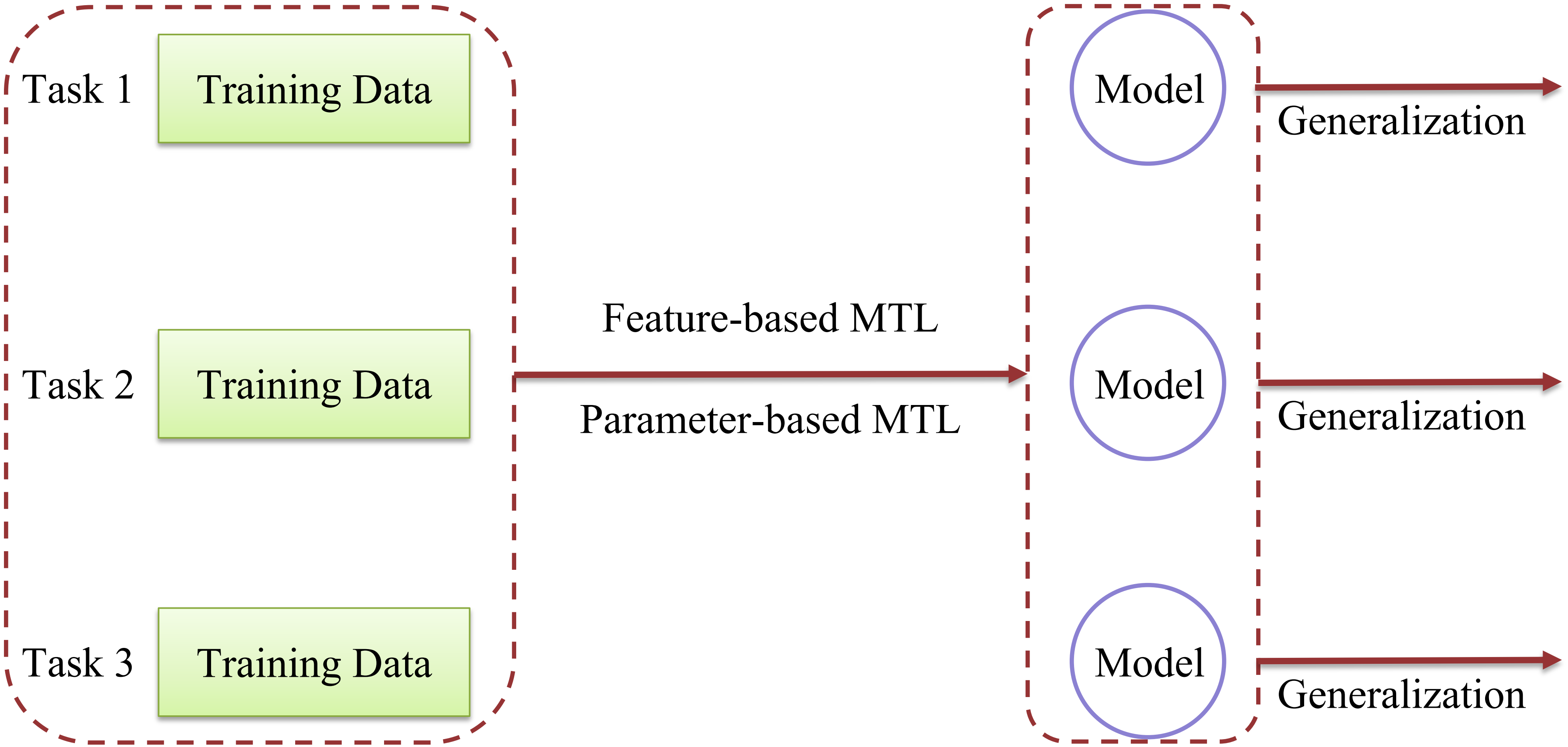
Module 10 - Project

Multi-Task Learning

AI VIET NAM
Nguyen Quoc Thai

Objectives

Multi-task Learning for Computer Vision



Outline

- Introduction
- Deep Multi-Task Architectures
- Optimization Strategy
- Experiment

Introduction

Single-Task Learning

- Image Classification

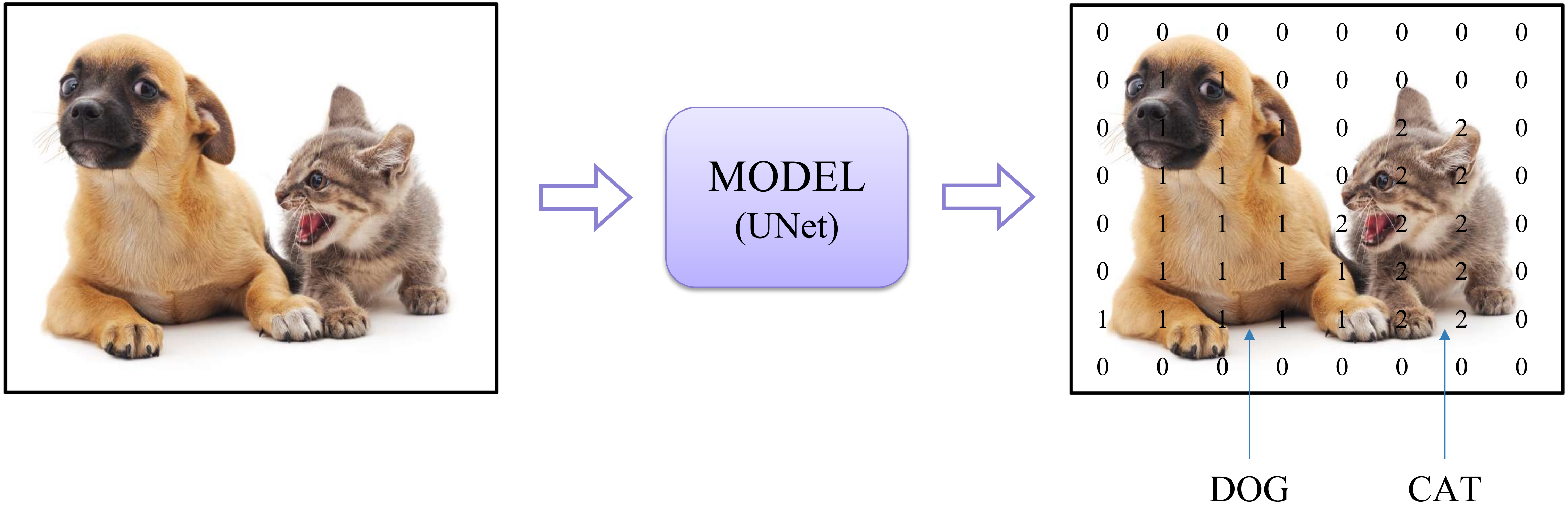
MODEL
(LeNet, ResNet,…)

