business innovation through algorithms

Transfer learning

CGnal s.r.l. – Corso Venezia 43 - Milano

23 novembre 2021| Milano

DAY 1

Introduction

- Brief overview of Machine Learning (Supervised, Unsupervised)
- Introduction to Graph, Graph Theory and main metrics for characterizing graphs

DAY 2

Graph Machine Learning

- Community detection on Graphs
- Supervised Machine Learning on Graphs

DAY 3

Explainability & Interpretability

- Introduction to explainability problem
- LIME & SHAP

DAY 4

Simple Neural Networks

- Introduction to Neural Networks, TensorFlow and Computational Graphs
- Implementation and training of simple Neural Networks

DAY 5

Advanced Neural Networks

- Convolutional Neural Networks and Recurrent Neural Networks
- Advanced Topics

Transfer learning

"*Transfer learning* and *domain adaptation* refer to the situation where what has been learned in one setting is exploited to improve generalization in another setting" (Ian Goodfellow)

Motivation

- In order to get high-performance results using neural networks, we need to train **very large and deep models**
- This requires a **lot of data** and **computing power**, both of which are often difficult and costly to obtain
  - Use some combination of *pre-training* and *transfer learning*
- Examples:
  - ✓ Natural Language Processing word embeddings: Glove, MUSE, BERT
  - ✓ Computer vision: AlexNet, Inception V3, etc.

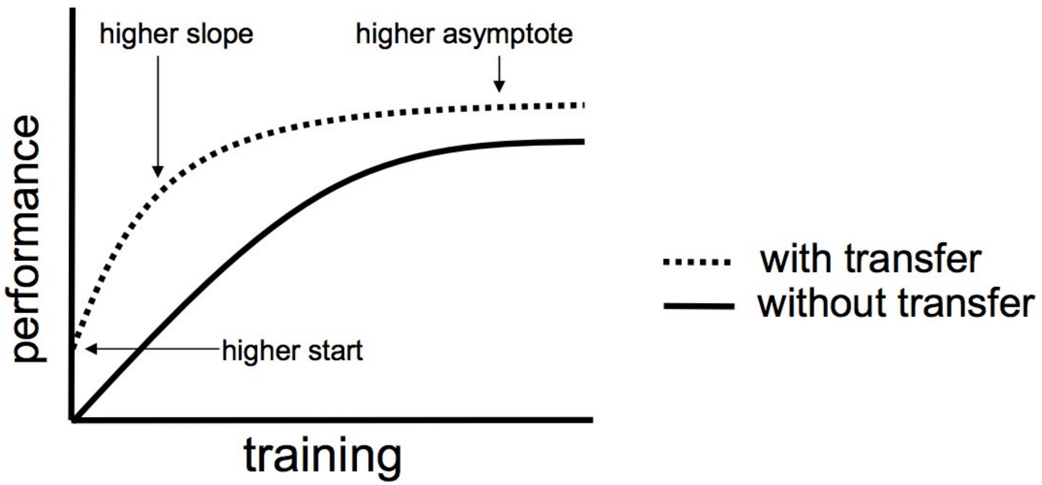
Transfer learning in deep learning

1. Train a neural network on an output Z , using the inputs X
2. Remove the learned output layer (or more than one) from this network, and attach another output layer to capture a new output Y
3. Train this new model, either using the weights from the first step as a starting point (*pre-training*) or freezing them (*transfer learning*)

When to use transfer learning

Transfer learning is an optimization, a shortcut to saving time or getting better performance.
In general, it is not obvious that there will be a benefit to using transfer learning in the domain until after the model has been developed and evaluated.

- ✓ Higher start. The initial skill (before refining the model) of the source model is higher than it otherwise would be.
- ✓ Higher slope. The rate of improvement of skill during training of the source model is steeper than it otherwise would be.
- ✓ Higher asymptote. The converged skill of the trained model is better than it otherwise would be.



Workflow for transfer learning

Workflow 1

1. Instantiate a base model and load pre-trained weights into it.
2. Freeze all layers in the base model (Keras: set `trainable = False`).
3. Create a new model on top of the output of one (or several) layers from the base model.
4. Train your new model on your new dataset.

Workflow for transfer learning

Workflow 1

1. Instantiate a base model and load pre-trained weights into it.
2. Freeze all layers in the base model (Keras: set `trainable = False`).
3. Create a new model on top of the output of one (or several) layers from the base model.
4. Train your new model on your new dataset.

Workflow 2

1. Instantiate a base model and load pre-trained weights into it.
2. Run your new dataset through it and record the output of one (or several) layers from the base model (**feature extraction**)
3. Use that output as input data for a new, smaller model.

Workflow for transfer learning

Workflow 1

1. Instantiate a base model and load pre-trained weights into it.
2. Freeze all layers in the base model (Keras: set `trainable = False`).
3. Create a new model on top of the output of one (or several) layers from the base model.
4. Train your new model on your new dataset.

Pros: can use data augmentation layers (e.g. Gaussian noise)
Cons: slower because all data must go through the pre-trained models at each epoch

Workflow 2

1. Instantiate a base model and load pre-trained weights into it.
2. Run your new dataset through it and record the output of one (or several) layers from the base model (**feature extraction**)
3. Use that output as input data for a new, smaller model.

Pros: faster, only one pass of the pre-trained model on the data
Cons: cannot perform data augmentation

Hands on

Exercise

Transfer learning with Inception V3