**TRANSFER LEARNING**

1. Transfer learning is a machine learning method where a model developed for a task is reused as a starting for the second task.
2. It trains the models on a large-scale dataset and then we can use the pre-trained model for another tasks.
3. In NLP, pre-trained models help us to capture and learn a variety of linguistic phenomena from the large corpus. Also using pre-trained models can be a huge time saver also.
4. Transfer learning is an optimization, a shortcut to saving time or getting better performance.
5. Fine-tuning means taking weights of a trained neural network and use it as initialization for a new model being trained on data from the same domain. It is used to speed up the training and overcome the small data size.
6. In general, if our dataset is not drastically different in context from the dataset which the pre-trained model is trained on, we should go for fine-tuning.
7. **Before BERT, fine-tuning was done by using the following models -**
8. **ELmo** - It uses bi-directional LSTM . Every sentence is read twice while processing the data. Once it is read from left to right during the forward pass and next it is read from right to left during the backward pass. The information from both the passes is merged/concatenated. This will help to get the context of the words which is not achieved during Word2Vec or Glove. But LSTM are type of RNN and RNN has a downside in NLP applications as they cannot remember long sequences.

Please refer to the link for learn more about ELmo - https://www.analyticsvidhya.com/blog/2019/03/learn-to-use-elmo-to-extract-features-from-text/?utm\_source=blog&utm\_medium=demystifying-bert-groundbreaking-nlp-framework

1. **OpenAI - GPT** - It uses transformers but this model is not bi-directional. It uses only left-to-right context.
2. **BERT uses transformers and it is also bi-directional.**
3. **Sequence-2-Sequence models in NLP -** A sequence to sequence model aims to map a fixed-length input with a fixed-length output where the length of the input and output may differ. The encoder-decoder also uses RNN and hence there is a disadvantage of losing the data or not remembering the long sequences. The Encoder is a stack of several recurrent units (LSTM or GRU cells for better performance) where each accepts a single element of the input sequence, collects information for that element and propagates it forward.Each new state of the encoder is computed from the previous word and next word. The final state of the encoder conveys the info to start decoding. Encoder acts as the initial hidden state of the decoder part of the model. Each decoder unit accepts a hidden state from the previous unit and produces and output as well as its own hidden state. If the provided sequence is very long then we might loose the information from the beginning of the sentence. To overcome this drawback, "ATTENTION MECHANISM" was added to RNN. During the decoding phase , we add an input called the "context vector" to our cells. Basically, "context vector" is used to convey the global information about the whole input sequence.
4. **Disadvantage of RNN** - It uses sequential processing and it is not global enough and loses information for long sequences.
5. Research paper for transformers - "All we need is Attention" - https://arxiv.org/abs/1706.03762
6. Transformers uses attention mechanism with encoder and decoder. Transformers capture the meaning of the whole sentence instead of going word by word. It does this by combining each items by the others.
7. **Check the video for more information regarding transformers -** https://www.youtube.com/watch?v=TQQlZhbC5ps