Class: CAT

Introduction

Single-Task Learning

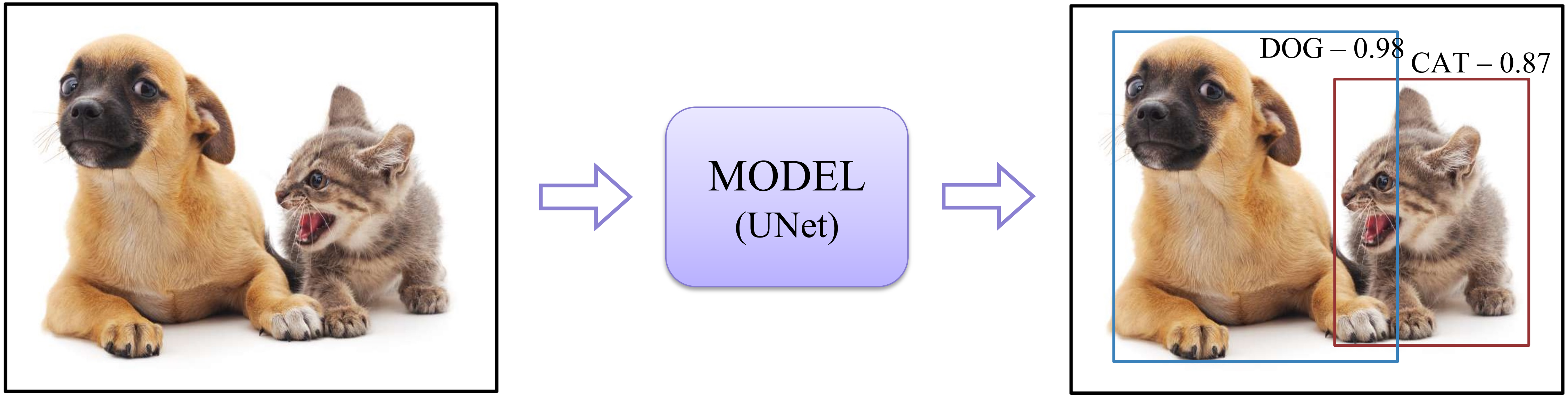
➢ Image Segmentation



Introduction

Single-Task Learning

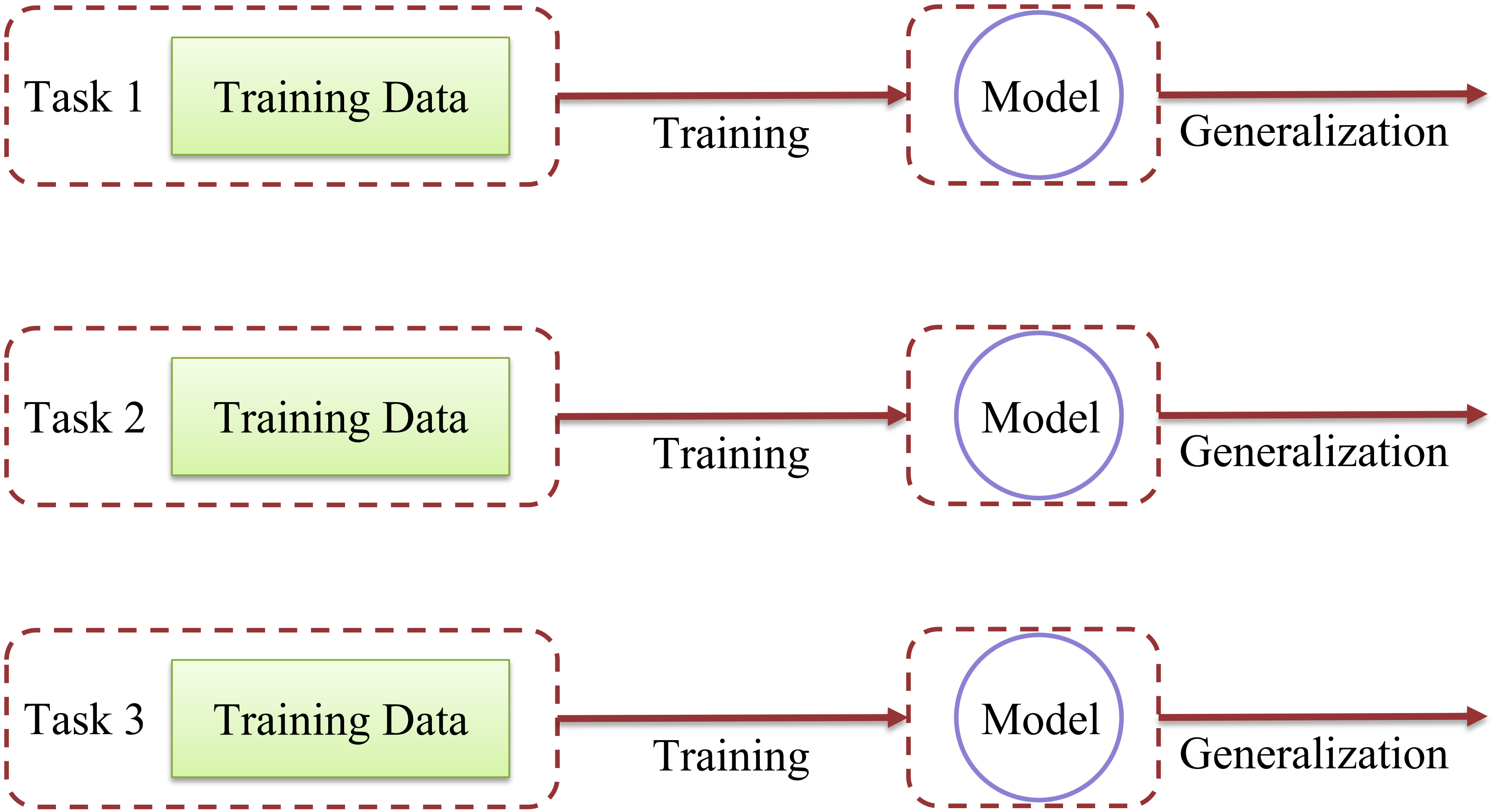
➢ Object Detection



Assign labels, bounding boxes
to objects in the image

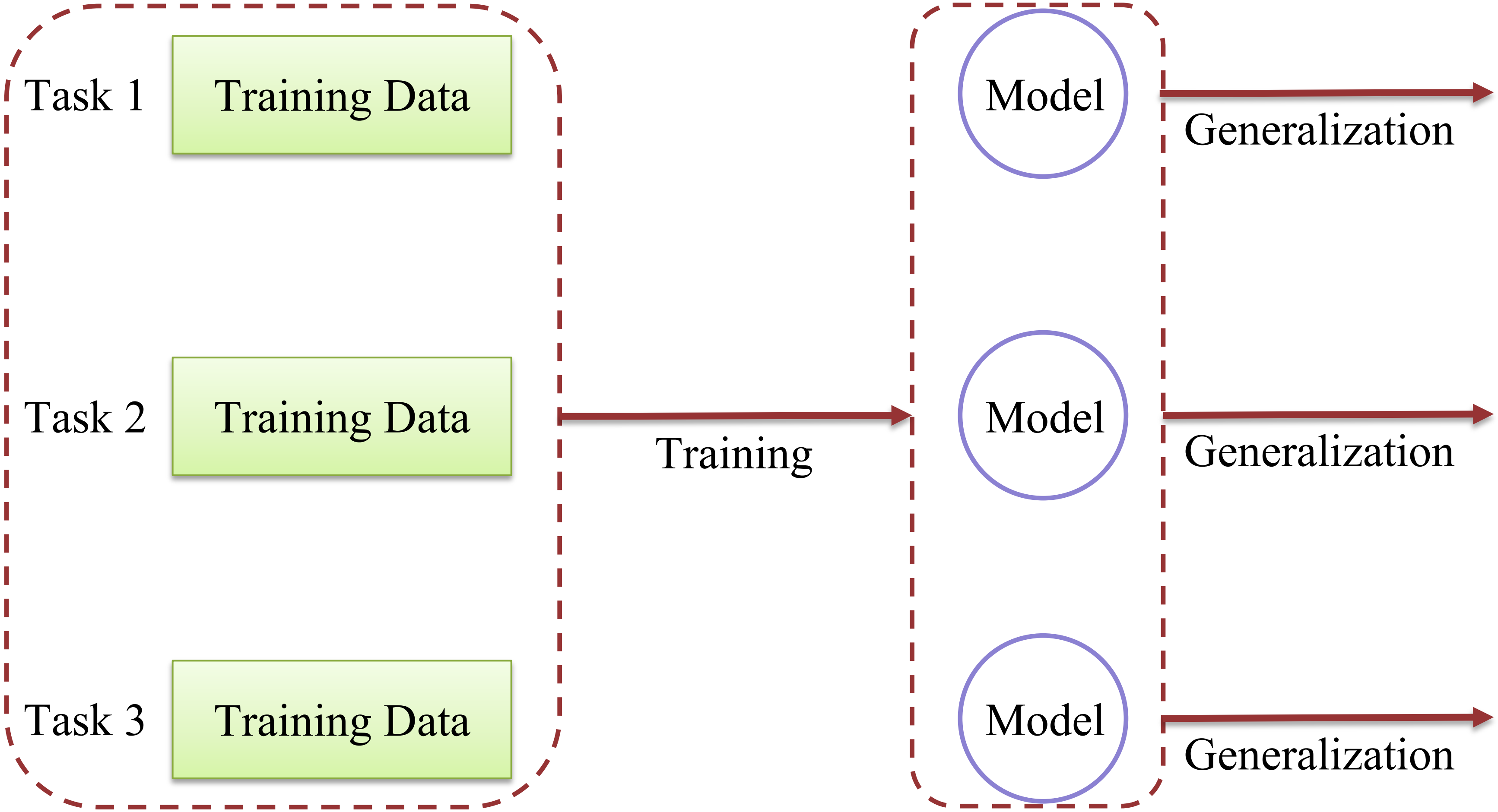
Introduction

Single-Task Learning



Introduction

Multi-Task Learning



Introduction

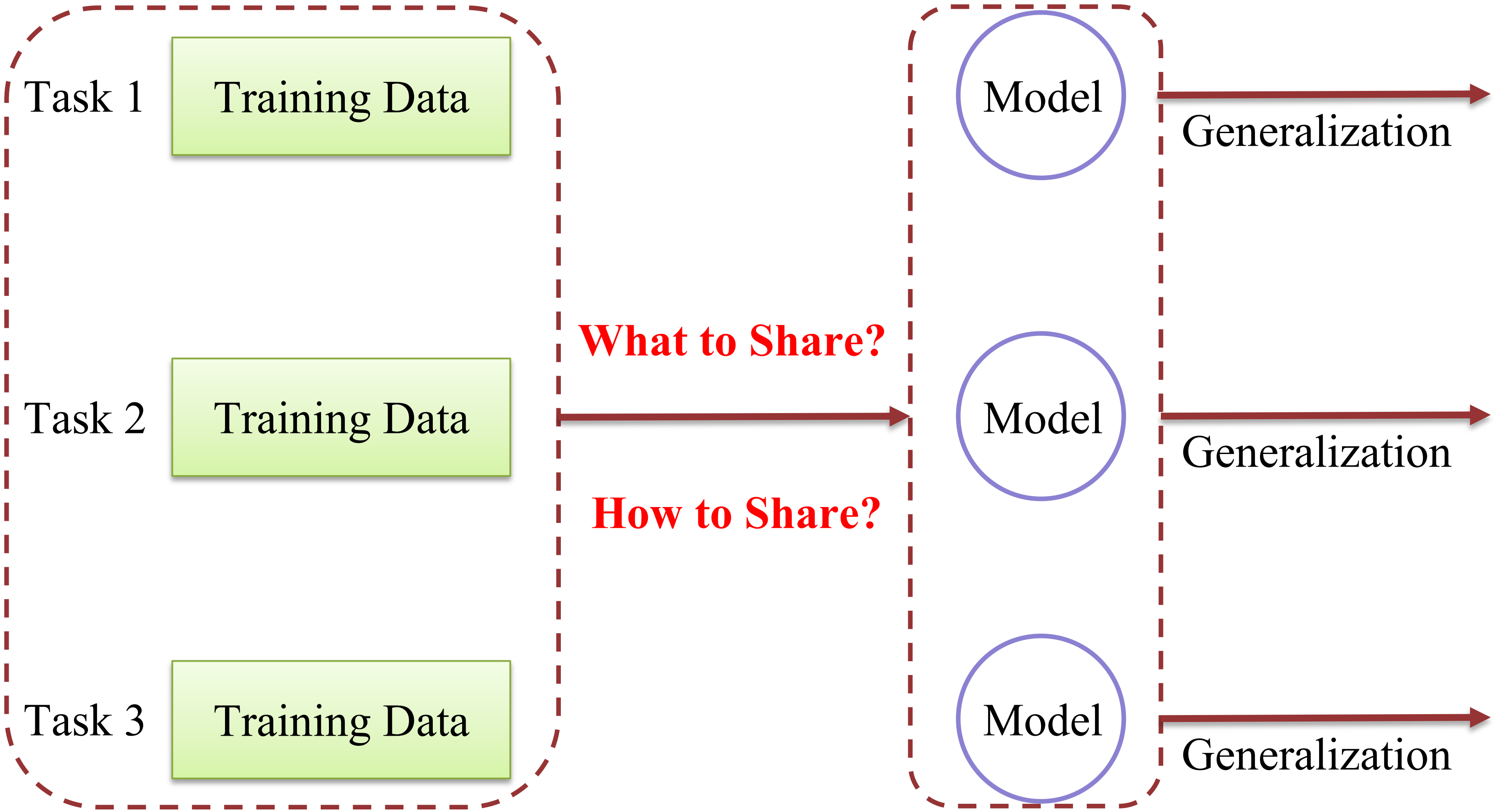
Motivation

- Learning multiple tasks jointly with the aim of mutual benefit
- Improves generalization on other tasks

