

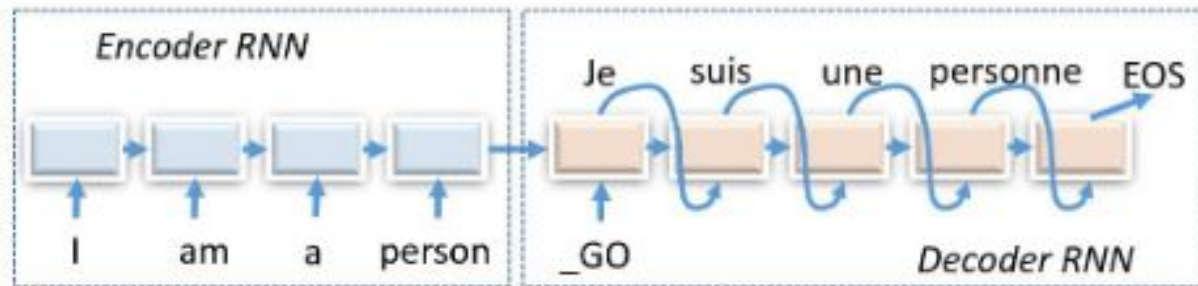
Rekurencyjne Sieci Neuronowe

Marcin Wierzbński



Przetwarzanie sekwencyjne

- Przetwarzanie sekwencyjne odnosi się do technik analizy danych, w których dane są ułożone w sekwencje
- Sekwencje mogą być szeregami czasowymi, sekwencjami tekstu, sekwencjami dźwięków itp.
- Jest szeroko stosowane w dziedzinach takich jak przetwarzanie języka naturalnego, rozpoznawanie mowy, analiza sygnałów czasowych i wiele innych.

