**LSTMs vs. Convolutions vs. GRUs**

### Convolutional Layers for Text

Just like you did with images, you can also use convolutional layers on text, where the convolution occurs across a sequence of words instead of across an image.

To use a convolutional layer on text inputs, you can place a Conv1D layer directly after the Embedding layer:

*# A 1D Convolutional layer with 128 filters and 5 words per filter*

tf.keras.layers.Conv1D(128, 5, activation=’relu’)

Note that you will need to use Flatten or GlobalAveragePooling1D on the output of this layer to connect to any fully-connected layers from there.

### GRUs

Gated Recurrent Units, or GRUs, have “update” and “reset” gates. These gates decide what to keep and what to throw away. They do not have a “cell state” like LSTMs do.

The code for these is very similar to an LSTM, where the GRU layer is wrapped in a Bidirectional layer.

*# A bidirectional GRU layer with 32 nodes*

tf.keras.layers.Bidirectional(tf.keras.layers.GRU(32))

### Comparing Training Amongst Layers

Next up, let’s take a quick look at how each of these models performed during training, and the total duration of training.