  Caused by the inductive bias provided by the auxiliary task

Introduction

Multi-Task Learning



Introduction

MTL Methods (based on what to share?)

- Feature-based MTL
  - Aims to learn common features among different tasks
- Parameter-based MTL
  - Learns model parameters to help learn parameters for other tasks
- Instance-based MTL
  - Identify useful data instances in a task for other task

Introduction

MTL Methods (based on how to share?)

- Feature-based MTL
  - Feature learning approach
  - Deep learning approach
- Parameter-based MTL
  - Low-Rank approach

Introduction

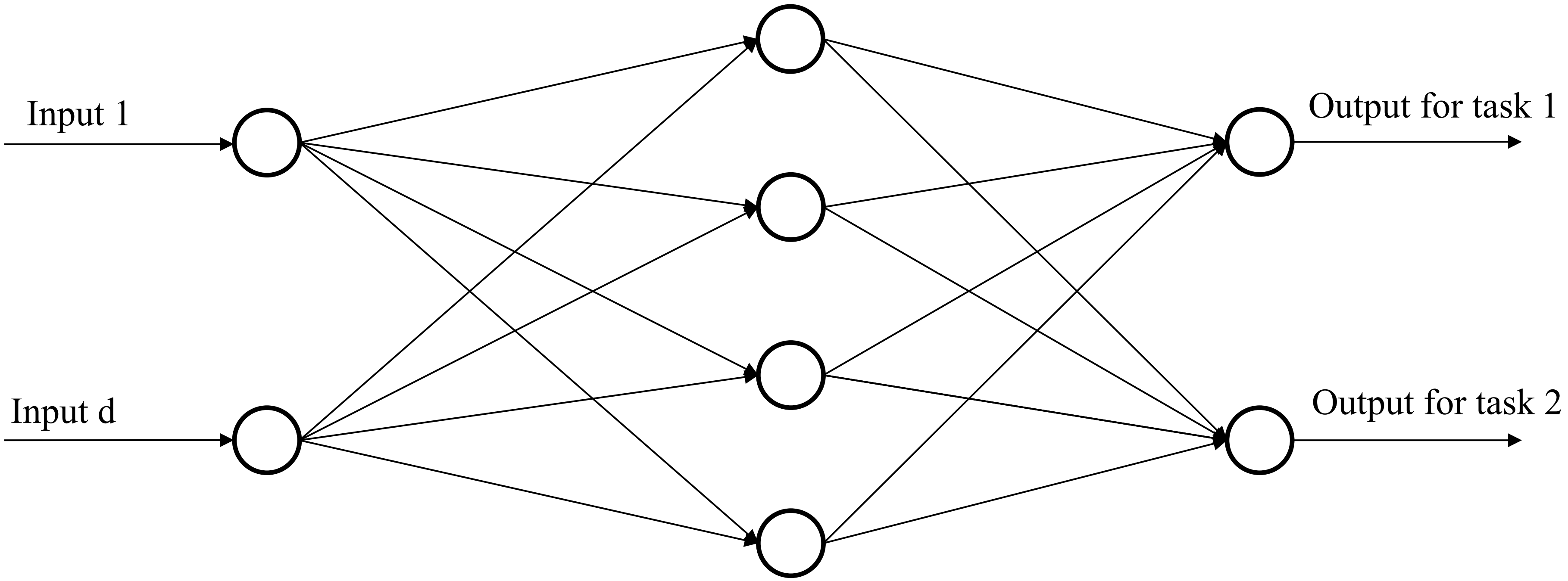
Feature Learning Approach

- Why need to learn common feature representations?
  - Original features may not have enough expressive power
- Two sub-categories
  - Feature transformation approach
  - Feature selection approach

Introduction

Feature Learning Approach

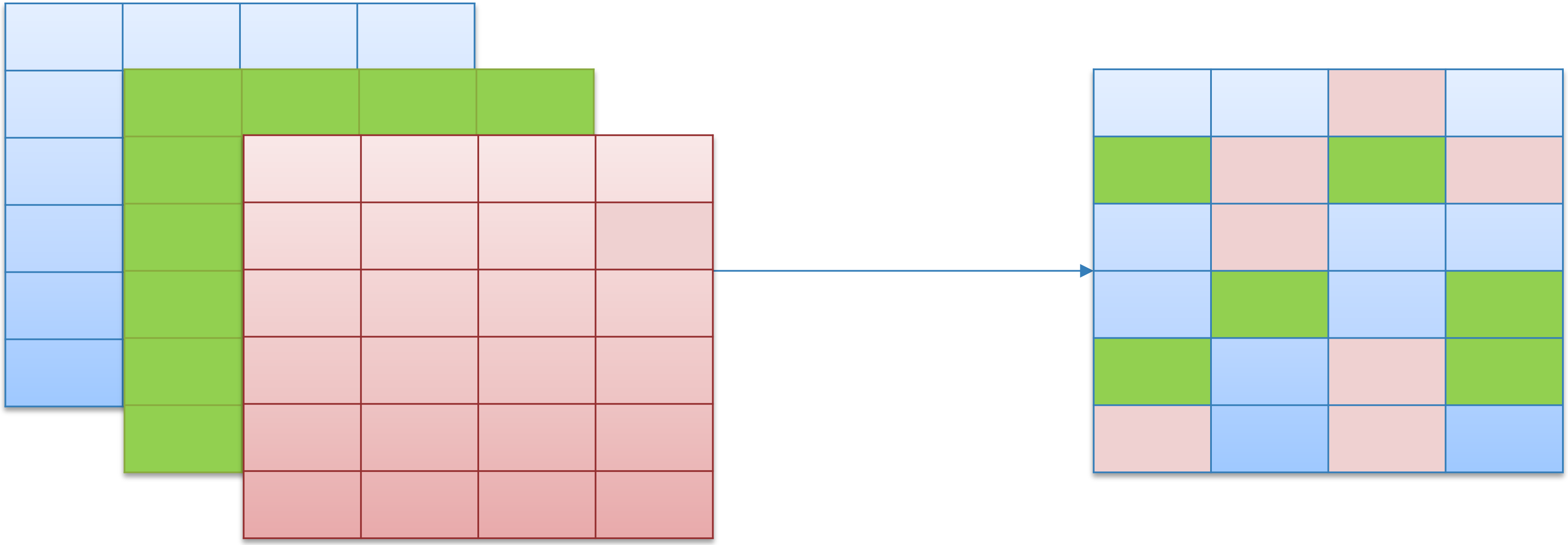
- Feature transformation approach
  - The learned features are a linear or nonlinear transformation of the original feature representation
  - Multi-task feedforward NN