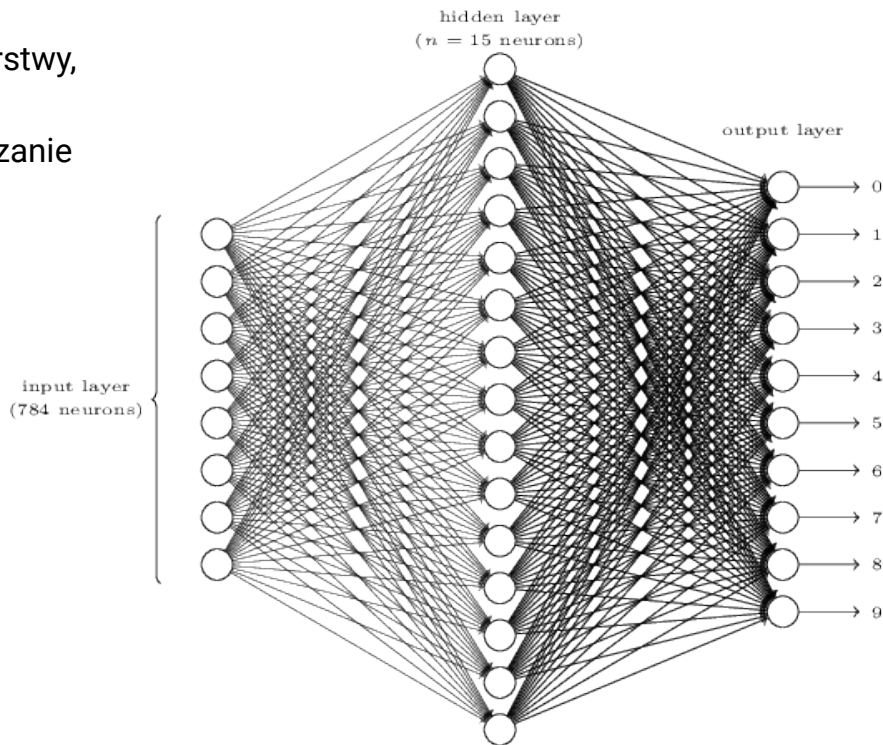


Techniki przetwarzania sekwencyjnego

- Rekurencyjne sieci neuronowe (RNN): RNN są specjalnie zaprojektowane do pracy z danymi sekwencyjnymi. Posiadają pamięć wewnętrzną, która pozwala na przechowywanie informacji o wcześniejszych elementach sekwencji i przetwarzanie ich w kontekście obecnie analizowanego elementu.
- Sieci LSTM (Long Short-Term Memory): LSTM to specjalny rodzaj RNN, który skutecznie radzi sobie z problemem zanikającego gradientu, który może występować w standardowych RNN. LSTM posiadają dodatkowe bramki, które kontrolują przepływ informacji przez sieć.
- Sieci GRU (Gated Recurrent Unit): GRU to kolejny rodzaj RNN, który ma podobne cechy do LSTM, ale zamiast oddzielnych bramek wejściowych, wyjściowych i zapomnienia, wykorzystuje kombinację bramek aktualizacji i resetowania.

Sieć neuronowa

- W sieci neuronowej, neurony są połączone w warstwy, gdzie wyjście jednej warstwy jest podawane na wejście kolejnej warstwy. To umożliwia przetwarzanie informacji na różnych poziomach.



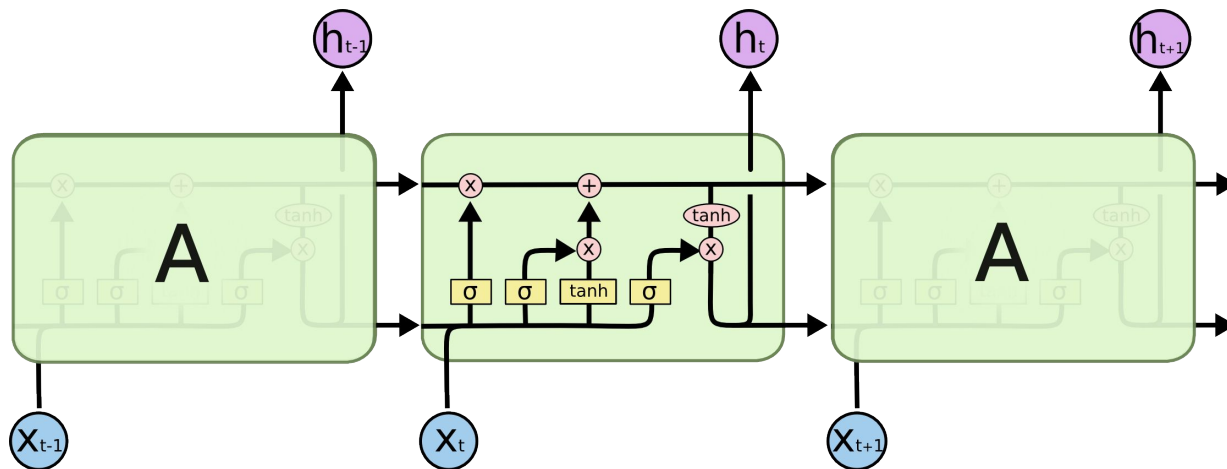
Rekurencyjna sieć neuronowa

- Rekurencyjna sieć neuronowa (RNN - Recurrent Neural Network), to rodzaj sztucznej sieci neuronowej, która ma zdolność do przetwarzania danych sekwencyjnych.
- RNN jest przystosowana do modelowania danych, w których kolejność ma znaczenie, takich jak sekwencje czasowe, zdania, dźwięki itp.
- Podstawową **różnicą** między sieciami rekurencyjnymi a innymi rodzajami sieci neuronowych, takimi jak sieci jednokierunkowe (feedforward networks), jest to, że sieci rekurencyjne mają **cykliczną strukturę**, która pozwala na przesyłanie informacji z **poprzednich kroków** czasowych do **aktualnego kroku**. Oznacza to, że sieć rekurencyjna ma pamięć, która może utrzymywać informacje z przeszłości i wykorzystywać je do podejmowania decyzji w bieżącym kontekście.

