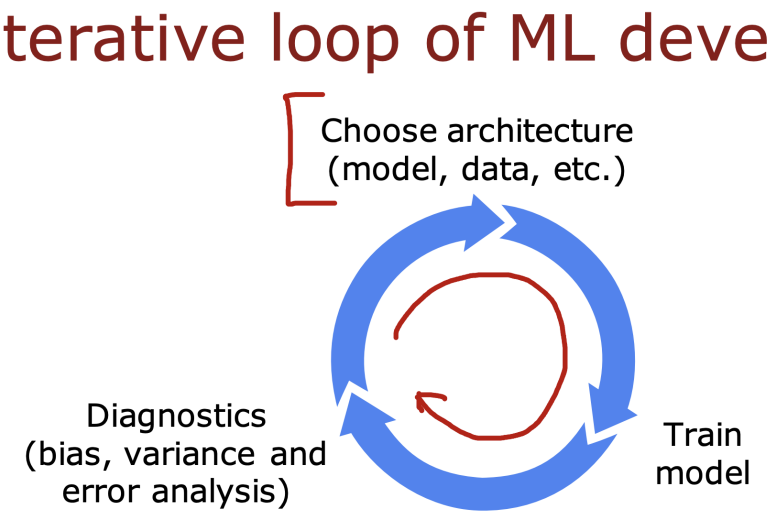
Machine learning development process



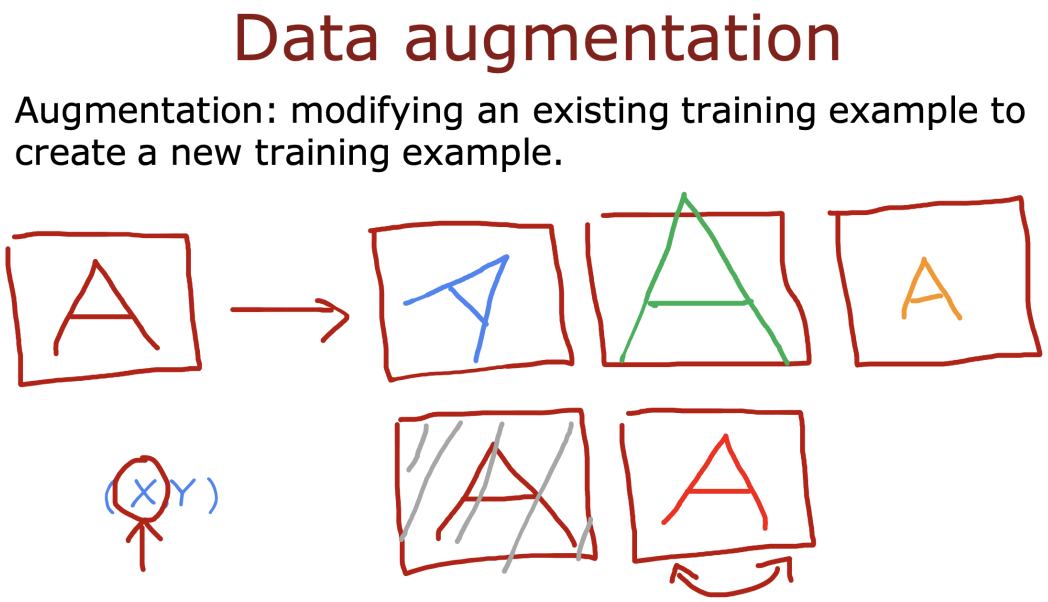
Error Analysis

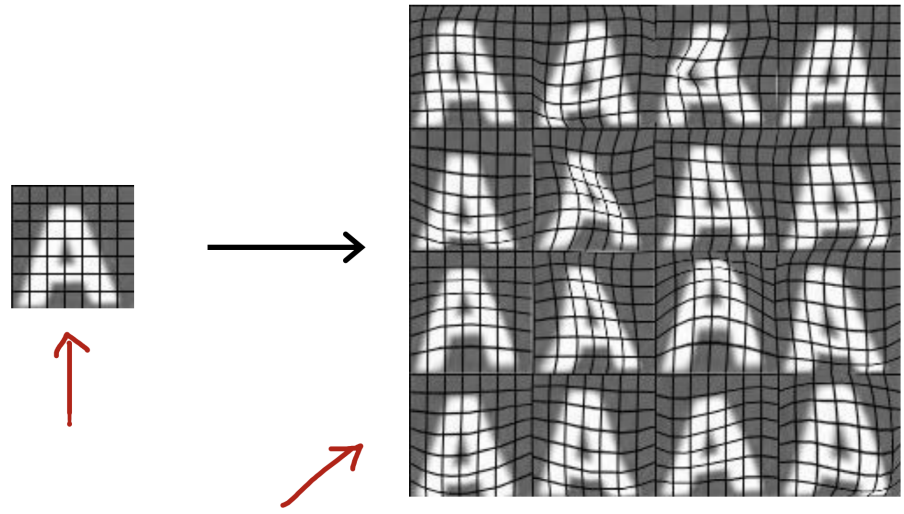
Error analysis involves manually examining examples that the algorithm has misclassified to gain insights into where the algorithm is going wrong. The speaker suggests grouping these examples into common themes or traits. For example, if a lot of misclassified spam emails are pharmaceutical sales, count how many emails fall into this category. This process can help prioritise what to work on next.

