Lecture 10 - Meta-Learning and Continual Learning

Advanced Machine Learning

Petr Šimánek

FIT CTU

Lecture Overview

1. X-learning
2. Introduction to Meta-Learning
3. Model-Agnostic Meta-Learning (MAML)
4. Introduction to Continual Learning
5. Elastic Weight Consolidation (EWC)
6. Progressive Neural Nets
7. Experience replay
8. Hypertransformers

X-learning

What is...

- Transfer learning
- Multitask learning
- Meta-learning
- Continual learning

Transfer Learning

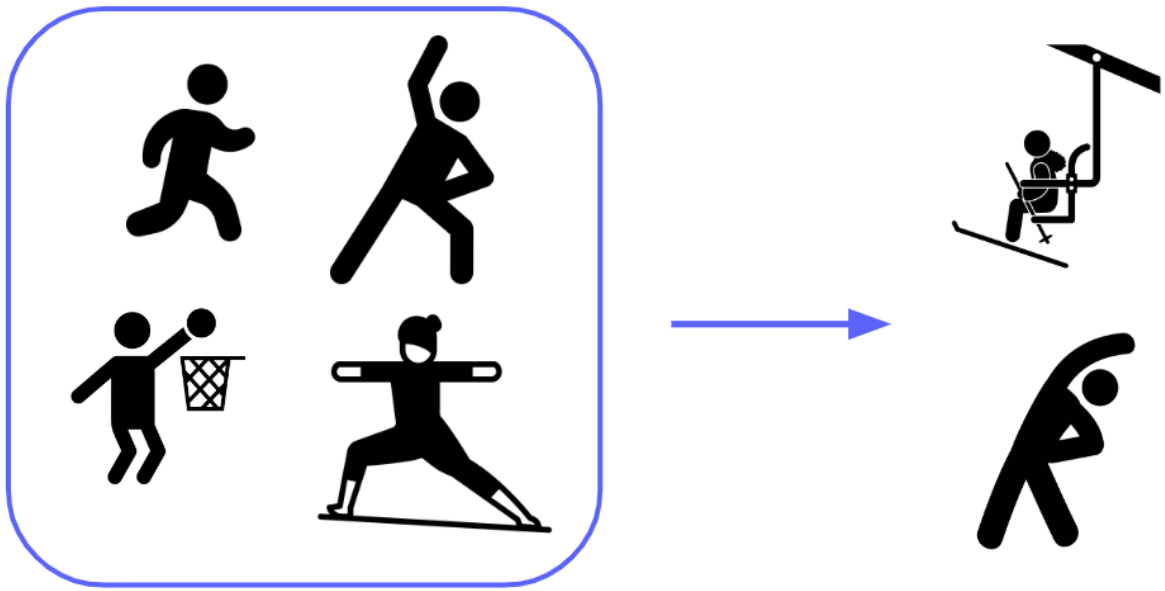


Figure: Transfer Learning

Fine-tuning of a pre-trained network. Important for LLMs - methods like LORA, DORA, Adapters.