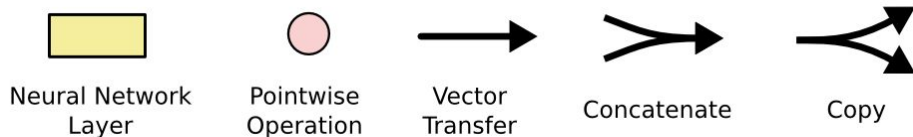
Najpopularniejszym typem sieci rekurencyjnej jest Long Short-Term Memory (LSTM),

Rekurencyjna sieć neuronowa

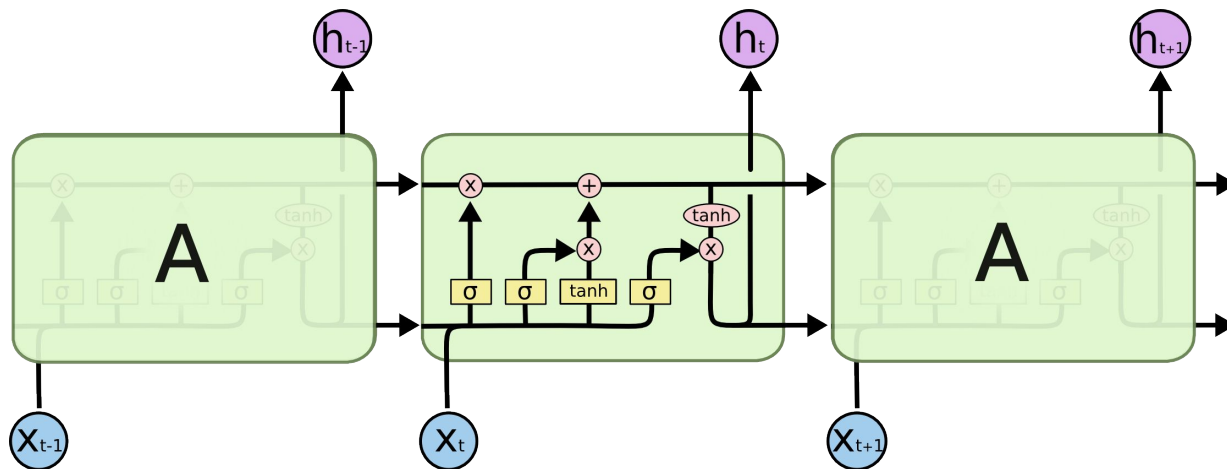
Schemat LSTM



$$\begin{aligned}
 i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i), \\
 f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \\
 c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \\
 o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \\
 h_t &= o_t \odot \tanh(c_t),
 \end{aligned}$$



Schemat LSTM

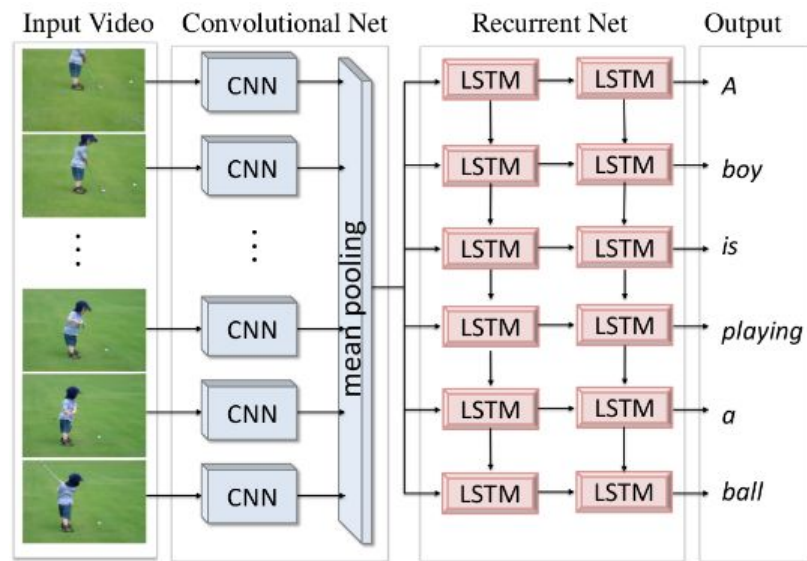


$$\begin{aligned}
 i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i), \\
 f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \\
 c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
 o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \\
 h_t &= o_t \odot \tanh(c_t),
 \end{aligned}$$

LSTM wprowadzają:

