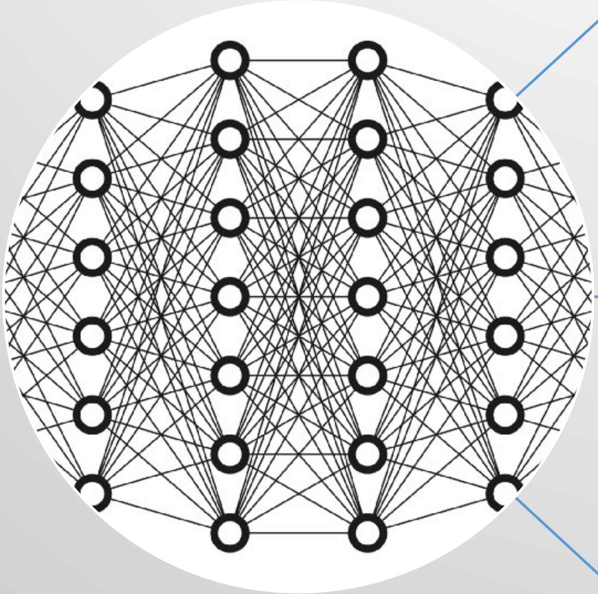
Natural Language Processing

Dr. Karen Mazidi

Part Six:
Deep Learning



Topics

- RNNs
- CNNs

Quizzes

- Q: DL; DLVariations

Homework

- Homework: Text classification

CNNs

- convolutional neural networks or covnets
- densely connected sequential layers learn global patterns in the data
- CNNs learn patterns in small windows
- Advantages:
    - a pattern learned in one location is recognized elsewhere
    - layers can learn hierarchies of shapes from edges and other features

CNN convolution

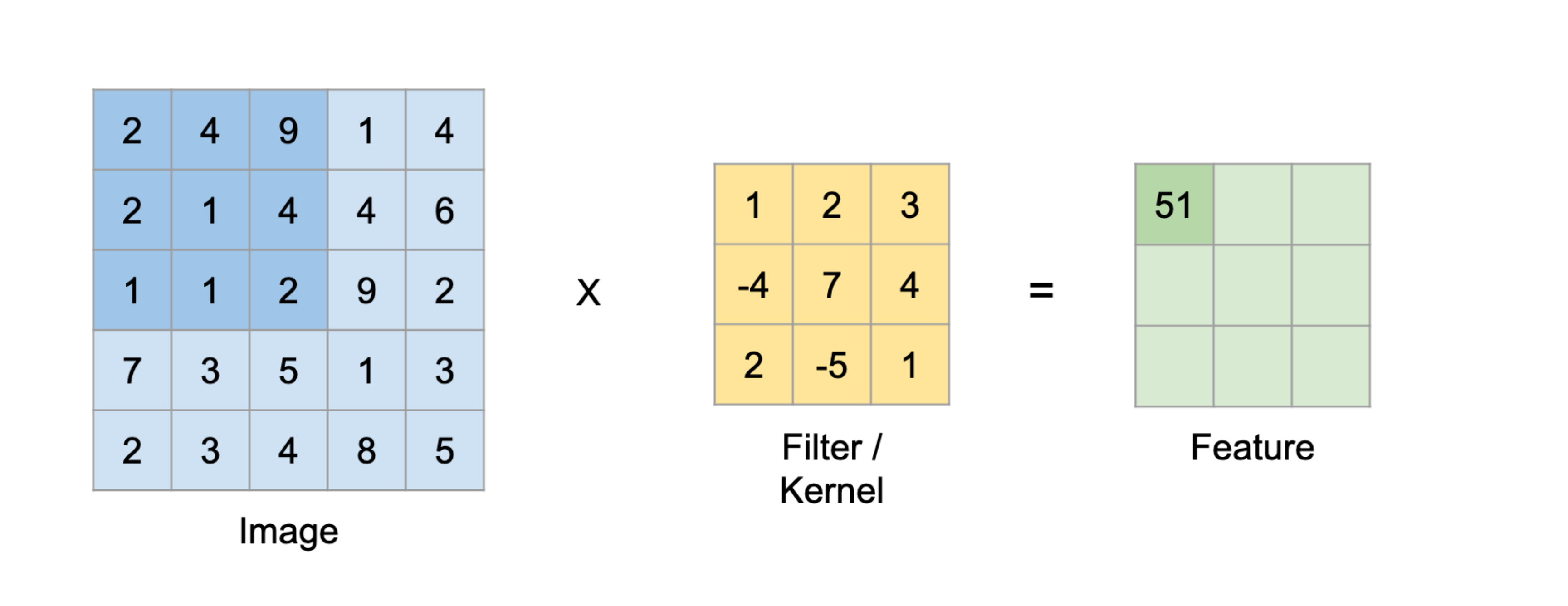
- a 4x4 ‘filter’ slides over the data, performing the convolution function

