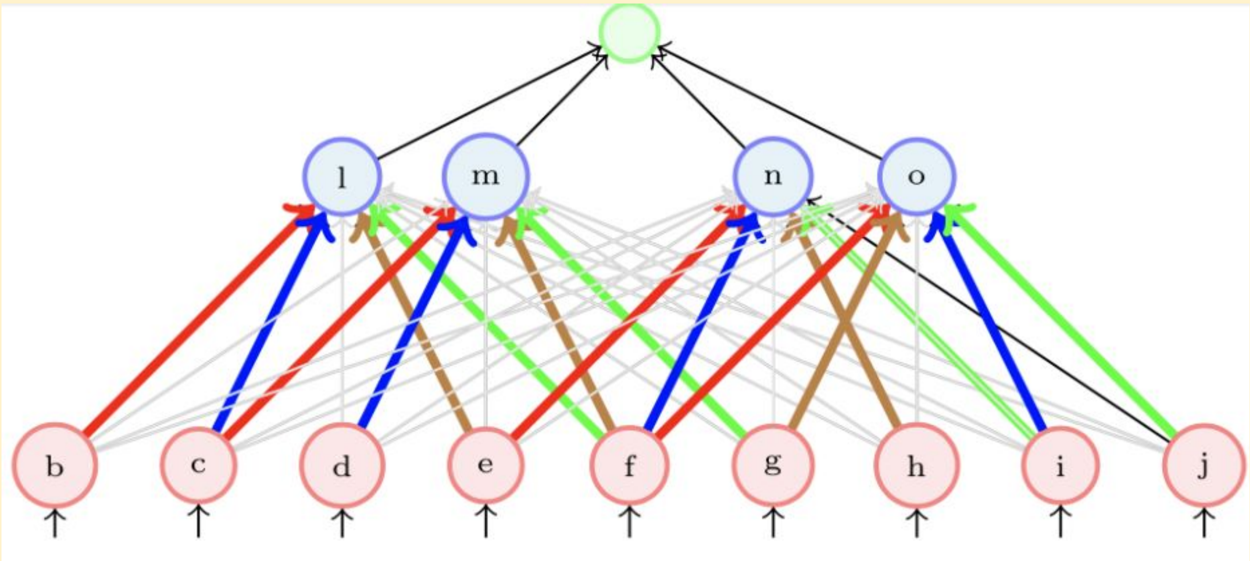
PadhAI: From Convolution Operation to Neural Network

One Fourth Labs

Summary and what next

Making sense of everything we have seen so far

1. By the Universal Approximation Theorem, we have learned that DNNs are powerful function approximators.
2. They can be trained using backpropagation
3. However, due to the large number of parameters, the function can become extremely complex, resulting in a high chance of overfitting the training data.
4. We looked into CNNs to see if we could have a Neural Network which are complex (many non-linearities) but with fewer parameters and hence be less prone to overfitting.



5. CNNs solve both of those shortcomings, with key points such as weight sharing and sparse connectivity. Non-Linearity functions such as ReLU are applied after each convolutional layer.
6. Training CNNs is also very similar to training FFNs, the only difference being we take a 0 value for all weights that we are not interested in.
7. We will then be able to apply learning to the filters(parameters), thereby reducing the overall loss of the CNN.
8. They can be easily implemented with frameworks such as PyTorch or Tensorflow.