- c_t - jednostka stanu, zależna w sposób liniowy od c_{t-1} (liniowa rekurencja)
- o_t wyjściowa bramka
- h_t ukryty stan
- f_t - bramka ukryta
- i_t - bramka wejściowa, aktywacja
- o - produkt Hadamarda (punktowy produkt dla macierzy)

Przetwarzanie sekwencyjne - CNN

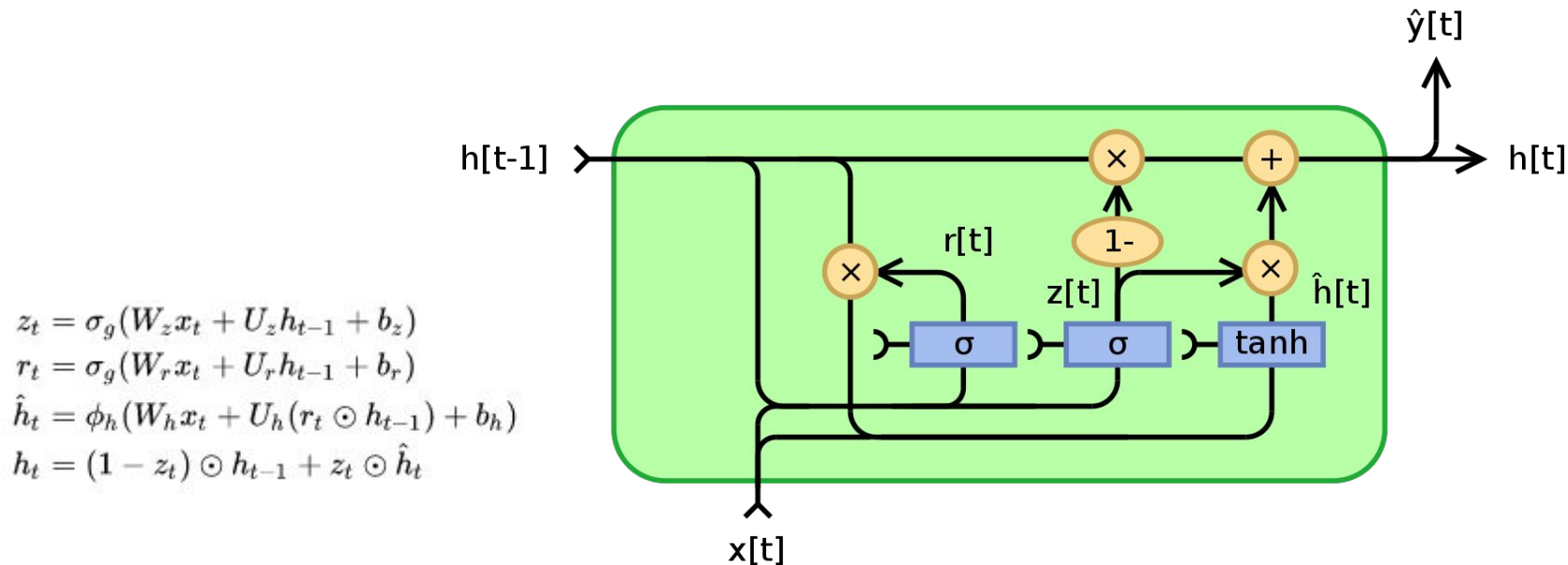


Zastosowanie LSTM:

- przewidywanie szeregów czasowych,
- rozpoznawanie mowy,
- uczenie gramatyczne,
- rozpoznawanie tekstu pisanego,
- tłumaczenie end to end,
- rozpoznawanie kontekstu,

Gated Recurrent Units (GRU)

- RNN cierpi z powodu znikających/wybuchających gradientów i nie może zapamiętać stanów na bardzo długo.
- GRU nie posiada bramki wyjściowej i bramki C. Cho, 2014,
- Mniej parametrów wzgl. LSTM, porównywalna jakość wyników dla wielu problemów
- bramka resetu: $r[t]$
- bramka aktualizacji $z[t]$



Historycznie:

- Pierwsza praca: LSTM: Hochreiter, Schmidhuber, 1997
- W 2009, RNNs ustanowił rekordy w rozpoznawaniu pisma ręcznego. Skalowalny system rozpoznawania tekstu pisanego ręcznie [\[1\]](#)
- W 2014 Kyunghyun Cho et al. wprowadził GRU
- W 2015, rozpoznawanie mowy przez Google otrzymana przy 47% skoku używając LSTM [\[2\]](#)
- Znacząco ulepszone tłumaczenie maszynowe, modelowanie języka i wielojęzyczne przetwarzanie języka (np. Google Translate).

