(a) What is the difference between a filter and a kernel in CNNs? Illustrate your answer.

A “Kernel” refers to a 2D array of weights. The term “filter” is for 3D structures of multiple kernels stacked together. Thus, Filters are always one dimension more than the kernels. For a 2D filter, filter is same as kernel. But for a 3D filter and most convolutions in deep learning, a filter is a collection of kernels.

(b) Convolution vs. Cross-Correlation - does it really matter and what is mostly used in apps like TensorFlow? Hint: in most cases it does not. However, for the final derivatives, the convolution was needed (recall the slides). Review the web and/or this link (or other links if you prefer) and write your view.

RE: <https://glassboxmedicine.com/2019/07/26/convolution-vs-cross-correlation/>

https://www.tensorflow.org/api\_docs/python/tf/nn/convolution

Cross-Correlation: sliding a kernel (filter) across an image.

Convolution: sliding a **flipped** kernel (filter) across an image.

In most case it does not because most convolutional neural networks in machine learning libraries are implemented using cross-correlation, but it doesn’t change the results in practice because if convolution were used instead, the same weight values would be learned in a flipped orientation.

Thus cross-correlation is mostly used in apps like TensorFlow but involve with flipped configuration.

(c) What is "Attention"?

It is the ability to dynamically highlight and use the salient parts of the information at hand

It is a form of iterative re-weighting, which dynamically highlights different components of a pre-processed input as they are needed for output generation. This makes it flexible and context dependent.

(d) What is LSTM? What is GRU? How do they compare?

LSTM: long short term memory is a special kind of RNN capable of learning long term sequences and order dependence. LSTM has **three** gates (namely input, output and forget gates).

GRU: Gated Recurrent Unit is a special kind of RNN capable of learning long term sequences and order dependence. GRU has **two** gates (reset and update gates)

These gates decide what information is allowed through to the output and can be trained to retain information from farther back. This allows it to pass relevant information down a chain of events to make better predictions.

GRUs train faster and perform better than LSTMs on less training data if you are doing language modeling.

GRUs are simpler and thus easier to modify, for example adding new gates in case of additional input to the network.

LSTMs should in theory remember longer sequences than GRUs and outperform them in tasks requiring modeling long-distance relations.

(e) What is "Transfer Learning"?

Transfer learning is a machine learning method where we reuse a pre-trained model as the starting point for a model on a new task.

To put it simply—a model trained on one task is repurposed on a second, related task as an optimization that allows rapid progress when modeling the second task.

(f) What is the key (major) difference between RNNs vs CNN/ANNs?

RNNs take dependent inputs (sequential data), same parameters(weights), need memory stores all information, recurrent process for each state with same function

CNN/ANNs take picture inputs, update weights and biases, different function for different layers

(g) What causes a vanishing or exploding gradient in RNNs? What are three methods to mitigate this?

Due to backpropagation in the RNNs, the vanishing occurs when multiply many small numbers together, exploding occurs when activation function is improperly used. And, for parameter initialization for large weights or biases

1. Choose relatively ‘better’ activation function: relu, tanh, sigmoid, …
2. Initialize weights to identity matrix and biases to zero
3. Choose learning rate properly with hyperparameter tuning