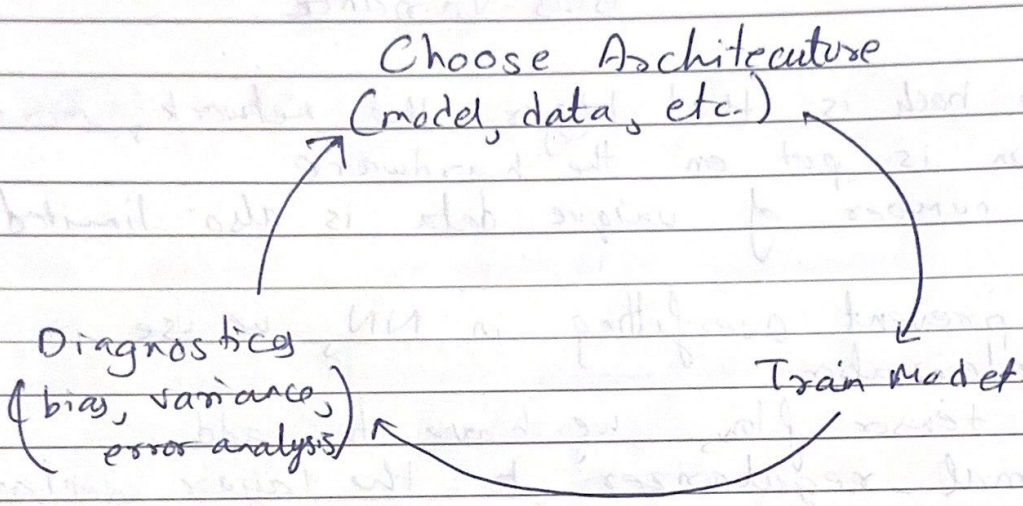
Iterative Loop of ML Development



Error Analysis

m_{cv} = 500 examples in cross validation set

Algorithm misclassifies 100 of them

Manually examine 100 examples and categorize the based on common traits

Ex. if we have a spam mail catcher and get following errors

Pharma: 21

Deliberate mispelling: 3

Unusual routing: 7

Steal Passwords (phishing): 18

Spam message embedded in image: 5

Note: Some errors might be overlapping

here 2 type of mails are giving most issues. So it is better to work on them than the rest

Once we find an common error, we can make changes like → get more data, adding more features, etc. for those specific errors.

Note. In real projects, the data size are huge, so we can take random 100 ~~rate~~ errors to do analysis.

Error Analysis is better for thing humans are good at.

Adding Data

When working, it is tempting to get more data on everything. However, it is important to get data of types where error analysis has indicated it might be of help

Eg. going to unlabeled data & finding more examples of Pharma related spam.

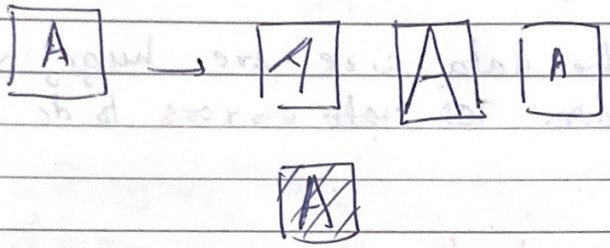
Other than getting brand new training examples, we can use 'Data Augmentation'

Note: Only augment data in ways which might be present in the test set.
It is not helpful to add purely random noise to our data.

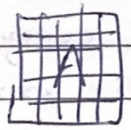
Data.

Augmentation: modifying an existing training example to create a new training example

Eg. image of 'A' detection

(1) 

(x, y)
↓
modifying x to another type having same label y

(2)  using a grid system and warping it at multiple areas

Eg. Speed recognition example

1 → original audio
2 + Crowd noise
3 + Car noise
4 + bad cellphone connection

] original audio
] noise

We can combine the original audio with raw each of the background noise to create more training data

Data

Synthesis : using artificial data inputs to create a new training example.

Eg. OCR - optical character recognition

When seeing an image, algorithm will detect characters.

For training, we have images of billboards that the computer uses to detect/learn how to detect characters from real world

Instead of getting more data, we can use a text editor and spam letters in different fonts to get more data.

Note: Algorithms to produce realistic synthetic data can take large amount of time to develop but help a lot when done.