LSTM w Kerasie

```
# Definicja modelu
```

```
model = Sequential()  
model.add(LSTM(4, input_shape=(1, 1)))  
model.add(Dense(1))
```

```
# Kompilacja modelu
```

```
model.compile(loss='mean_squared_error', optimizer='adam')
```

```
# Trenowanie modelu
```

```
model.fit(X_train, y_train, epochs=100, batch_size=1)
```

```
# Ocena modelu
```

```
train_loss = model.evaluate(X_train, y_train)  
test_loss = model.evaluate(X_test, y_test)  
print('Train Loss:', train_loss)  
print('Test Loss:', test_loss)
```

Tokenizacja

Tokenizacja to proces podziału ciągu tekstowego na mniejsze jednostki, nazywane tokenami. Tokeny mogą być pojedynczymi słowami, znakami, lub innymi jednostkami o znaczeniu w kontekście analizowanego tekstu.

Przetwórz teksty na postać liczb lub wektorów za pomocą tokenizacji i kodowania tekstów.

Zbiór danych IMDB

Zbiór danych IMDB (Internet Movie Database) to popularny zbiór danych wykorzystywany w dziedzinie przetwarzania języka naturalnego (NLP) do zadań analizy sentymentu tekstu. Składa się on z recenzji filmowych zebranych z witryny IMDb, która jest jednym z największych internetowych źródeł informacji o filmach.

```
{'fawn': 34701, #
```

```
'tsukino': 52006,...} #fikcyjna postać
```

```
imdb.get_word_index()['best'] # 115
```

```
imdb.get_word_index()['worst'] # 245
```

```
imdb.get_word_index()['year'] #288
```

Warstwa Embedding w Keras

- Warstwa Embedding w Keras służy do przekształcenia indeksów słów (liczb całkowitych) na gęste wektory liczb rzeczywistych o mniejszej liczbie wymiarów.
- Warstwa ta jest często wykorzystywana w modelach przetwarzania języka naturalnego i NLP, gdzie tekst jest reprezentowany jako sekwencja słów.
- Warstwa Embedding przyjmuje na wejściu sekwencje indeksów słów (lub innych jednostek tekstowych) i zamienia je na gęste wektory liczb rzeczywistych.
- Wektory te są inicjalizowane losowo lub mogą być nauczane w trakcie uczenia modelu.
- Proces uczenia polega na optymalizacji tych wektorów w taki sposób, aby minimalizować koszt (stratę) modelu w danym zadaniu (np. klasyfikacja, generowanie tekstu).

Embedding i pad_sequences Keras

```
max_features = 10000
```

```
max_len = 200
```

```
model = Sequential()
```

```
model.add(Embedding(max_features, 128))
```

```
model.add(LSTM(128, dropout=0.2, recurrent_dropout=0.2))
```

```
model.add(Dense(1, activation='sigmoid'))
```

```
X_train = pad_sequences(X_train, maxlen=max_len)
```