Introduction

Feature Learning Approach

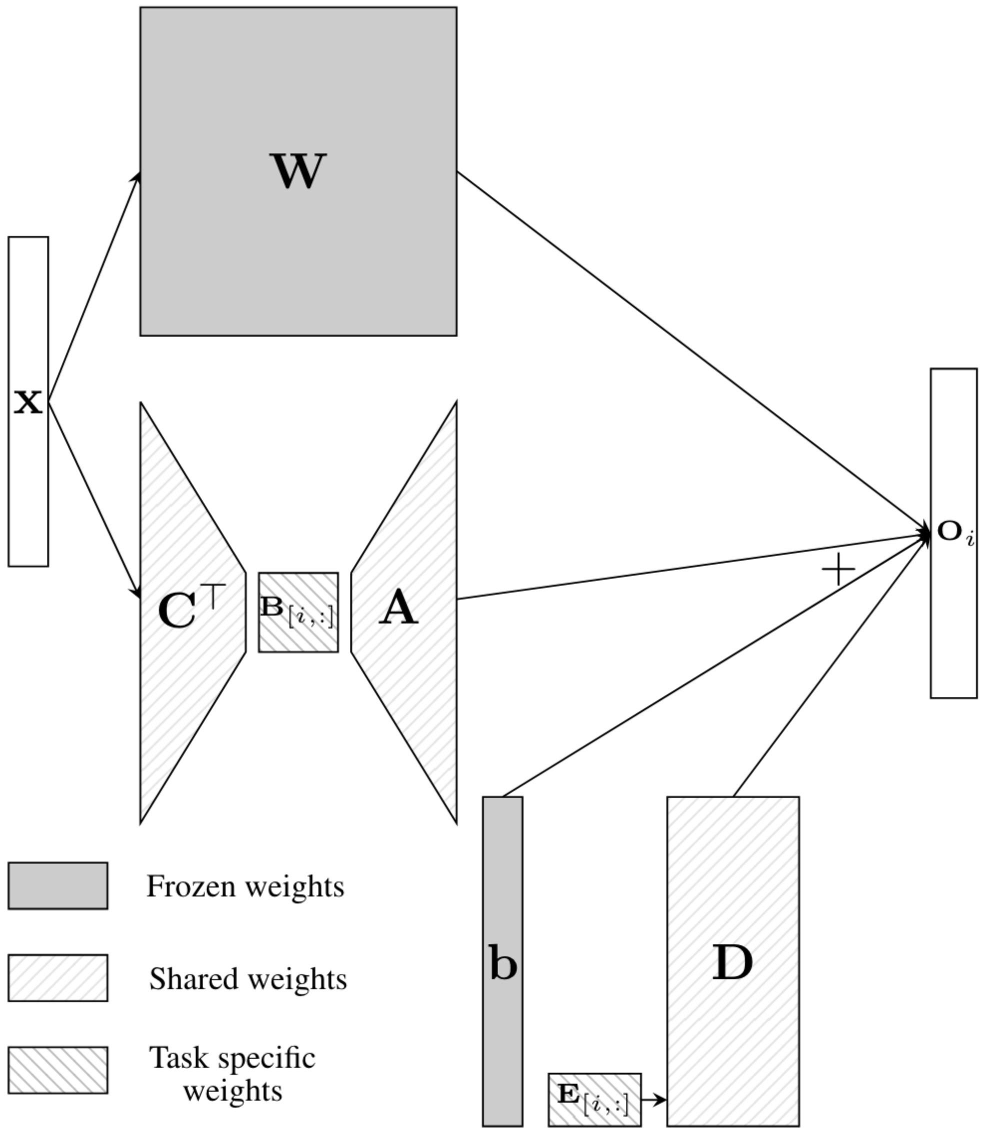
- Feature selection approach
  - Select a subset of the original features as the learned representation
  - Eliminates useless features based on different criteria



Introduction

Low-Rank Approach

- Assumes the model parameters of different tasks share a low-rank subspace



Introduction

Deep Learning Approach

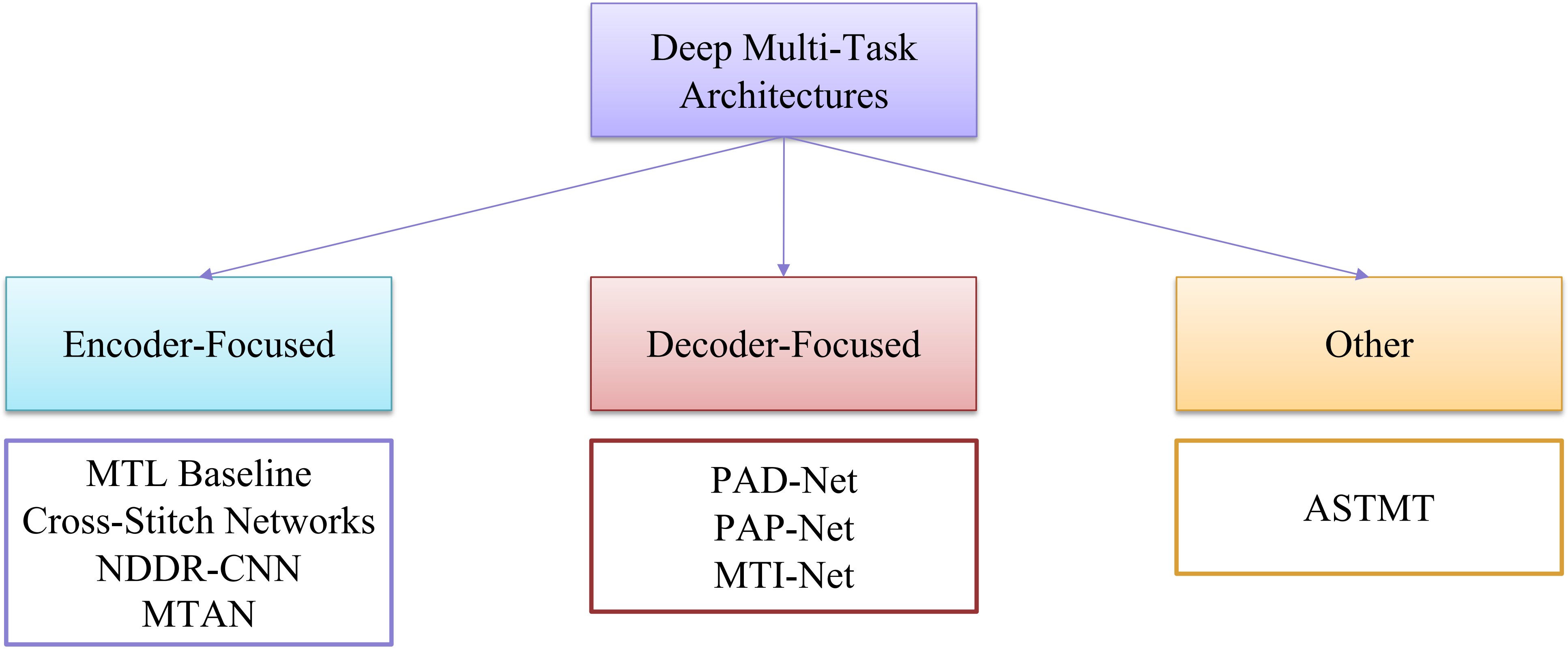
- Deep Multi-Task Architectures
  - Encoder-Focused
  - Decoder-Focused
- Optimization Strategy Methods
  - Task Balancing
  - Other: Heuristics, Gradient Sign Dropout

Outline

- Introduction
- Deep Multi-Task Architectures
- Optimization Strategy
- Experiment

Deep Multi-Task Architectures

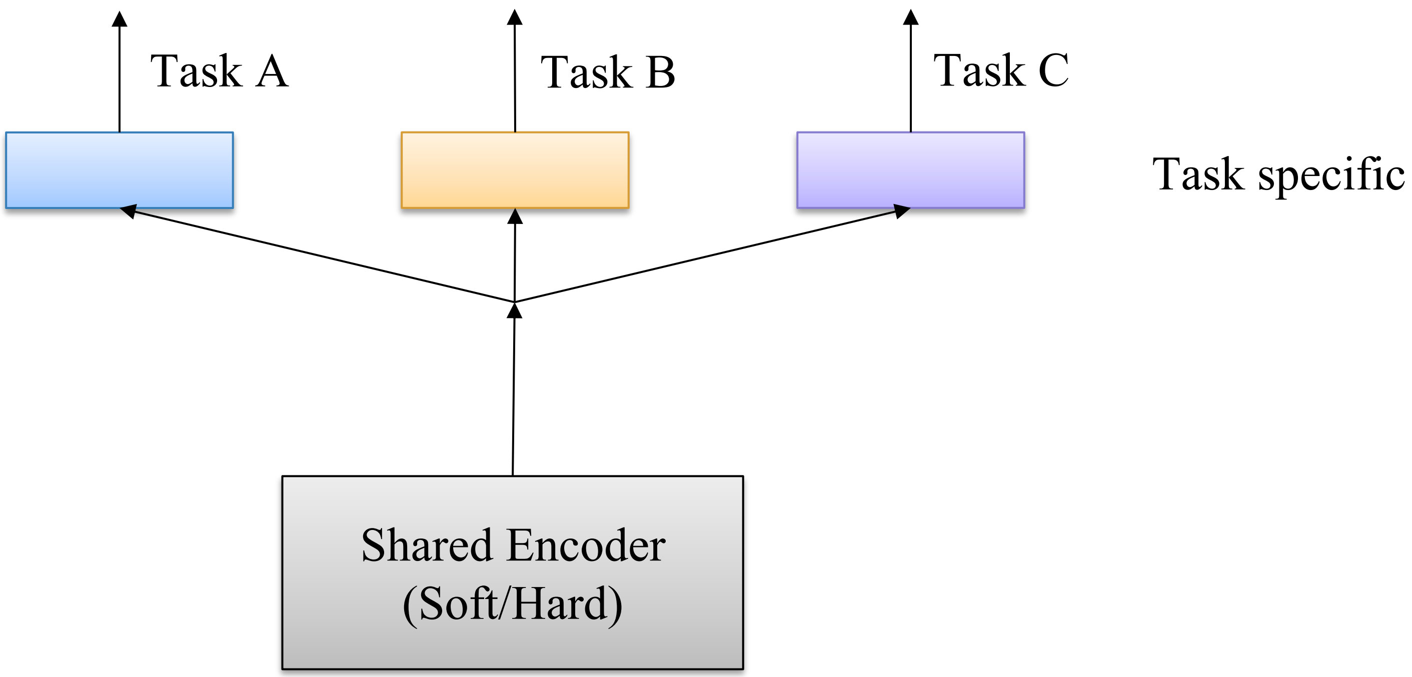
Deep Multi-Task Architectures used in Computer Vision



