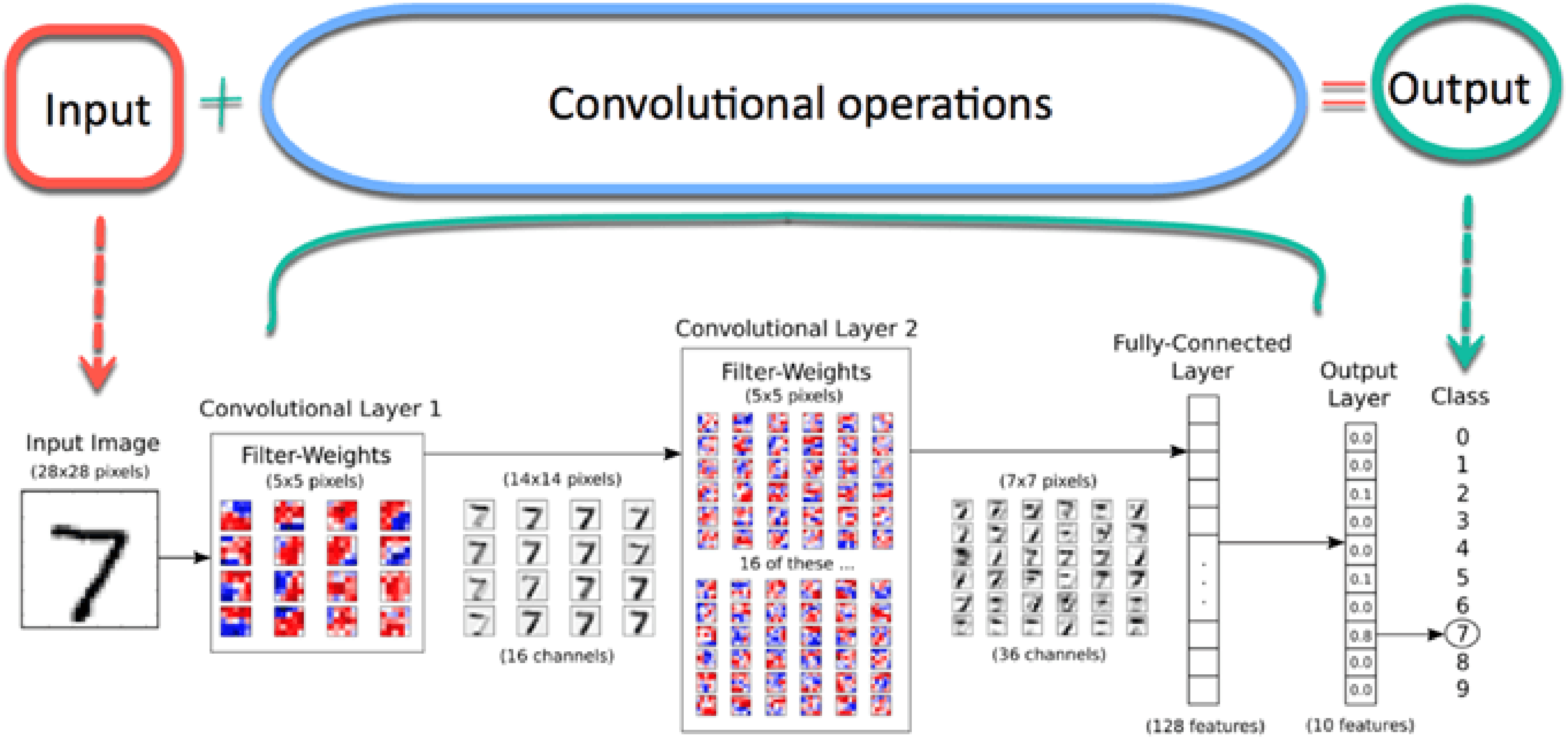
Convolutional Neural Network w problemie MNIST