Multitask Learning

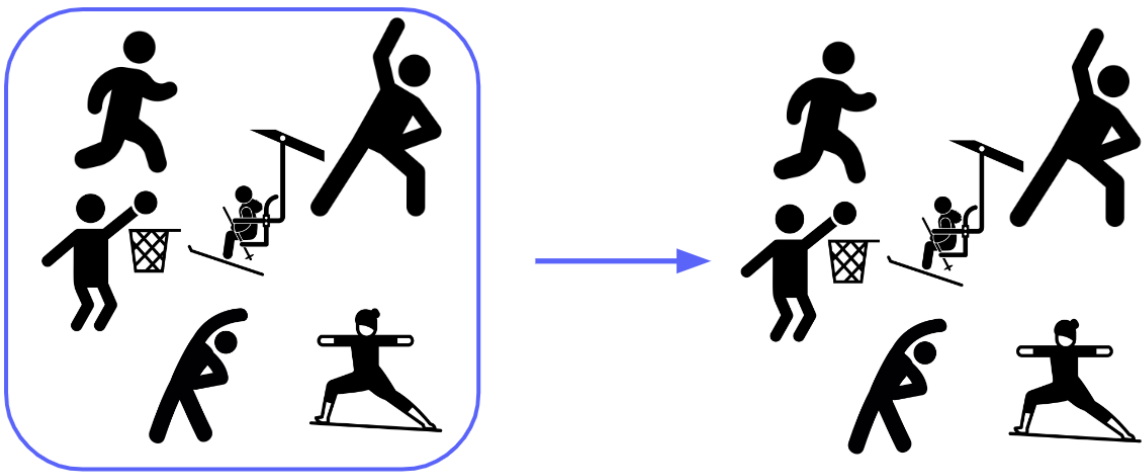


Figure: Multitask Learning

Training a larger NN, hopefully learning different tasks at the same time would be of benefit.

Meta-Learning

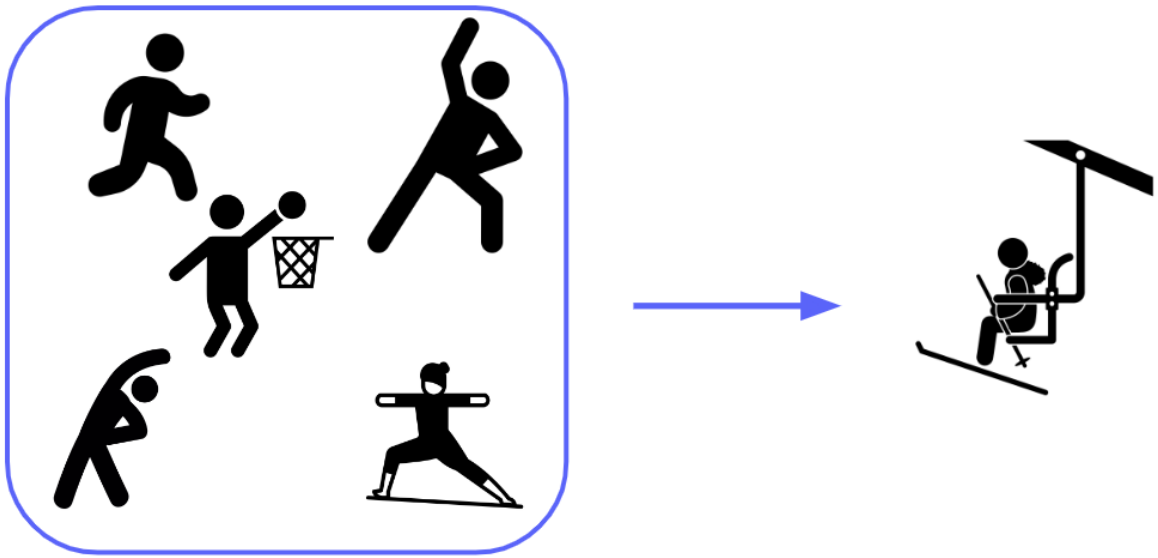


Figure: Meta-Learning

Training to quickly learn a new task.

Continual Learning

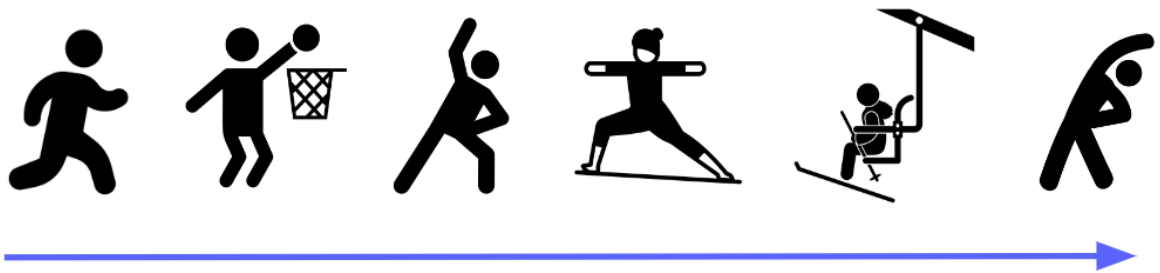


Figure: Continual Learning

Training tasks one by one.