Deep Multi-Task Architectures

Encoder-Focused

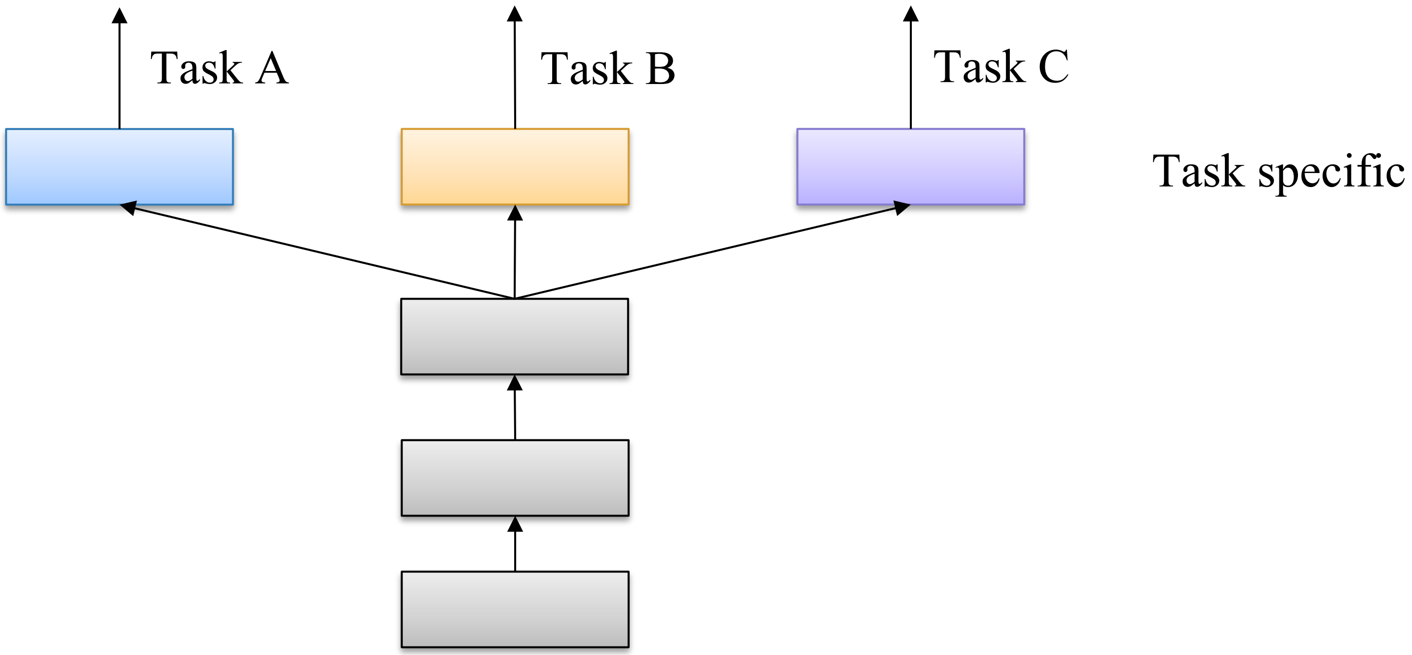
- Share the task features in the encoding stage



Deep Multi-Task Architectures

Encoder-Focused

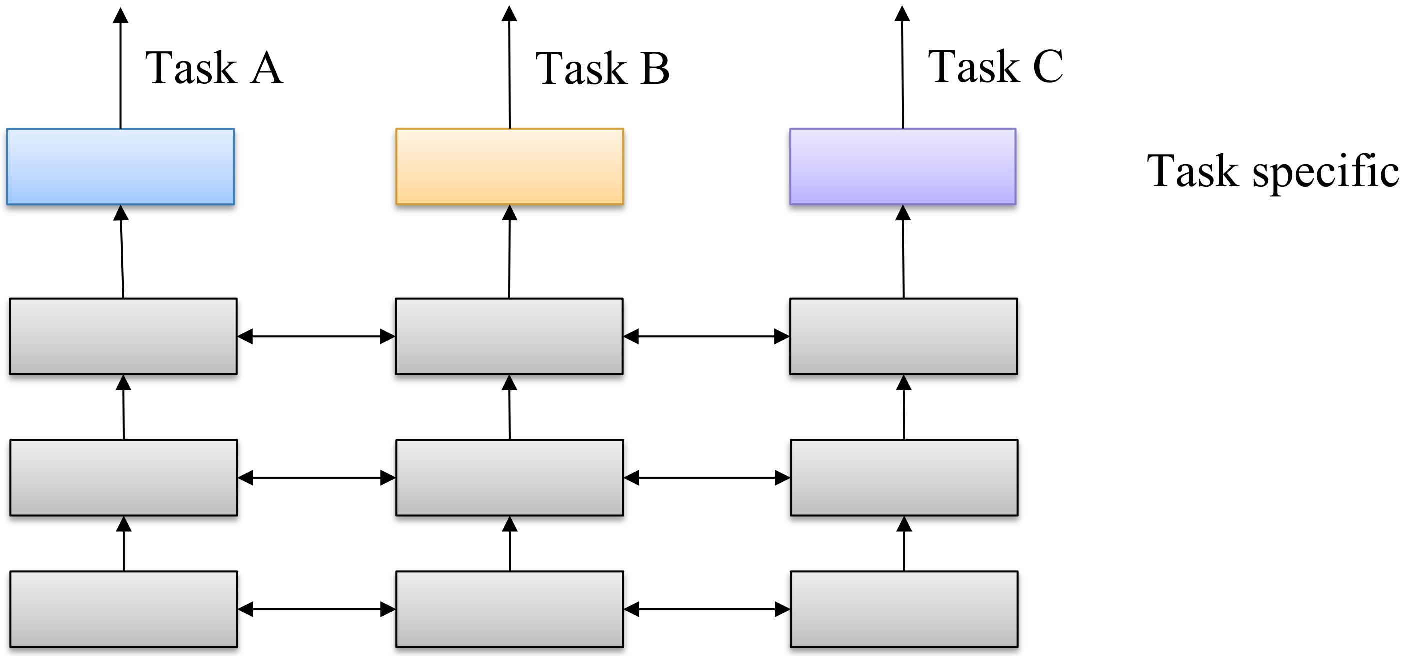
- **Hard Parameter Sharing**
  - Generally applied by sharing the hidden layers between all tasks
  - Keep several task-specific output layers



Deep Multi-Task Architectures

Encoder-Focused

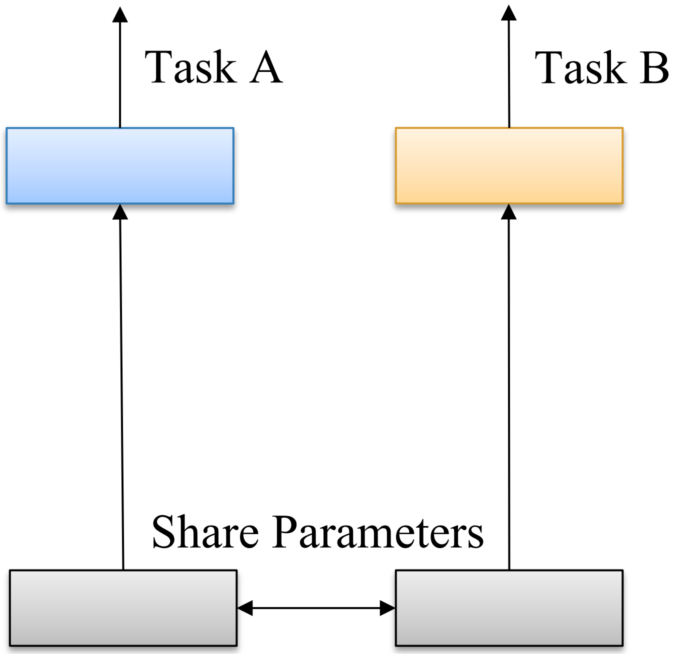
- **Soft Parameter Sharing**
  - Each task has its own model with its own parameters
  - Uses a linear combination in every layer of the task-specific networks

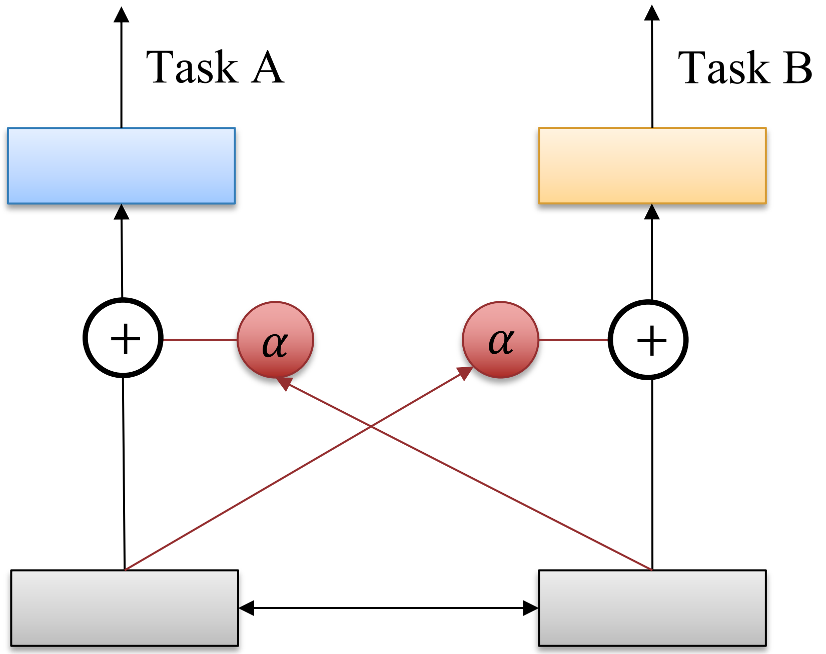


Deep Multi-Task Architectures

Encoder-Focused

- **Cross-Stitch Networks**
  - Shared the activations amongst all single-task networks in the encoder