0.47557	0.13031	0.26269	0.98775	0.54559	0.70388	0.41101	0.10889
0.25782	0.69232	0.53866	0.20306	0.01652	0.45732	0.49489	0.47130
0.87015	0.03241	0.00089	0.95473	0.25201	0.67926	0.66318	0.35740
0.13696	0.20884	0.20363	0.72029	0.26433	0.42732	0.87660	0.59141
0.51279	0.81518	0.50046	0.89543	0.77181	0.77192	0.45861	0.25983
0.03777	0.12560	0.54588	0.06574	0.31243	0.50573	0.60777	0.85029
0.82038	0.42600	0.16205	0.80647	0.10582	0.45355	0.59760	0.08356
0.71715	0.42875	0.85921	0.60168	0.92237	0.62636	0.71523	0.14542
0.09399	0.43249	0.84148	0.23740	0.30299	0.93350	0.03851	0.33104
0.30386	0.63560	0.72024	0.38294	0.78565	0.72367	0.52017	0.93030
0.97332	0.02479	0.31189	0.74439	0.62472	0.62113	0.13827	0.92139

Figure 24.1: Convolving

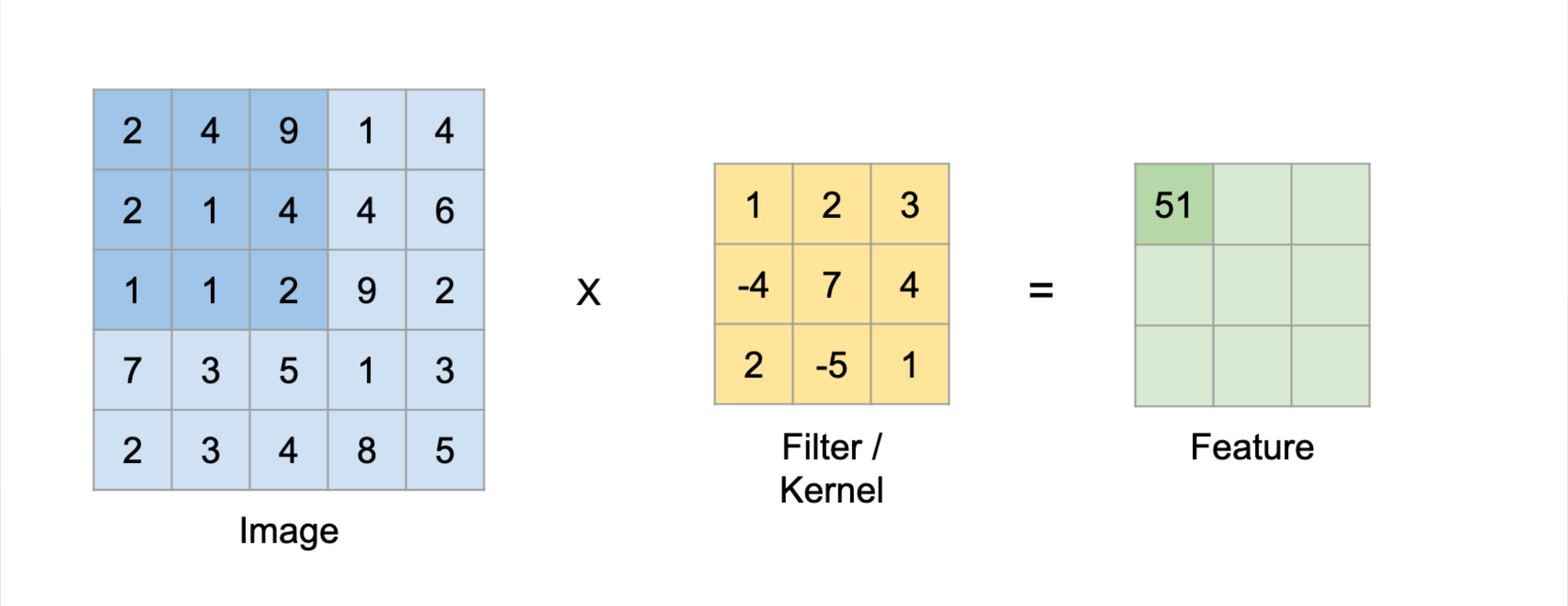
convolution

- convolution – mathematical process of combining two functions
- the filter (aka kernel) moves with overlap in strides, the smaller the stride, the more the overlap