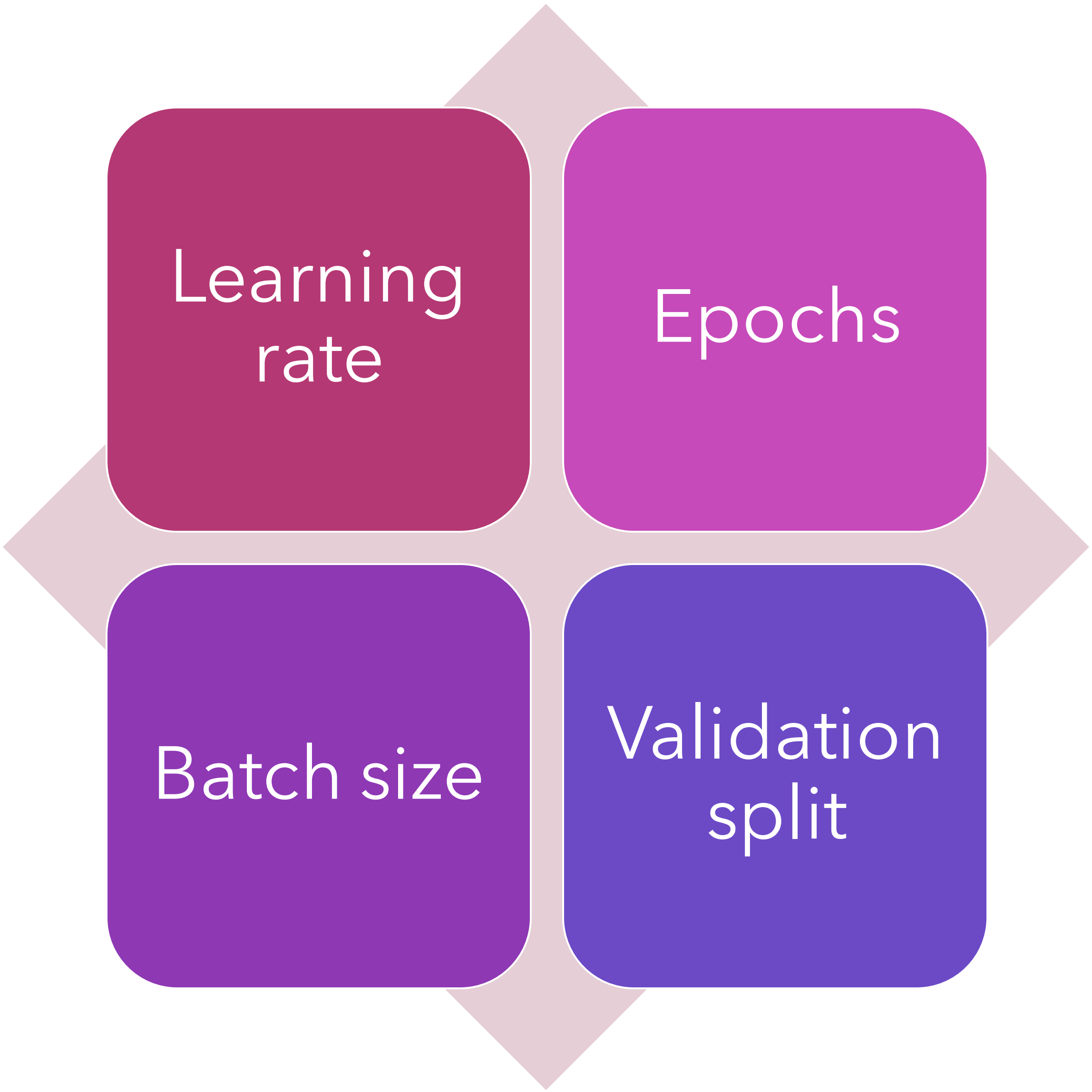
Przygotował Michał Tracewicz

Proces uczenia

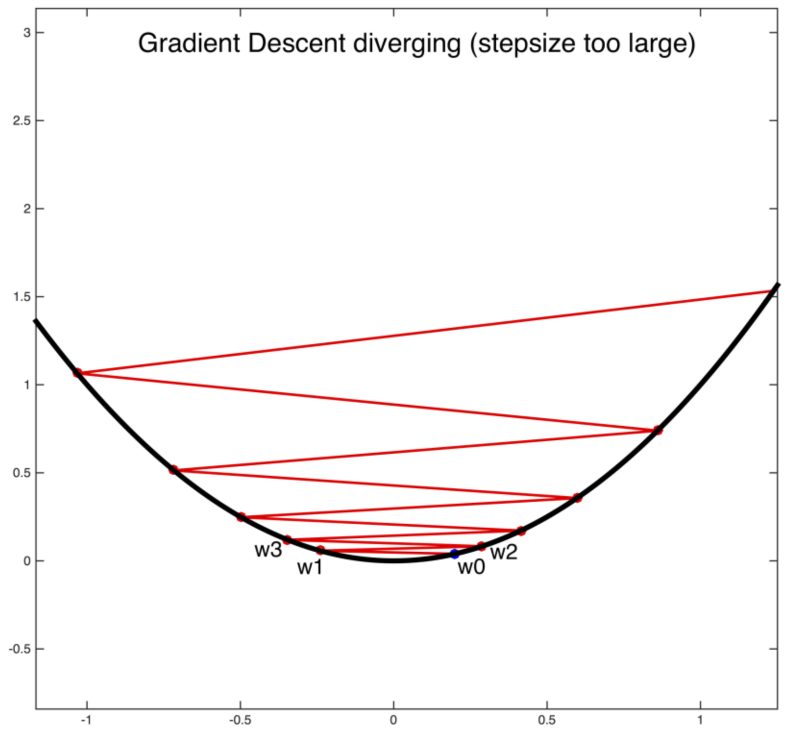


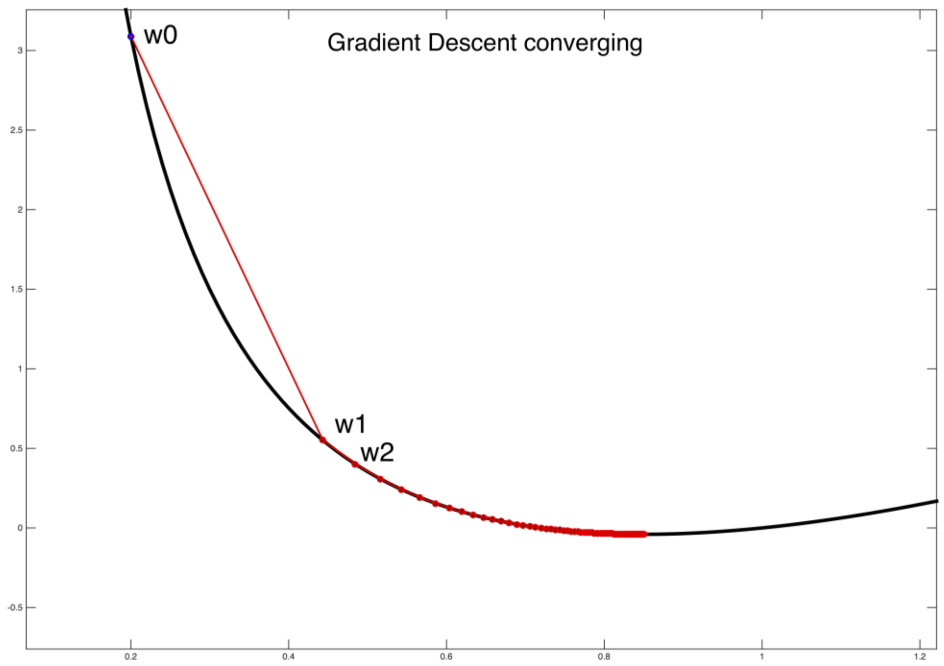
https://www.guru99.com/convnet-tensorflow-image-classification.html

Parametry



Metoda gradientu prostego





Architektura sieci

```
model.add(tf.keras.layers.Conv2D(filters=15,kernel_size=2,padding='same',activation='relu',
input_shape=(28,28,1)))
model.add(tf.keras.layers.MaxPooling2D(pool_size=2))
model.add(tf.keras.layers.Dropout(rate=0.2))

model.add(tf.keras.layers.Conv2D(filters=32, kernel_size=2,padding='same',activation='relu'))
model.add(tf.keras.layers.MaxPooling2D(pool_size=2))
model.add(tf.keras.layers.Dropout(rate=0.2))

model.add(tf.keras.layers.Conv2D(filters=64, kernel_size=2,padding='same',activation='relu'))
model.add(tf.keras.layers.MaxPooling2D(pool_size=2))
model.add(tf.keras.layers.Dropout(rate=0.2))

model.add(tf.keras.layers.Flatten())
model.add(tf.keras.layers.Dense(units=512, activation='sigmoid'))
model.add(tf.keras.layers.Dropout(rate=0.2))
model.add(tf.keras.layers.Dense(units=10, activation='softmax'))
```