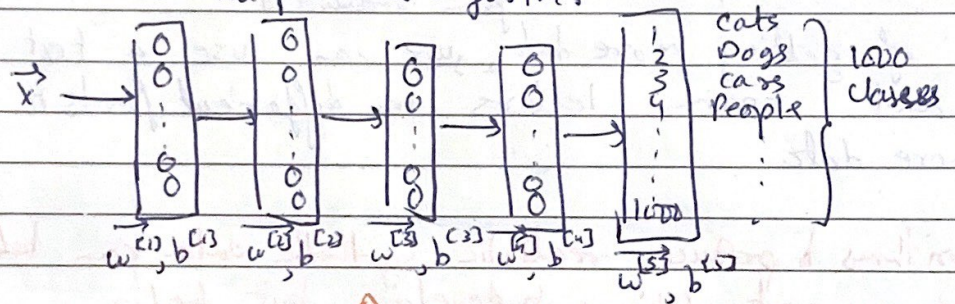
Note: Most notably seen in computer vision ~~tasks~~. development.

Transfer Learning

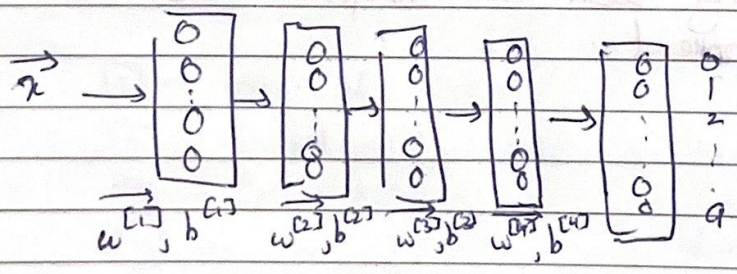
1. Download NN parameters pretrained on a large dataset with same input typ (eg. images, audio, text) as your application (or train your own)

2. Further train (fine tune) the network on your own data.
   - train only output layer (for small data set)
   - train all parameters further

Ex



supervised pretraining



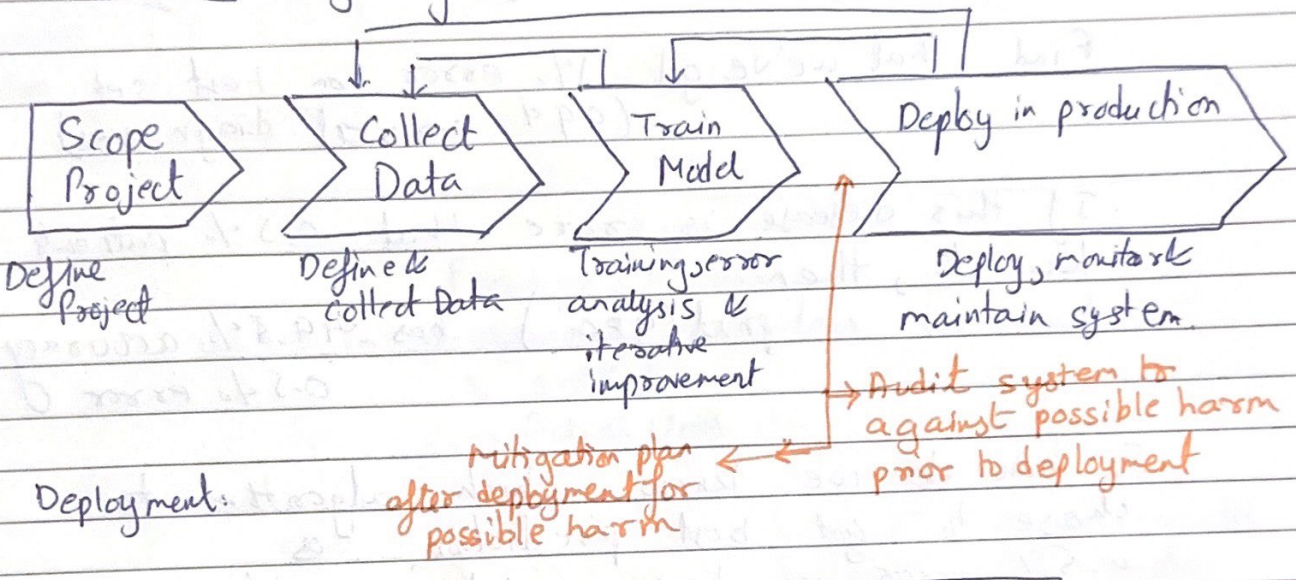
fine tuning

Using a pretrained NN helps as several thing like edge detection / corners and other features are common amongst all NN

