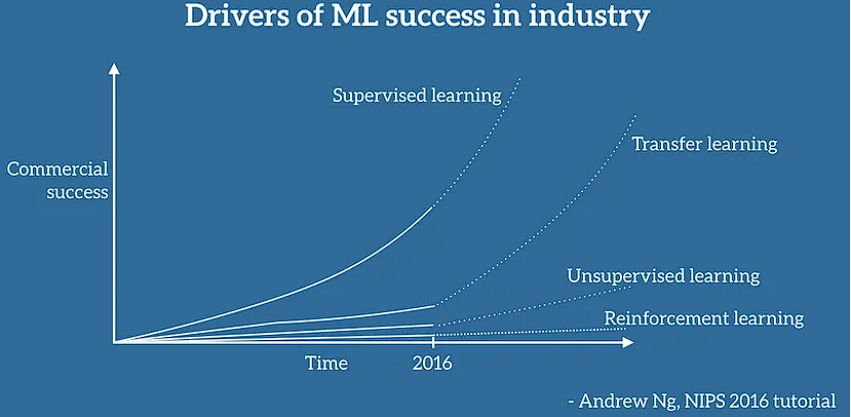
**Transfer Learning**

****

What is transfer learning?

Instead of building a neural network from scratch to solve our task we can reuse existing models (vgg-16, resnet, etc.,) which is already trained on a different dataset. Since learning from scratch is a tedious task as DL methods are data hungry and we need lots of labelled data (huge datasets) even if we download from the internet, still the labels would be not there.

Advantages:-

It saves time.

We can get great results.

No need to set the parameter and then validate.

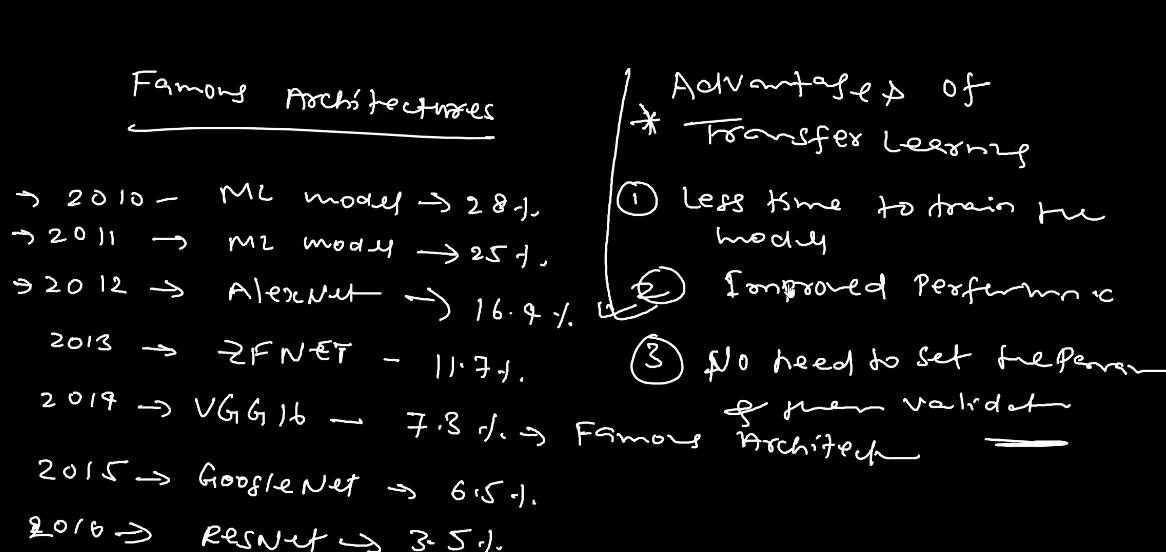
We can say that we already have a solved problem and that solution is the Pre-Trained model(model which is already trained on huge datasets mnist, fashion mnist, etc.,). For ex, ImageNet competition came and some researchers participated in that competition and they solved the problem then. And they knew that whenever a new problem comes they can easily use that model which they created, rather than creating a new model from scratch.

The learned models have already learnings for weight and bias matrices so we can easily use these weights and bias calculations in my new problem.

How the training happens using a pre-trained model?

1. Import the necessary libraries.
2. Load the dataset.
3. Pre-processing
4. Load weight and biases of pre-trained model. \*\*
5. Fine tune the model for current problem. \*\*
6. Validate if the model is working fine, iterate again if it does not works.

The steps 4 and 5 are called Transfer Learning. Once the model is ready, we can use it to make predictions on new data.



Transfer learning is of two types:

1. Feature extraction:- freeze all the parameters and only make changes in the DNN side.
2. Fine Tuning:- here we can change the parameter along with the DNN model as well.

Note:- if we have the same labels in our new data similar to that of our pre-trained model, we can use feature extraction. Also, if the labels are completely different we can go with Fine Tuning (the case which we mostly encounter).