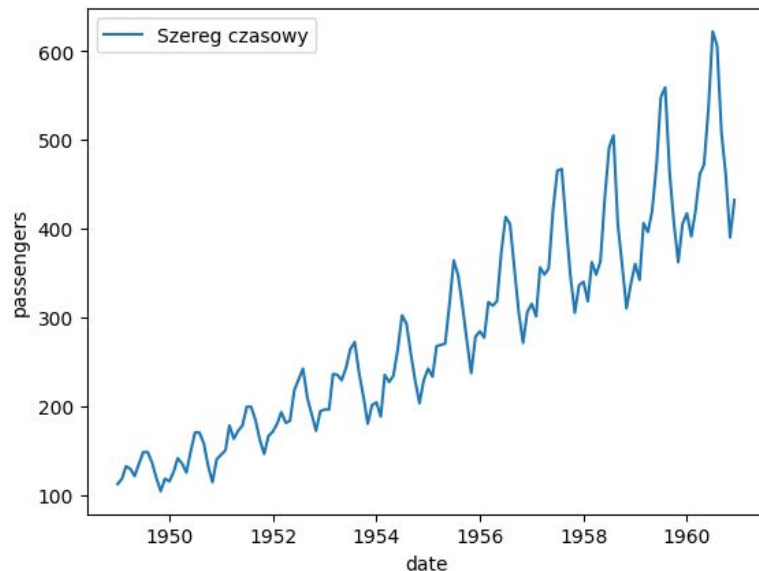
```
X_test = pad_sequences(X_test, maxlen=max_len)
```


Przykład:

? this film was just brilliant casting location scenery story direction everyone's really suited the part they played and you could just imagine being there robert ? is an amazing actor and now the same being director ? father came from the same scottish island as myself so i loved the fact there was a real connection with this film the witty remarks throughout the film were great it was just brilliant so much that i bought the film as soon as it was released for ? and would recommend it to everyone to watch and the fly fishing was amazing really cried at the end it was so sad and you know what they say if you cry at a film it must have been good and this definitely was also ? to the two little boy's that played the ? of norman and paul they were just brilliant children are often left out of the ? list i think because the stars that play them all grown up are such a big profile for the whole film but these children are amazing and should be praised for what they have done don't you think the whole story was so lovely because it was true and was someone's life after all that was shared with us all. **Positive**

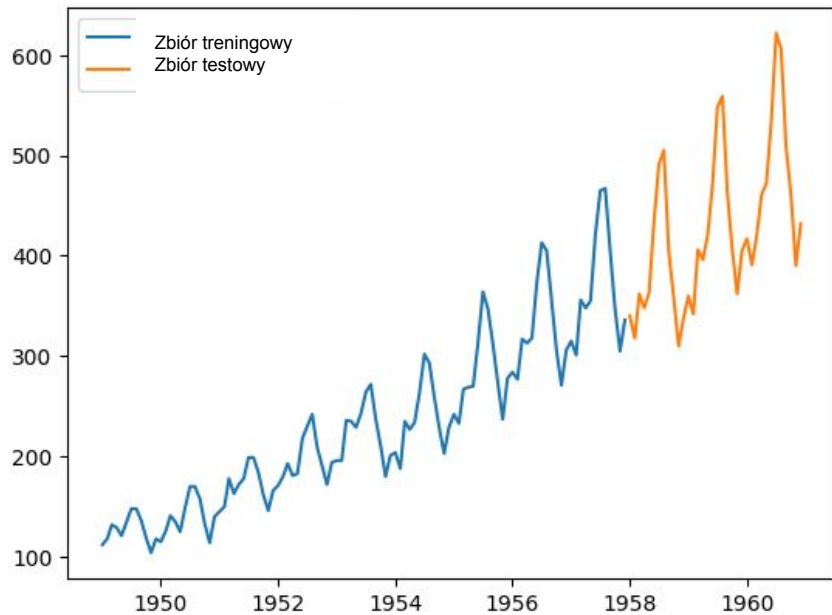
Przykład zbioru danych

- Zbiór danych zwany "Airline Passengers", który jest szeregiem czasowym reprezentującym miesięczną liczbę pasażerów przewoźnika lotniczego od stycznia 1949 do grudnia 1960 roku.
- Zbiór ten zawiera 144 obserwacje i jest często wykorzystywany do celów analizy i modelowania szeregów czasowych.



	year	month	passengers
0	1949	January	112
1	1949	February	118
2	1949	March	132
3	1949	April	129
4	1949	May	121
5	1949	June	135
6	1949	July	148
7	1949	August	148
8	1949	September	136
9	1949	October	119

Prognoza lotów wynik:



Linki:

- https://e2eml.school/convolution_one_d.html
- https://en.wikipedia.org/wiki/Time_series
- <https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1>
-