Introduction to Meta Learning

- Definition: Learning to learn; training a model to adapt quickly to new tasks with minimal data
- Main goal: Reduce the need for extensive retraining and improve generalization
- Meta-learning enables AI systems to leverage prior knowledge to learn new tasks quickly, making them more versatile and data-efficient
- Two main approaches: metric-based and optimization-based
- Real-world significance and applications: meta-learning can be applied to various domains, including computer vision, natural language processing, and reinforcement learning

Metric-Based Meta Learning

- Learn a similarity metric between tasks to facilitate knowledge transfer and adaptation
- Examples:
  - Siamese Networks (finding similarity)
  - Matching Networks (embedding + differentiable KNN)
- Advantages: Simple, interpretable, and easily applicable to few-shot learning problems
- Limitations: May not be optimal for tasks that require learning complex relationships or updating mechanisms

Optimization-Based Meta Learning

- Learn an initialization or update rule to adapt quickly to new tasks
- Examples:
  - MAML: Model-Agnostic Meta-Learning
  - Reptile
  - FOMAML, iMAML
  - Meta-SGD
- Advantages: Generalizable across different learning tasks and model architectures
- Limitations: Can be computationally expensive, especially for models with a large number of parameters

Model-Agnostic Meta-Learning (MAML)

- Introduced by Finn et al. (2017)
- Optimization-based approach
- Learn an initial set of parameters that can be fine-tuned efficiently for new tasks
- Compatible with any model that can be trained using gradient-based optimization

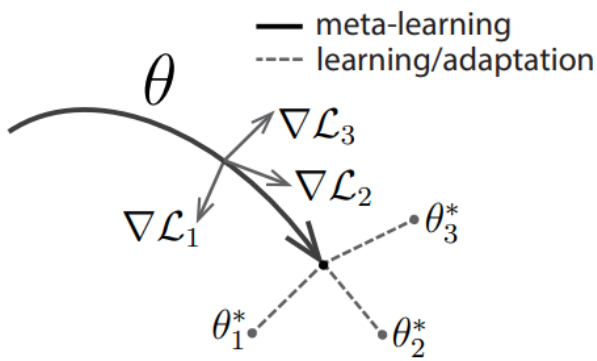


Figure: MAML

MAML Algorithm

MAML algorithm overview:

1. Initialize the model parameters θ
2. Sample a batch of tasks
3. For each task:
   - 3.1 Compute task-specific parameters using gradient descent on the task loss
   - 3.2 Evaluate the task-specific parameters on the task validation set
4. Update the initial parameters θ using the gradients of the validation loss with respect to θ
5. Repeat steps 2-4 for a fixed number of meta-iterations

MAML: Loss Function and Gradients

- For each task i, let $L_i(\theta)$ denote the task-specific loss function
- Perform one or more steps of gradient descent on $L_i(\theta)$ to obtain task-specific parameters $\theta_i' = \theta - \alpha \nabla_\theta L_i(\theta)$
- Here, α is the inner learning rate (task-specific learning rate)
- The meta-objective is to minimize the sum of validation losses across tasks: $\sum_i L_i^{\text{val}}(\theta_i')$
- Update the initial parameters θ with gradient descent on the meta-objective using the outer learning rate (meta-learning rate): $\theta \leftarrow \theta - \beta \nabla_\theta \sum_i L_i^{\text{val}}(\theta_i')$
- The process is repeated for a fixed number of meta-iterations

MAML: Key Insights and Limitations

- Insights:
  - Importance of good initialization
  - Role of second-order gradients
  - Compatibility with various learning tasks
- Limitations:
  - Computational complexity
  - Sensitivity to hyperparameters

LLMs: Meta-in-context learning

- Some LLMs can learn in-context and also meta-learn-in-context
- Given enough examples of very similar problems, LLMs can better solve a new task

