
- Tools and Applications (guest lecture + seminars)

We will also have some labs where we apply the methodology

Scenarios, Evaluation, Metrics

- **Scenarios**: types of continual learning problems and how to categorize them
- **Evaluation**: how to evaluate CL models
- **Metrics**: how to measure transfer, forgetting, computational cost and their trend over time

Methodologies – Baselines and Replay

- **Baselines**: naive finetuning, cumulative training, model freezing …
- **Replay**: storing past samples
  - Insertion/deletion/update policies

Methodologies – Regularization

Regularization:

- Can we approximate the loss on the past data without looking at the original data?
- Can we prevent forgetting when we update the model?

Architectural M. and Incremental Classifiers

Architectural Methods:

- Modular architectures
- Masking and sparse networks
- We will revisit MTL in a CL setting

Incremental Classifiers

- Classifier bias towards new classes
- How to fix it

General References

- CL Course: https://course.continualai.org/
  - Ph.D. level but it's recorded and it has most (but not all) of the topics covered here
- Avalanche tutorials: https://avalanche.continualai.org/from-zero-to-hero-tutorial/01_introduction

Next

- Notebooks:
  - Avalanche e2e
  - Forgetting
  - Avalanche standalone