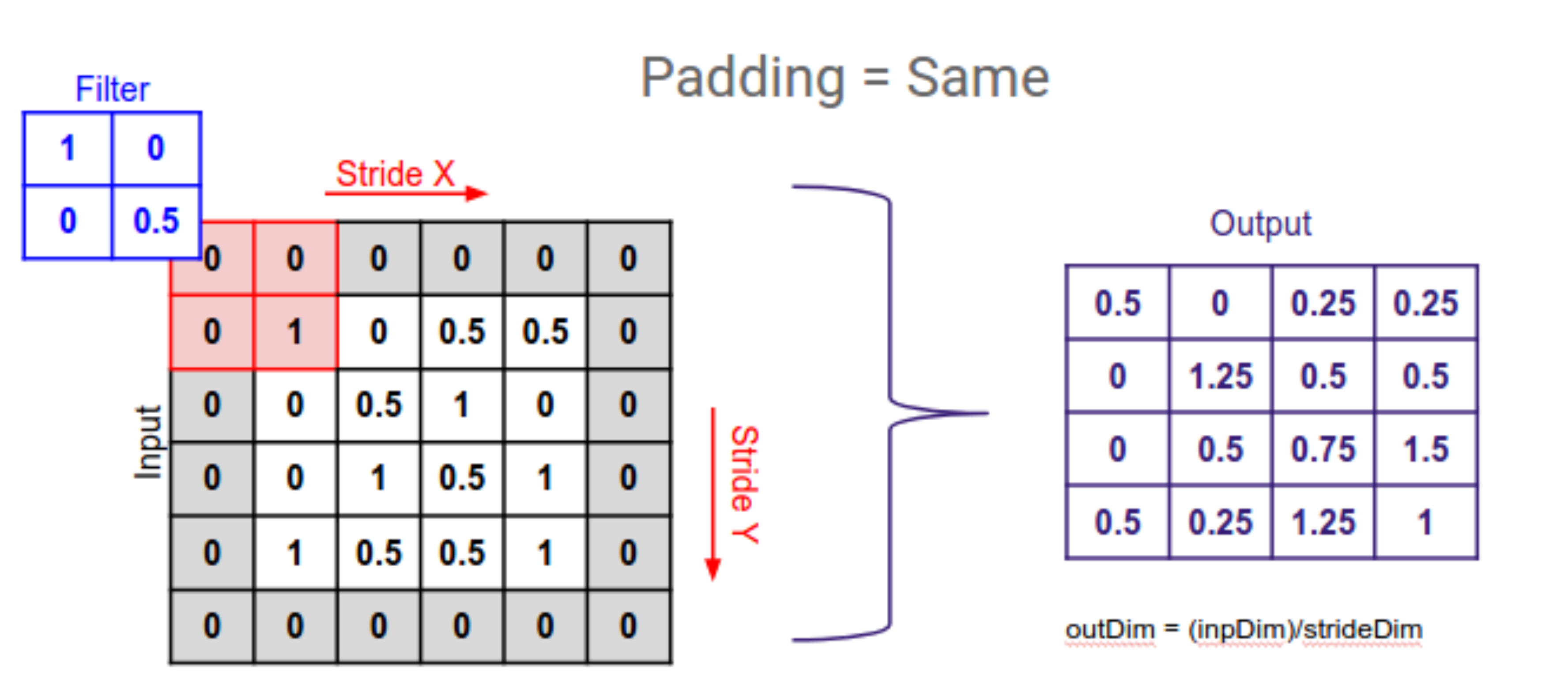
padding

- notice the shrinkage of the data, at least one per dimension
- this can be avoided by padding the data



padding

- padding = same gives same output size of data
- padding = none doesn't do padding