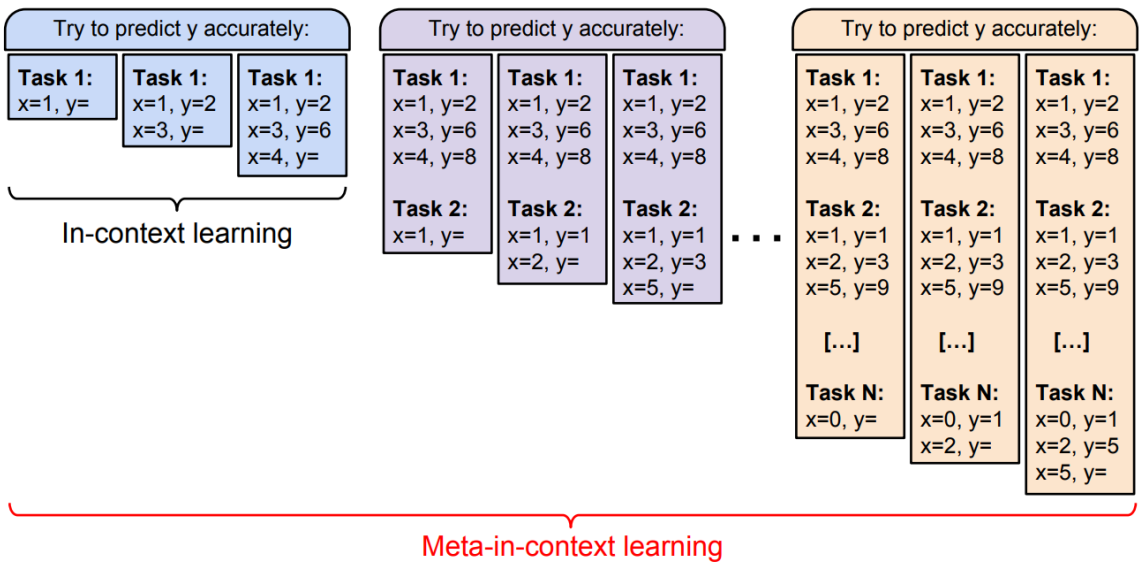


Figure: Coda-Forno, 2023

Introduction to Continual Learning

- Definition: Learning multiple tasks sequentially without catastrophic forgetting, allowing models to adapt and accumulate knowledge over time
- Importance: Enabling AI systems to learn and adapt continually in real-world settings, where tasks and data are non-stationary
- Catastrophic forgetting: The challenge of maintaining previously learned knowledge while learning new tasks; occurs when the model overwrites its weights during the learning process

online learning, lifelong learning, gradual learning, incremental learning, streaming data...

Continual Learning in the Brain

- The brain exhibits a remarkable ability to learn and retain multiple tasks without catastrophic forgetting
- Synaptic plasticity: Dynamic changes in synaptic strength facilitate learning and memory retention
- Synaptic consolidation: Stabilization of memory traces over time to protect against interference from new learning
- Neuromodulatory systems: Regulate plasticity in different brain regions based on the importance and context of new information
- Hippocampus and neocortex: Complementary learning systems with distinct roles in memory storage and consolidation
- Note: The exact neural mechanisms behind continual learning are not fully understood and are an active area of research

Positive and negative transfer

What do you want from some continual learning algorithm?

- *positive forward transfer*: previous tasks cause you to do better on future tasks
- *positive backward transfer*: new tasks cause you to do better on previous tasks

Approaches to Continual Learning

1. Regularization-based methods
2. Dynamic architectures
3. Memory-based methods
4. Bayesian methods
5. Hypertransformer

Regularization-based Methods

- Examples:
  - Elastic Weight Consolidation (EWC)
  - Synaptic Intelligence (SI)
  - Learning without Forgetting (LwF)
- Advantages: Relatively simple to implement and compatible with existing models
- Limitations: Can lead to suboptimal solutions due to the trade-off between preserving old knowledge and learning new tasks

Elastic Weight Consolidation (EWC)

- Introduced by Kirkpatrick et al. (2017)
- Regularization-based continual learning method
- Main idea: Penalize changes in important model parameters when learning new tasks
- Objective function: Add a quadratic penalty term to the standard loss function, based on the Fisher Information Matrix (FIM)
- Helps prevent catastrophic forgetting by preserving important weights while allowing other weights to be updated for new tasks