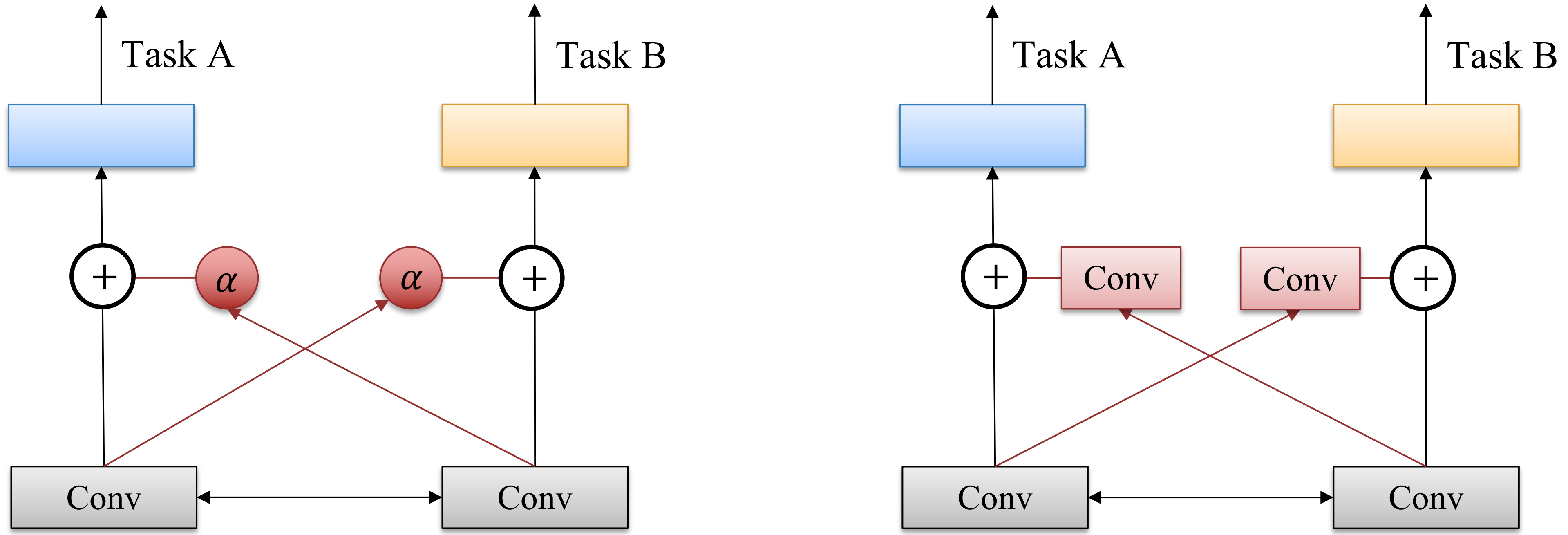




Deep Multi-Task Architectures

Encoder-Focused

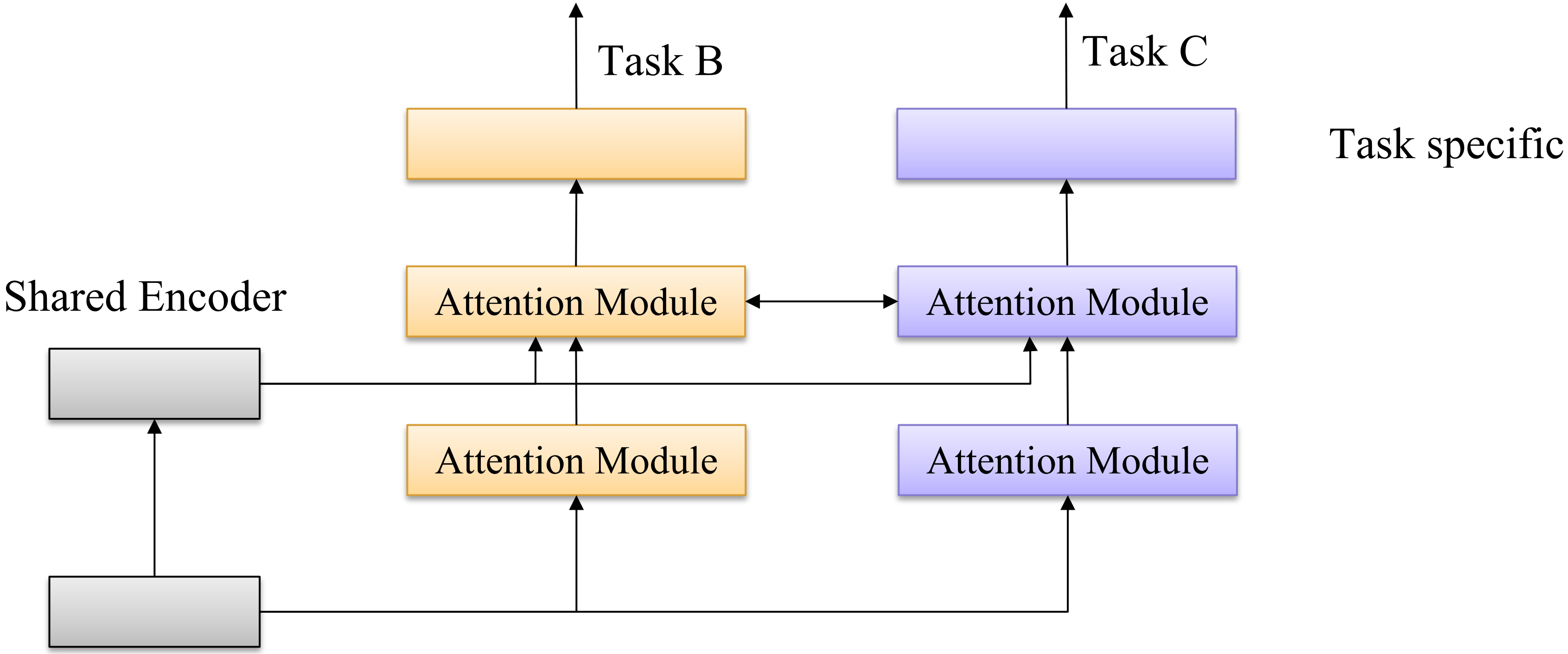
- **Cross-Stitch Networks**
  - Shared the activations amongst all single-task networks in the encoder
  - Cross connection



Deep Multi-Task Architectures

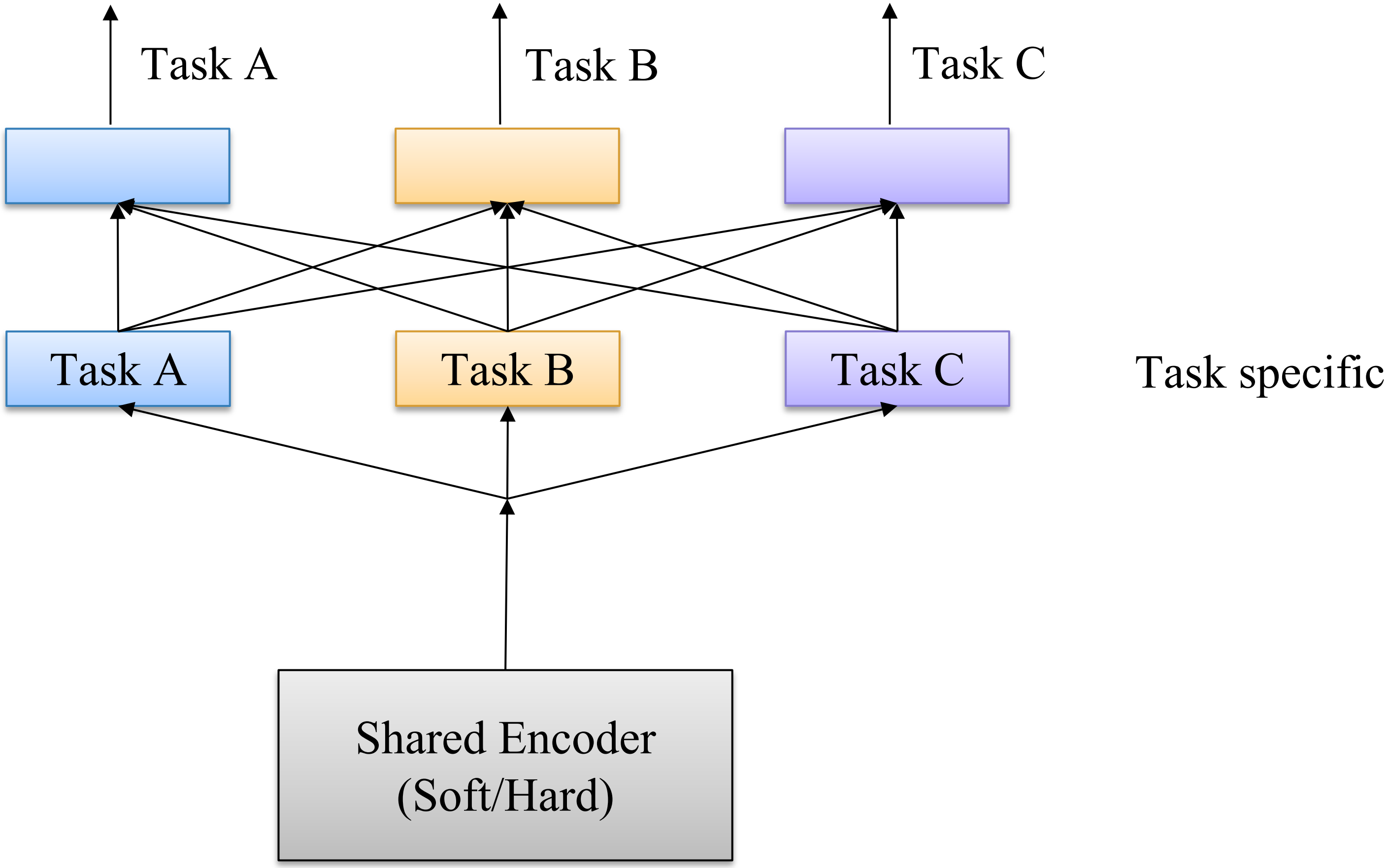
Encoder-Focused

- **Multi-Task Attention Networks**
  - Used a shared backbone network in conjunction with task-specific attention modules in the encoder