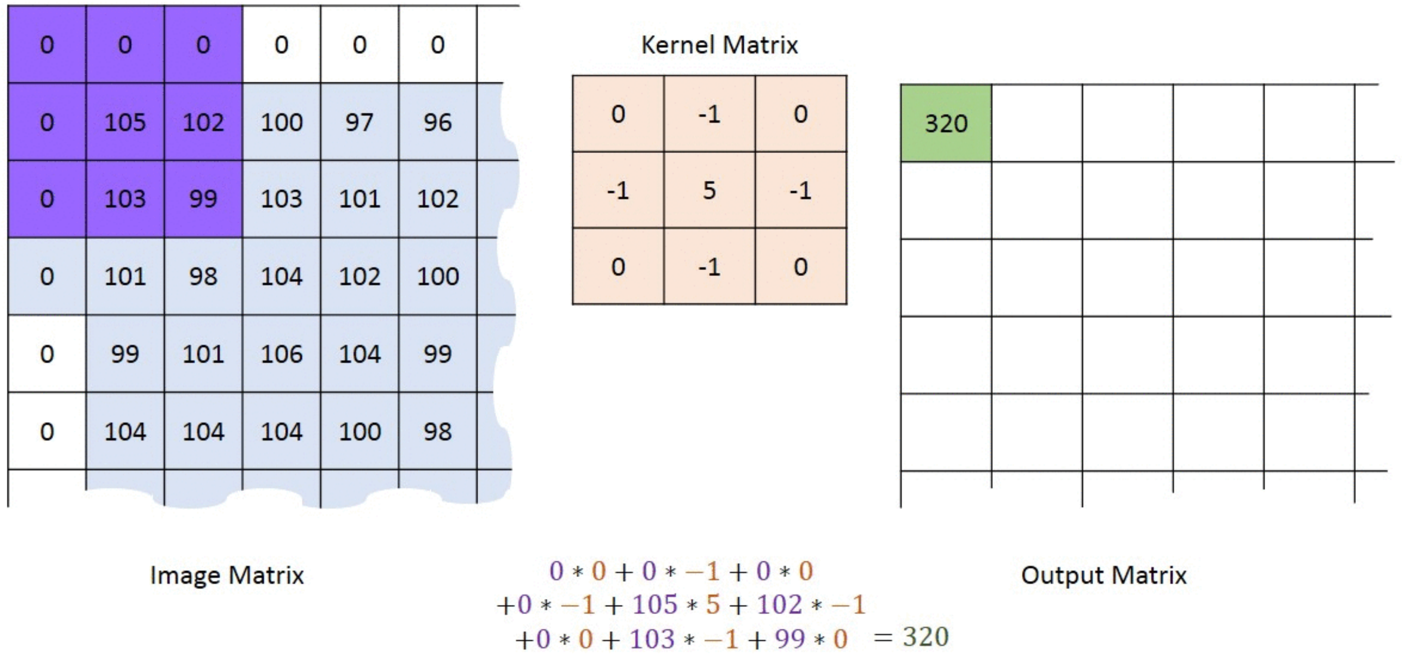
Warstwa wejściowa (ang. Input)

- Obraz 28x28x1 (trzy wymiary)
- Tablica zawierająca 784 elementy (jeden wymiar)
- Każdy piksel ma wartość z przedziału [0,255]
- Przekonwertowane na wartości z przedziału [0.0,1.0]

Rodzaje warstw ukrytych

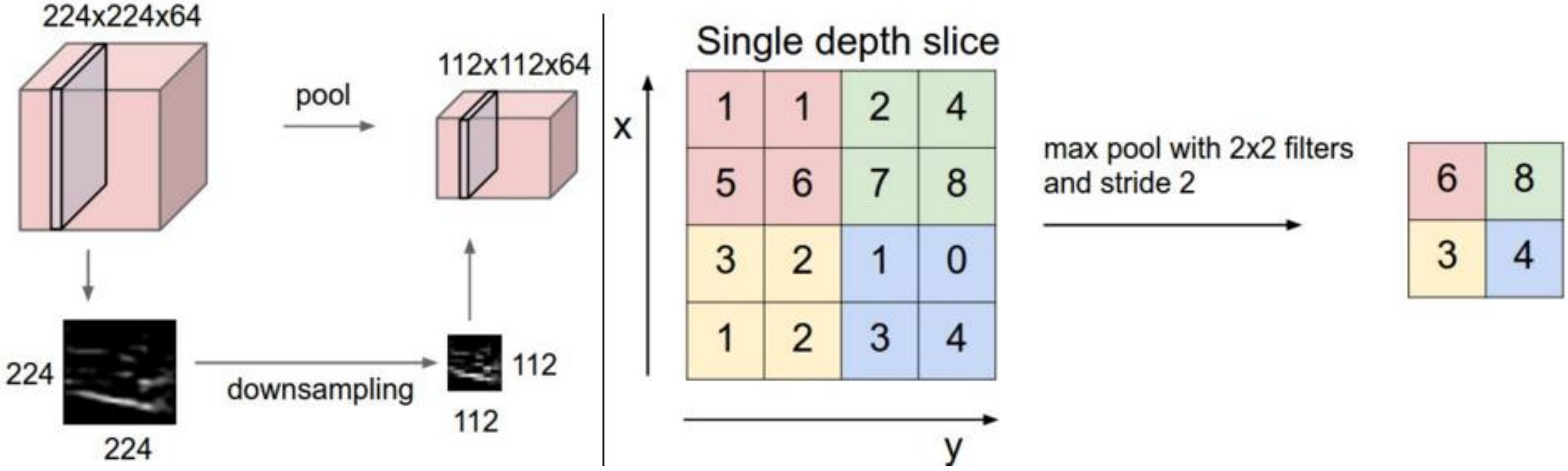
- Convolutional
- Pooling
- Dropout
- Dense