Adding Data

1. **Targeted Data Collection**: Instead of collecting more data of all types, which can be slow and expensive, focus on adding more data of the types where analysis has indicated it might help. For example, if error analysis reveals that a machine learning algorithm is struggling with a certain type of spam email, focus on collecting more examples of that type of spam.

2. **Data Augmentation**: This technique involves modifying existing data to create new data. For example, if you're working on an optical character recognition problem, you could create new training examples by rotating, enlarging, shrinking, or changing the contrast of images of letters. This technique is especially useful for image and audio data.





Data augmentation for speech

Speech recognition example

Original audio (voice search: ”What is today’s weather?”)

\+ Noisy background: Crowd

\+ Noisy background: Car

\+ Audio on bad cellphone connection

1. **Data Synthesis**: Using artificial data inputs to create a new training examples. For example, in a photo optical character recognition task, you could generate synthetic data by typing random text in different fonts and colors in a text editor and taking screenshots.

Artificial data synthesis for photo OCR

Real data

Synthetic data

[Adam Coates and Tao Wang]

Transfer Learning

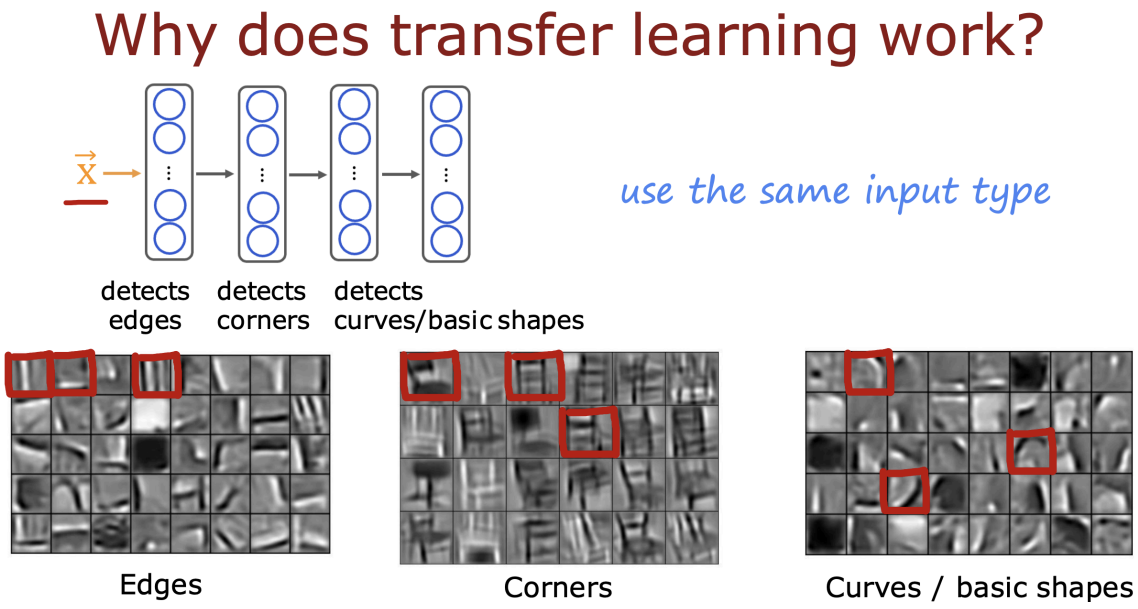
Transfer learning is a technique where a pre-trained model, trained on a large dataset, is used as a starting point for a related task.

The process involves two main steps: supervised pre-training and fine-tuning.

In supervised pre-training, a neural network is trained on a large dataset, often of a different but related task. The learned parameters from this step are then used as a starting point for the second step, fine-tuning.