Deep Multi-Task Architectures

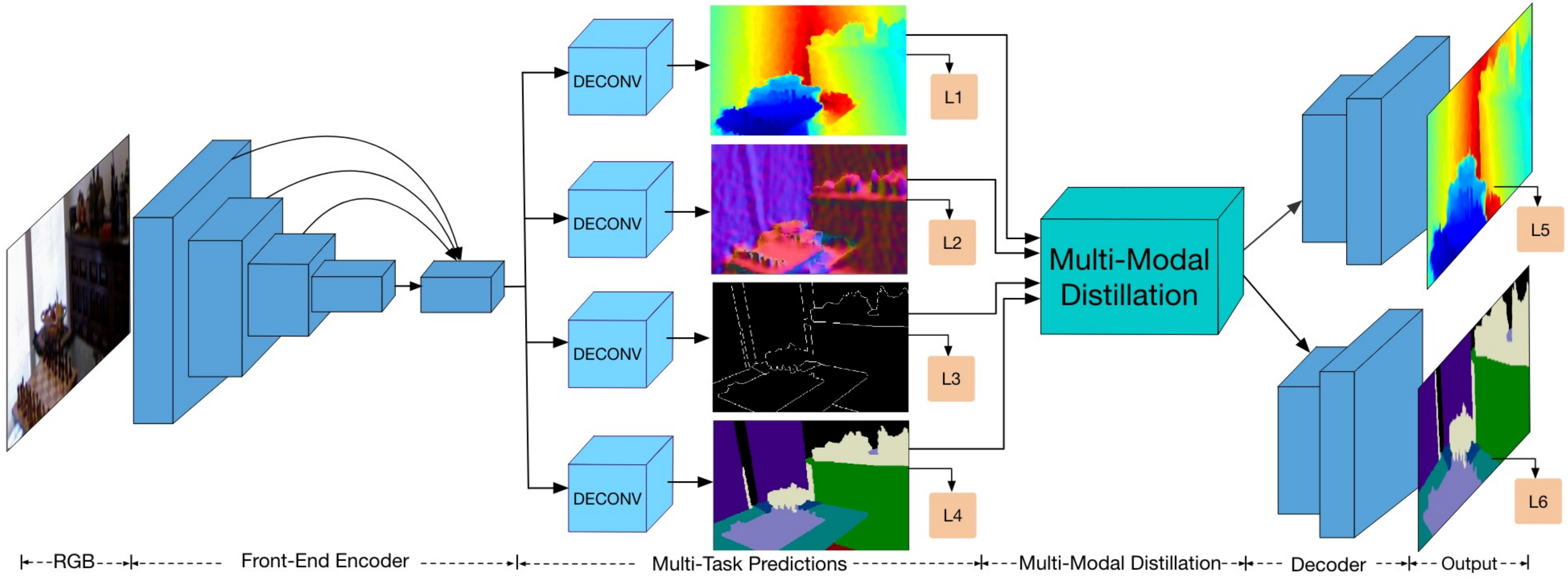
Decoder-Focused



Deep Multi-Task Architectures

Decoder-Focused

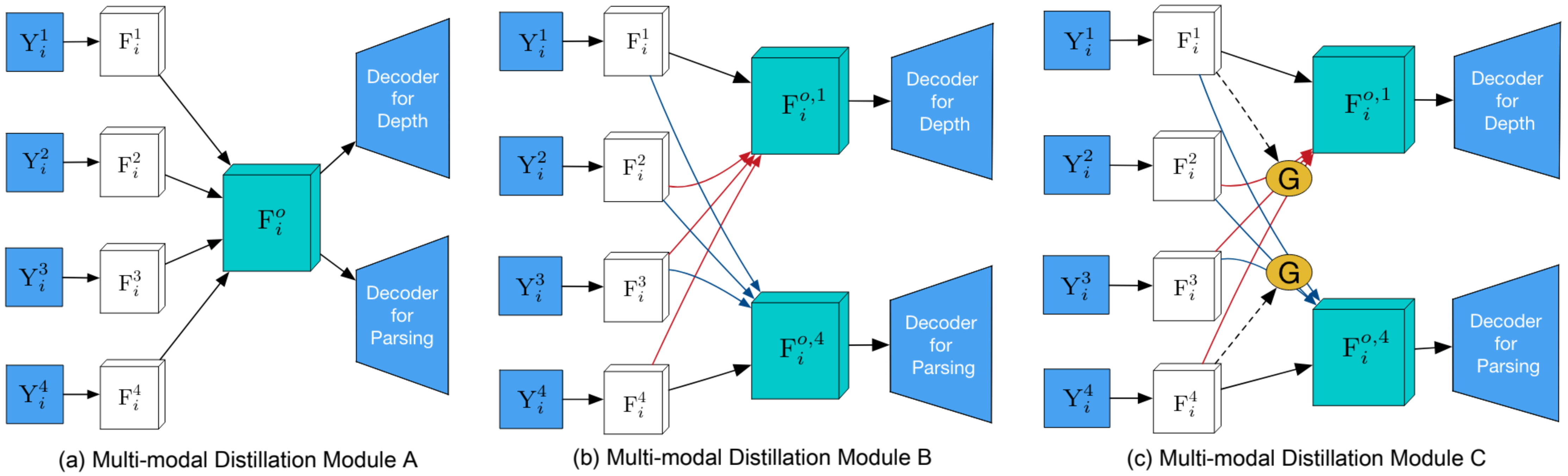
- **PAD-Net**
  - Multi-Tasks Guided Prediction-and-Distillation Network for Simultaneous Depth Estimation and Scene Parsing



Deep Multi-Task Architectures

Decoder-Focused

- **PAD-Net**
  - Deep Multimodal Distillation



Outline

- Introduction
- Deep Multi-Task Architectures
- Optimization Strategy
- Experiment

Optimization Strategy

Task Balancing Approaches

- Set a unique weight for each task

$$\mathcal{L}_{MTL} = \sum_i w_i . \mathcal{L}_i$$

- Use SGD to minimize the objective

$$W_{shared} = W_{shared} - \gamma \sum_i w_i \frac{\partial \mathcal{L}_i}{\partial W_{shared}}$$

Optimization Strategy

Uncertainty Weighting

- Use the homoscedastic uncertainty to balance the single-task losses
- Optimize the model weights W and noise parameters

$$\mathcal{L}(\mathrm{W}, \sigma_1, \sigma_2) = \frac{1}{2\sigma_1^2}\mathcal{L}_1(W) + \frac{1}{2\sigma_2^2}\mathcal{L}_2(W) + \log(\sigma_1\, \sigma_2)$$

Optimization Strategy

Dynamic Weight Averaging (DWA)

- Learns to average task weighting over time by considering the rate of change of loss for each task

Training Time

Relative Loss Change

$$w_i(t) = \frac{N \exp\left(\frac{r_i(t-1)}{T}\right)}{\sum_n \exp\left(\frac{r_n(t-1)}{T}\right)}, r_n(t-1) = \frac{L_n(t-1)}{L_n(t-2)}$$

Temperature
(Softness of Task Weighting)