Elastic Weight Consolidation (EWC) - Technical Details

- Objective function: $L(\theta) = L_{new}(\theta) + \sum_i \frac{\lambda}{2} F_i(\theta_i - \theta_{i,old})^2$
  - $L_{new}(\theta)$: Loss function for the new task
  - λ: Regularization hyperparameter
  - F_i: Diagonal element of the Fisher Information Matrix (FIM) for parameter i
  - θ_i: Current value of parameter i
  - $\theta_{i,old}$: Value of parameter i learned from previous tasks
- Fisher Information Matrix (FIM):
  - A measure of the sensitivity of the likelihood function to changes in the model parameters
  - FIM approximates the importance of each parameter for the previously learned tasks
  - For a neural network with parameters θ, FIM can be approximated as $F = E[(\nabla_\theta \log p(y|x, \theta))^2]$
- EWC encourages updates that do not conflict with the previously learned tasks by penalizing changes in important parameters

EWC: Advantages and Limitations

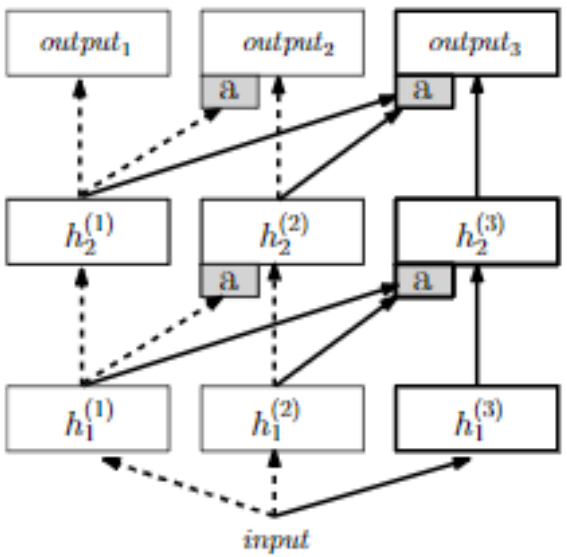
- Advantages:
  - Simple to implement and compatible with existing models
  - Effectively mitigates catastrophic forgetting
- Limitations:
  - May lead to suboptimal solutions due to the trade-off between preserving old knowledge and learning new tasks
  - Computationally expensive due to the calculation of the FIM

Dynamic Architectures

- Examples:
  - Progressive Neural Networks (PNN)
  - Dynamically Expandable Networks (DEN)
- Advantages: Can effectively handle changing tasks and prevent catastrophic forgetting
- Limitations: May result in increased model complexity and computational costs

Progressive Neural Networks (PNN)

- Introduced by Rusu et al. (2016)
- Dynamic architecture-based continual learning method
- Main idea: Expand the network architecture when learning new tasks by adding new columns
- Each column consists of several layers and is responsible for a specific task
- Connections between columns allow for knowledge transfer and prevent catastrophic forgetting



Progressive Neural Networks (PNN) - Technical Details

- When learning a new task, PNN adds a new column to the network
  - Each column consists of several layers, e.g., convolutional, pooling, fully connected
  - Columns have lateral connections to previous columns, which are fixed and not updated during training
- Lateral connections use adapter functions, such as element-wise multiplication or concatenation
- When training a new column, only the weights in the new column are updated, preventing catastrophic forgetting of previous tasks
- Inference: Each column produces a task-specific output, and the final output is the sum of all column outputs

PNN: Advantages and Limitations

- Advantages:
  - Effectively handles changing tasks and prevents catastrophic forgetting
  - Allows for task-specific capacity expansion
- Limitations:
  - May result in increased model complexity and computational costs
  - Scalability issues with a large number of tasks

Memory-based Methods

- Examples:
  - Experience Replay
  - Generative Replay
  - Coreset Selection
- Advantages: Enables the consolidation of knowledge while learning new tasks
- Limitations: May require additional memory and computational resources to store and replay experiences