In fine-tuning, these parameters are further optimised on the specific task at hand.

There are two options for fine-tuning: one can either train only the output layer's parameters, keeping the rest fixed, or train all the parameters in the network. The choice between these two options depends on the size of the training set for the specific task.



The intuition behind transfer learning is that by learning to recognize a variety of objects, the neural network learns to detect generic features of images, such as edges, corners, and basic shapes, which are useful for many other computer vision tasks.

One restriction of transfer learning is that the input type has to be the same for both the pre-training and fine-tuning steps. For instance, if the final task is a computer vision task, then the pre-training step also needs to be a neural network trained on images.

Transfer learning summary

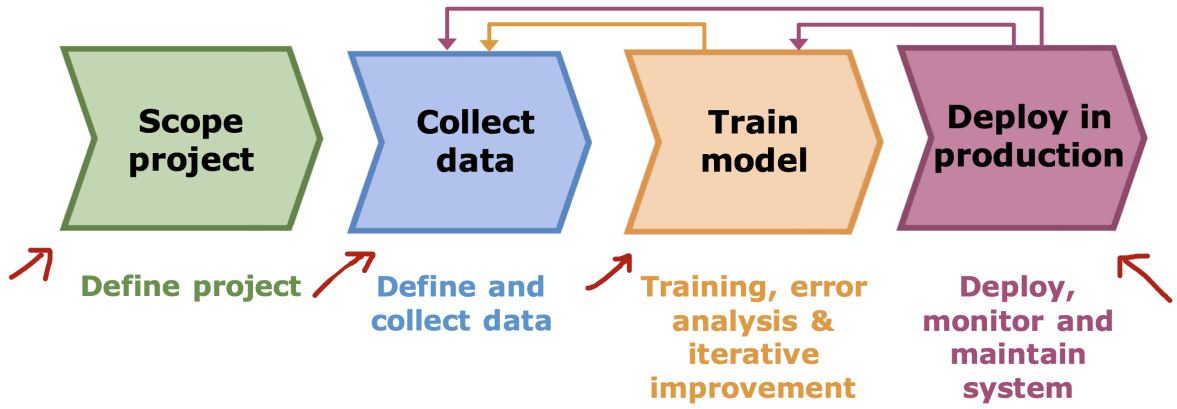
1. Download neural network parameters pretrained on a large dataset with same input type (e.g., images, audio, text) as your application (or train your own).

1M

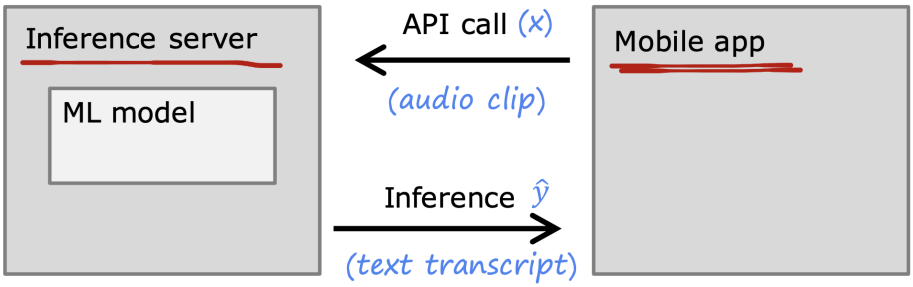
2. Further train (fine tune) the network on your own data.

1000
50

Full cycle of a machine learning project



Deployment



→ Software engineering may be needed for:
- Ensure reliable and efficient predictions
- Scaling
- Logging
- System monitoring
- Model updates

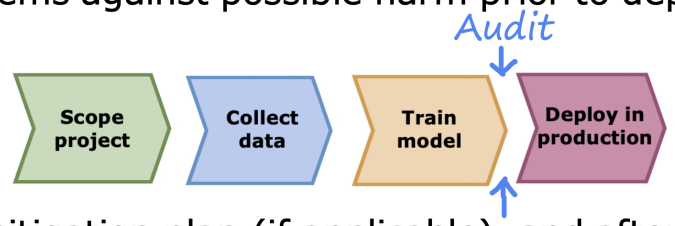
MLOps
machine learning operations

Guidelines

Get a diverse team to brainstorm things that might go wrong, with emphasis on possible harm to vulnerable groups.

Carry out literature search on standards/guidelines for your industry.

Audit systems against possible harm prior to deployment.



Develop mitigation plan (if applicable), and after deployment, monitor for possible harm.