Optimization Strategy

Other methods

- Gradient Normalization
- Dynamic Task Prioritization

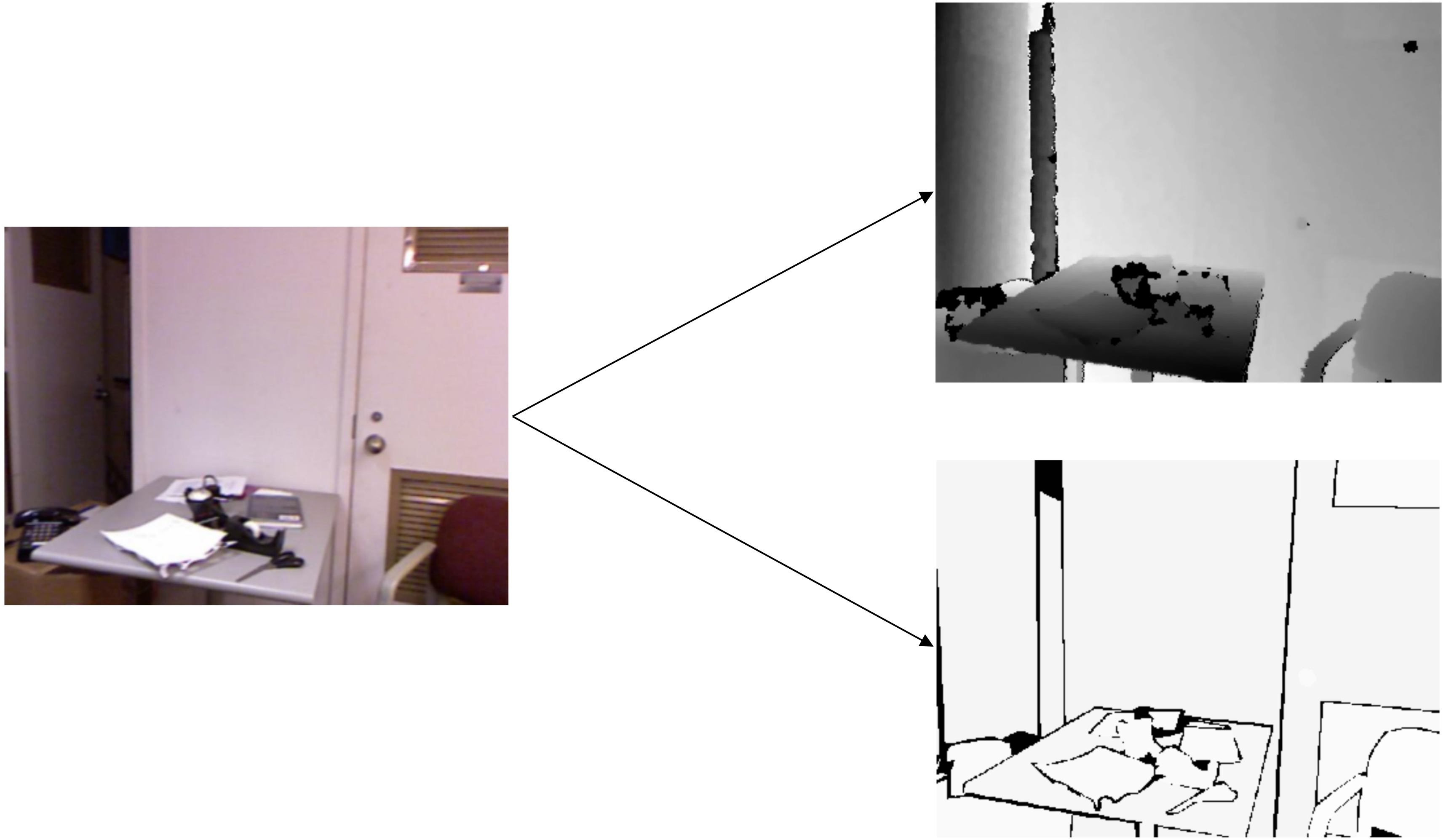
Quiz

Outline

- Introduction
- Deep Multi-Task Architectures
- Optimization Strategy
- Experiment

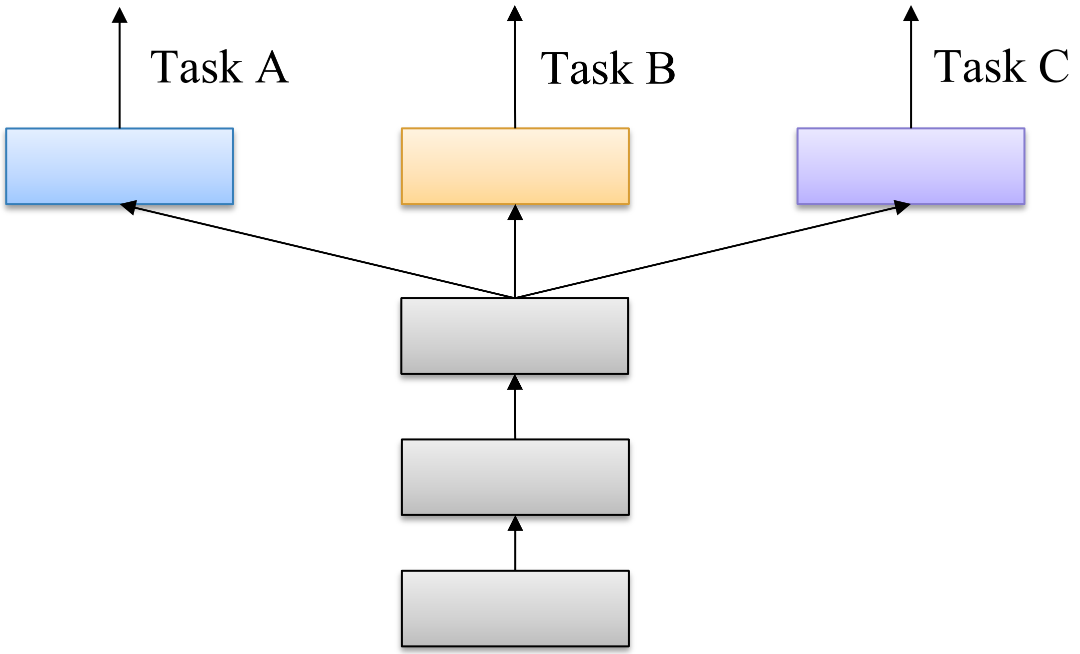
Experiment

NYUD-v2 Dataset

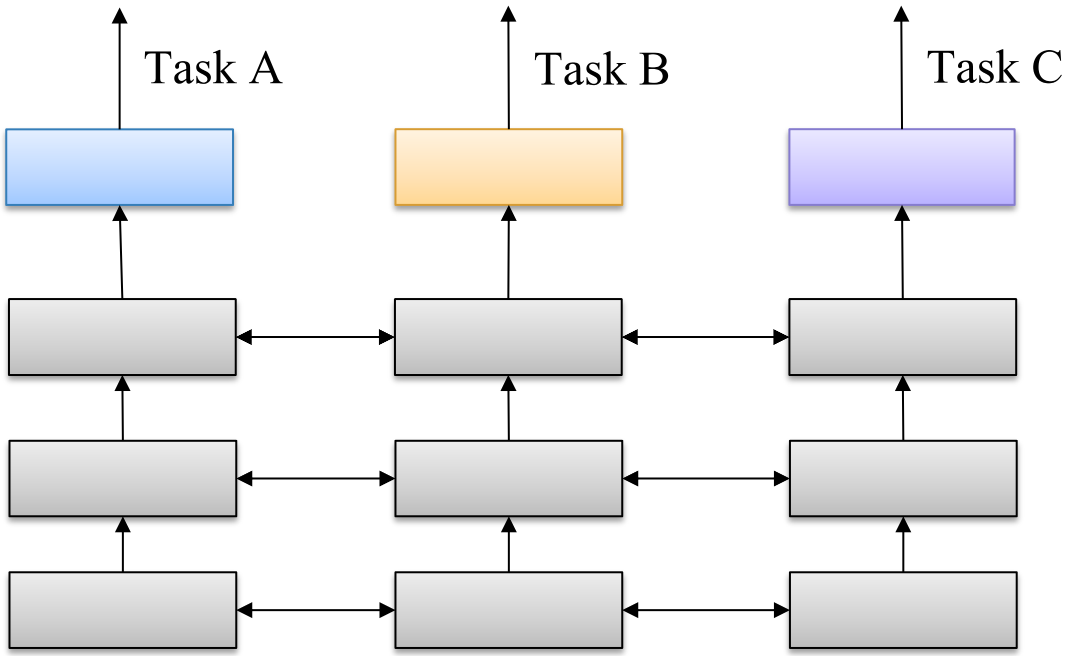


Experiment

Model



Hard Parameter Sharing

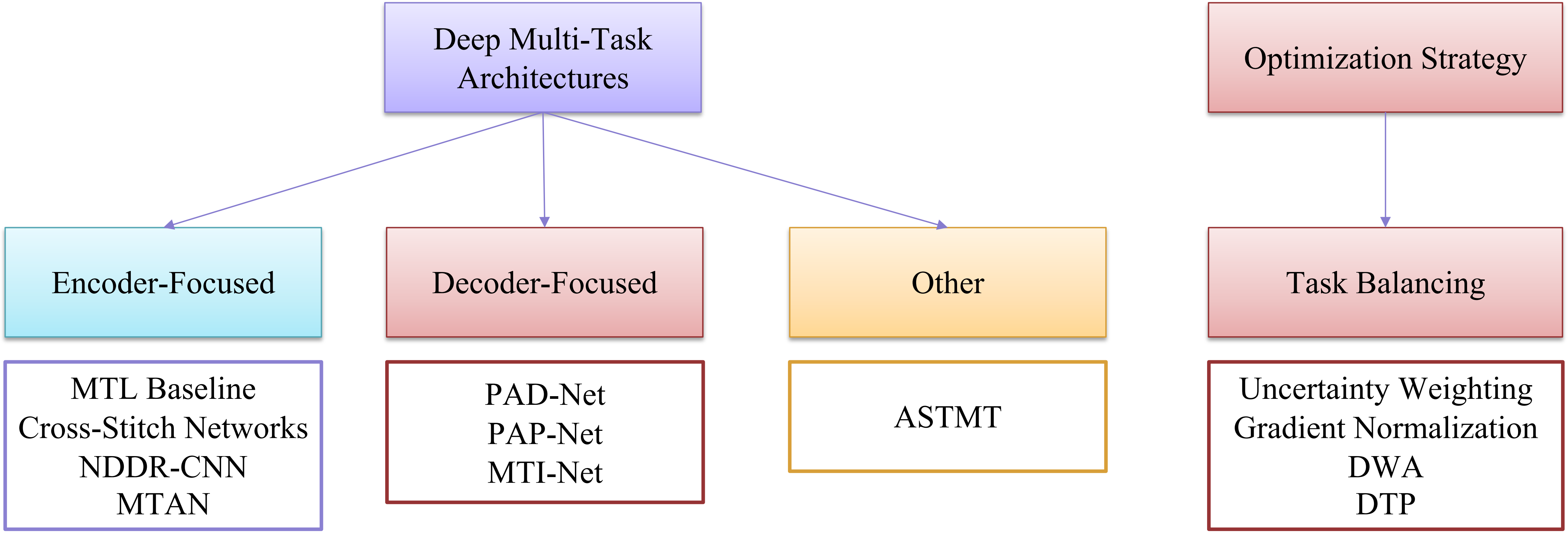


Soft Parameter Sharing

Experiment

Code

Summary


Deep Multi-Task Architectures
Optimization Strategy
Encoder-Focused
Decoder-Focused
Other
Task Balancing
MTL Baseline
Cross-Stitch Networks
NDDR-CNN
MTAN
PAD-Net
PAP-Net
MTI-Net
ASTMT
Uncertainty Weighting
Gradient Normalization
DWA
DTP

Thanks!

Any questions?