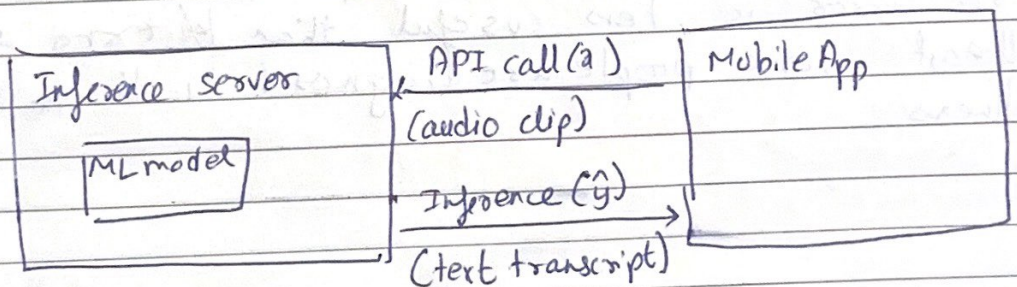
transferring parameters of each layer except final layer from trained NN to our NN.

After transfer training our NN helps smoothen out the NN for its tasks.

Full Cycle of a Machine Learning Project



Deployment:



Software Engineering may be needed for.

- Ensure reliable & efficient predictions
- Scaling
- Logging
- System monitoring
- Model updates

Skewed Data Set

Skewed datasets are data sets that do not have a 50-50 classification training set.

Eg. Train classifier $f_{\vec{w},b}(\vec{x})$ ($y=1$ if disease is present, $y=0$ otherwise)

Find that we've got 1% error on test set (99% correct diagnoses)

If this desease is rare that 0.5% patients have it, then

print("$y=0$") has 99.5% accuracy
0.5% error

So how do we know which algorithm to choose to get best prediction as 0.5% error is less useful than 1% error so atleast some people are diagnosed with the illness.

Precision / Recall

$y = 1$ in presence of rare class we want to detect

Predicted class \ Actual class	1	0
1	True Positive: 15	False Positive: 5
0	False Negative: 10	True Negative: 70

Precision:
of all patients predicted $y = 1$, what fraction actually have the rare disease?

$$\frac{\text{True positives}}{\text{True pos} + \text{False Pos}} = \frac{15}{15+5} = 0.75$$

Recall:
of all patients that actually have the rare disease, what fraction did we correctly detect

$$\frac{\text{True Positives}}{\text{True pos} + \text{False neg}} = \frac{15}{15+10} = 0.6$$

Precision and Recall Tradeoff

Suppose logistic regression: $0 < f_{\vec{w},b}(\vec{x}) < 1$

↳ Predict 1 if $f_{\vec{w},b}(\vec{x}) \geq 0.5$

↳ Predict 0 if $f_{\vec{w},b}(\vec{x}) < 0.5$

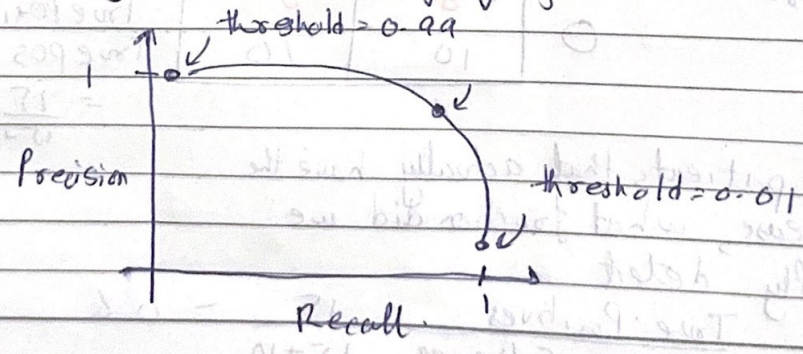
Suppose we want to predict $y=1$ (rare disease) only if very confident

higher precision, lower recall

Suppose we want to avoid missing too many cases of rare disease (when in doubt predict $y=1$)

lower precision, higher recall

More generally, predict 1 if: $f_{\vec{w},b}(\vec{x}) \geq$ threshold



We can chose the threshold manually or using 'F1 score'

Eg.

	Precision (P)	Recall (R)	F_1 Score
Algorithm 1	0.5	0.4	0.444 ← better
Algorithm 2	0.7	0.1	0.175
Algorithm 3	0.02	1.0	0.0392

$$F_1 \text{ Score} = \frac{1}{\frac{1}{2}\left(\frac{1}{P} + \frac{1}{R}\right)}$$

$$= \boxed{\frac{2PR}{P+R}}$$

Harmonic Mean Formula in Maths.