Splotowa (ang. Convolutional)

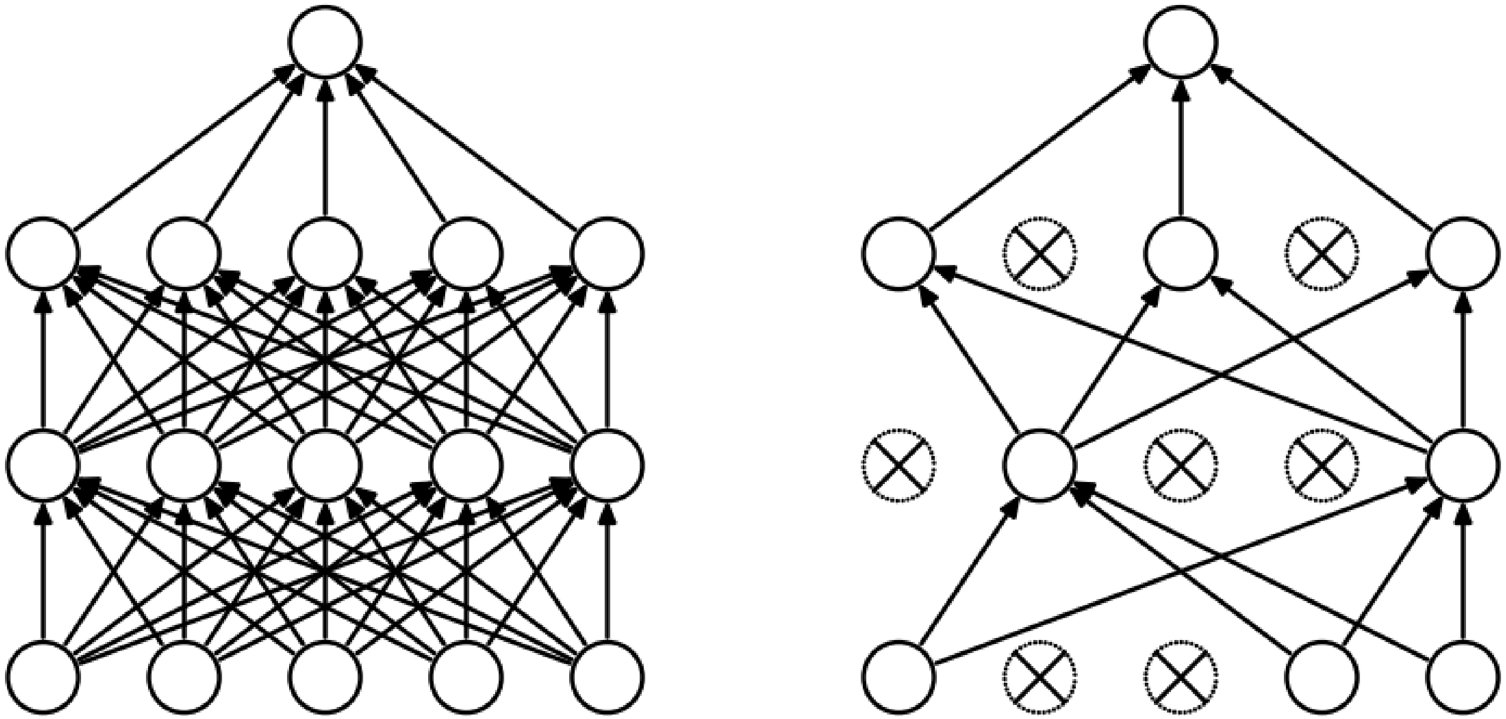


Convolution with horizontal and vertical strides = 1

Łącząca (ang. Pooling)



Odrzucająca (ang. Dropout)



(a) Standard Neural Net (b) After applying dropout.

W pełni połączona (ang. Dense) oraz warstwa wyjściowa (ang.output)