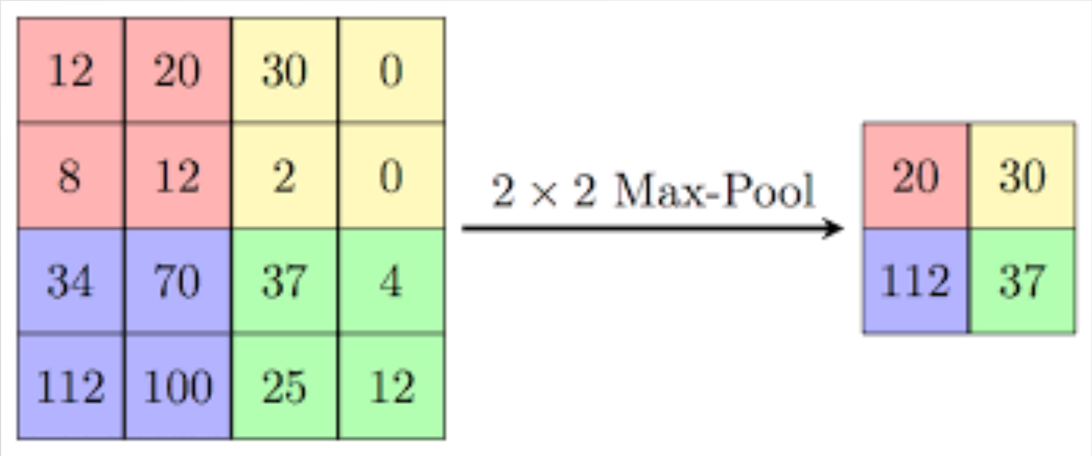


CNN

- Conv1D layers work well on text data
- stacks of conv layers and max-pooling layers are common, followed by a flatten layer, then a dense layer for the final classification
- max pooling also reduces dimensions and helps prevent overfitting

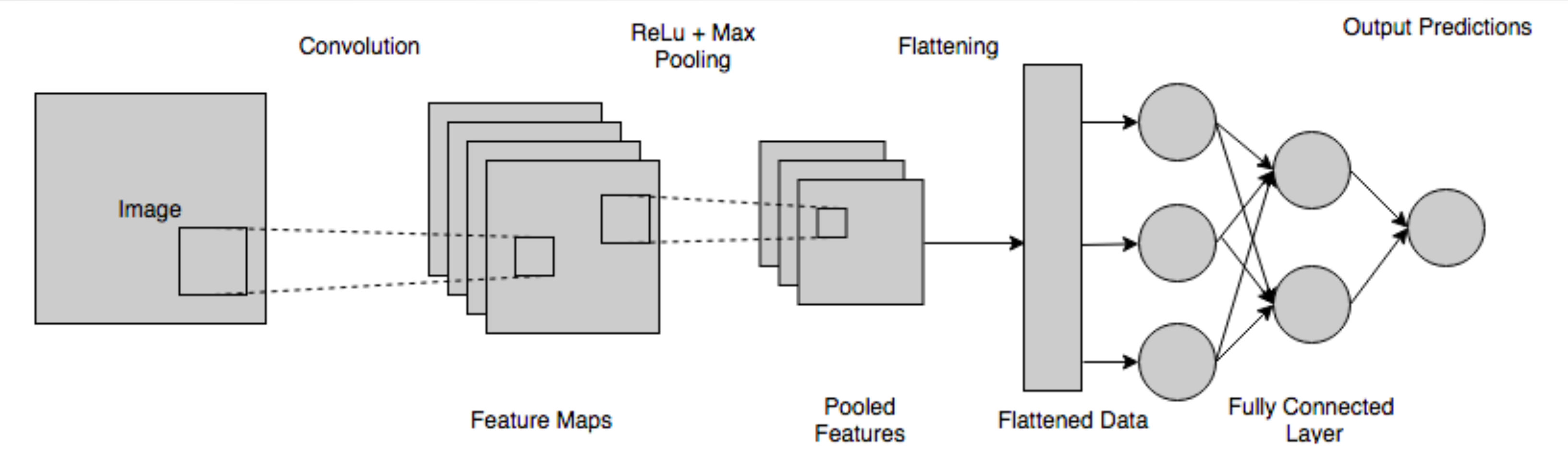
max pooling

- dimensionality reduction



Flatten

- after conv-max pooling layers, flattening reshapes the data for further processing



CNN visualization

- https://www.youtube.com/watch?v=YRhxdVk_sIs

Keras: IMDB data

- each example was shortened or padded to length 500
- input shape is (25000, 500)
- the embedding layer learns connections between words

```
model = models.Sequential()
model.add(layers.Embedding(max_features, 128, input_length=maxlen))
model.add(layers.Conv1D(32, 7, activation='relu'))
model.add(layers.MaxPooling1D(5))
model.add(layers.Conv1D(32, 7, activation='relu'))
model.add(layers.GlobalMaxPooling1D())
model.add(layers.Dense(1))
```