Experience Replay

- Experience replay buffer: A fixed-size memory buffer that stores a subset of previously seen data
  - Can store entire samples or just the most recent model errors
  - Buffer management strategies: Random replacement, prioritized sampling, reservoir sampling, etc.
- During training, the model is updated using a mix of new data and replayed experiences
  - Minibatches can contain both new samples and samples from the buffer
  - This interleaved training helps prevent interference between tasks and improves generalization
- Experience replay can be combined with other continual learning methods, such as EWC or PNN, for improved performance

Experience Replay: Advantages and Limitations

- Advantages:
  - Enables the consolidation of knowledge while learning new tasks
  - Compatible with various learning algorithms, including reinforcement learning
- Limitations:
  - Requires additional memory and computational resources to store and replay experiences
  - Selection of the appropriate subset of data can be challenging

Bayesian Methods

- Examples:
  - Bayesian Neural Networks
  - Variational Continual Learning (VCL)
  - Uncertainty-guided Continual Learning
- Advantages: Incorporates uncertainty estimation and prior knowledge, leading to more robust learning
- Limitations: It may be computationally intensive and complex to implement

Hypertransformer

Zhmoginov (2023) and Vladymyrov (2023).

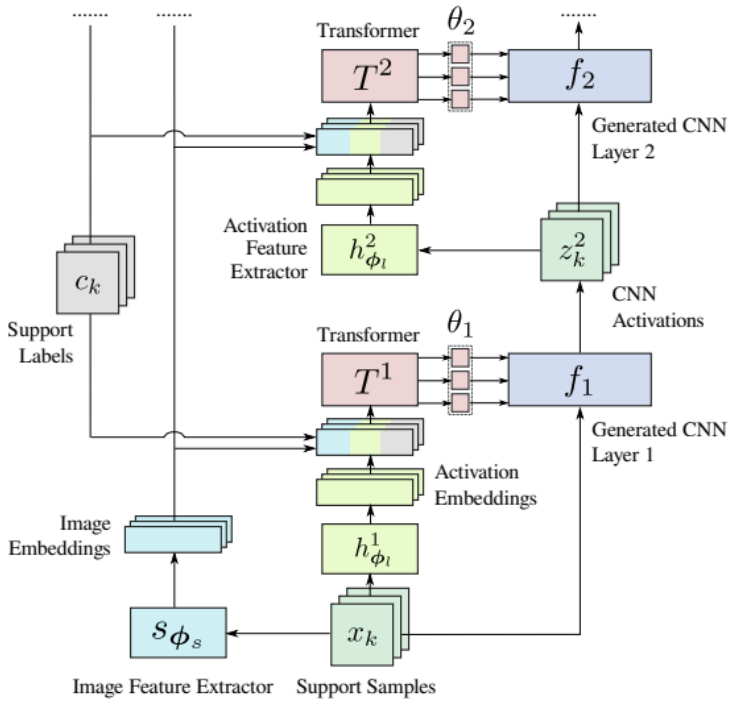


Figure: Hypertransformer

Can be used for both continual and meta-learning, we create CNN weights from scratch for each dataset. Transformer acts as a simple classifier.

Challenges in Meta-Learning and Continual Learning

- Scalability: Developing methods that can scale to large models and real-world datasets
- Data efficiency: Improving data efficiency to learn from limited or noisy data
- Task distribution and diversity: Handling diverse task distributions and learning in non-stationary environments

Future Directions

- Combining meta-learning and continual learning
- Lifelong learning models with both forward and backward positive transfer.
- Explainable and interpretable models

Real-World Applications

- Robotics: Meta-learning and continual learning can enable robots to adapt to new environments and tasks quickly and efficiently
- Healthcare: Personalized medicine, disease prediction, and treatment planning can benefit from models that adapt to new patient data and medical advances
- Natural language processing: META uses meta-learning for content filtering.
- Autonomous vehicles: Continual learning can help self-driving cars adapt to changing traffic patterns, road conditions, and driving scenarios
- LLMs: Transfer and continual learning for fine-tuning (DORA, LORA, Adapters, IA3)

Concluding Remarks

- Importance of these techniques in advancing AI research and applications: Both meta-learning and continual learning contribute to creating more versatile, adaptive, and data-efficient AI systems
- These areas of research offer numerous opportunities for innovation, and continued advancements will bring us closer to more capable and adaptable AI systems