Keras: IMDB data

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 500, 128)          1280000
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 494, 32)           28704
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 98, 32)            0
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 92, 32)            7200
_________________________________________________________________
global_max_pooling1d_1 (Glob (None, 32)                0
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 33
=================================================================
Total params: 1,315,937
Trainable params: 1,315,937
Non-trainable params: 0
```

Keras: IMDB data

- train and test as before
- results: a couple of points higher than the sequential model

Recurrent models

- a recurrent neural network, RNN, has memory, or state, which enables it to learn a sequence
- the looping mechanism produces a new hidden state at each iteration
- the final hidden state is a representation of previous states

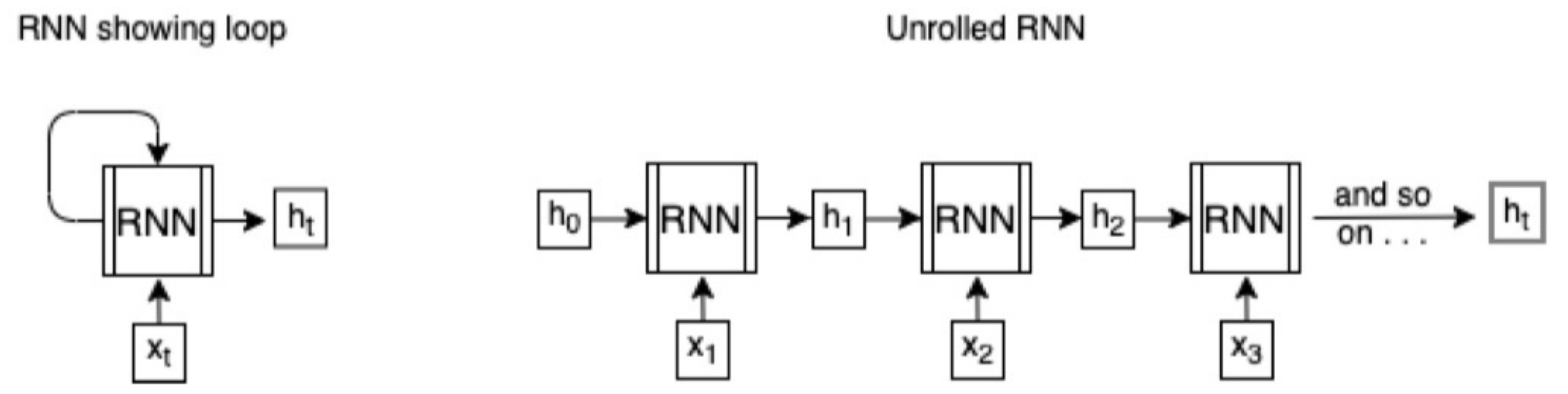


Figure 24.3: Recurrent Neural Network

RNNs

- vanishing gradient problem: with more layers, the back-propagated gradient becomes smaller and smaller
- LSTM (Long Short-Term Memory) is an RNN variation that helps the vanishing gradient problem
- keeps memory path independent of the back prop path
- LSTM allows information to be remembered or forgotten
- GRU is simpler than LSTM and may train faster

LSTM and GRU visualization

- https://www.youtube.com/watch?v=8HyCNIVRbSU

RNN on IMDB data

```
model = models.Sequential()
model.add(layers.Embedding(max_features, 32))
model.add(layers.SimpleRNN(32))
model.add(layers.Dense(1, activation='sigmoid'))
```

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_3 (Embedding)      (None, None, 32)          320000
_________________________________________________________________
simple_rnn_1 (SimpleRNN)     (None, 32)                2080
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 33
=================================================================
Total params: 322,113
Trainable params: 322,113
Non-trainable params: 0
```

LSTM on IMDB data

```
# build a model with LSTM
model = models.Sequential()
model.add(layers.Embedding(max_features, 32))
model.add(layers.LSTM(32))
model.add(layers.Dense(1, activation='sigmoid'))
```

GRU on IMDB data

```
model = models.Sequential()
model.add(layers.Embedding(max_features, 32))
model.add(layers.GRU(32))
model.add(layers.Dense(1, activation='sigmoid'))
```

Code Examples

- Keras imdb 2 with RNN
- Keras imdb 3 with CNN

Essential points to note

- CNNs perform well on image data but also work on text data
- RNNs were created for sequential data like text, but suffer from vanishing gradients
- LSTM and GRU are improvements over the RNN

To Do

- Quiz on deep learning variations
- Portfolio: Text classification

Next topic

embeddings