



Wydział Informatyki Politechniki Białostockiej
Studenckie Koło Naukowe Math4You
Koło Matematyków Studentów UJ

Dysonanse nie tylko poznawcze

Muzyka generatywna

Jan Gromko

Wrocław, 12 IV 2019



Fakt 1:
Wytrenowanie modelu
rozpoznającego gatunki muzyczne
lub kompozytorów jest możliwe.

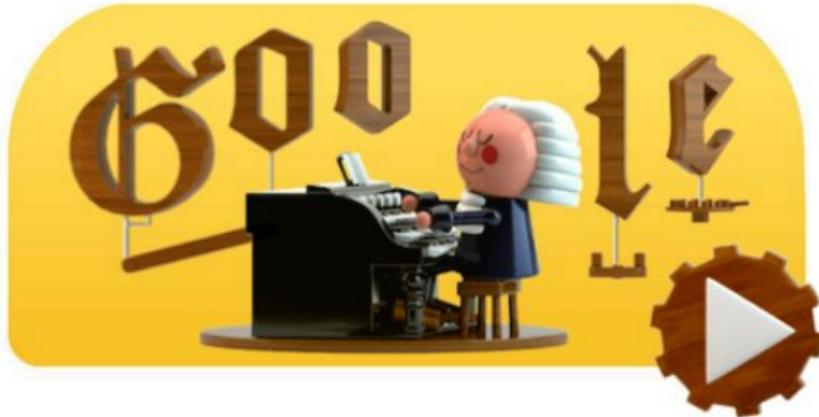


Fakt 2:
Wytrenowanie takiego modelu
nie jest trudne.



Pytanie 1:
Jak trudne jest w takim razie
wytrenowanie modelu, który będzie
twórcą sztuki?





Problem 1: Jaka muzyka się nam podoba?











Pytanie 2: Jak taki model stworzyć?



Prosta koncepcja



12

- ▶ Losowe generowanie dźwięków,
- ▶ dopuszczalny zbiór czystych akordów,
- ▶ z góry zadany rytm.

Prosta koncepcja



- ▶ Losowe generowanie dźwięków,
- ▶ dopuszczalny zbiór czystych akordów,
- ▶ z góry zadany rytm.

Muzyka bez struktury melodycznej, ograniczone możliwości „kreatywnego” generowania utworów.

Problem 1: Jaka muzyka się nam podoba?



Spokojna, z ładną, prostą melodią,
z czystymi akordami?



Spokojna, z ładną, prostą melodią,
z czystymi akordami?



Spokojna, z ładną, prostą melodią,
z czystymi akordami?



Prosta koncepcja



- ▶ Losowe generowanie dźwięków,
- ▶ dopuszczalny zbiór czystych akordów,
- ▶ z góry zadany rytm.

Muzyka bez struktury melodycznej, ograniczone możliwości „kreatywnego” generowania utworów.

Potrzebujemy bardziej zaawansowanego modelu.



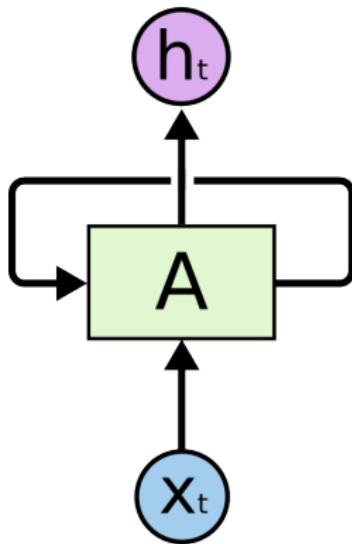
Rekurencyjne sieci neuronowe



Rekurencyjne sieci neuronowe



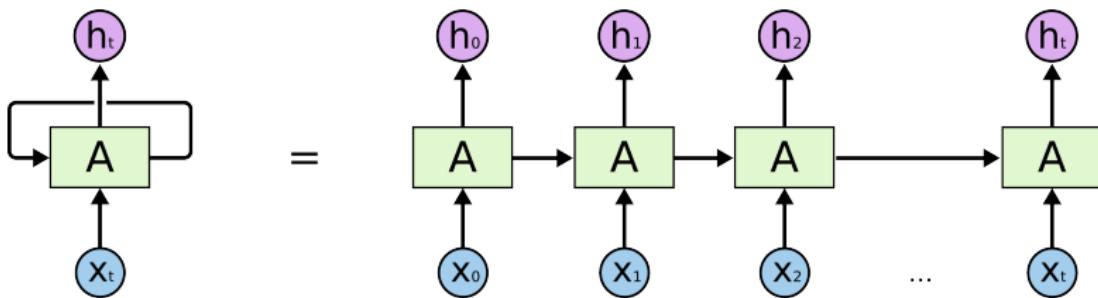
21



Rekurencyjne sieci neuronowe



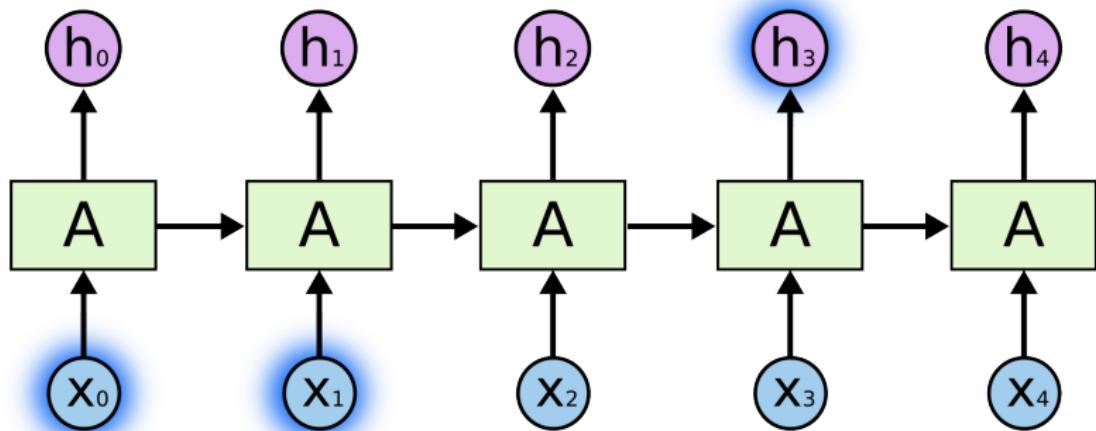
22



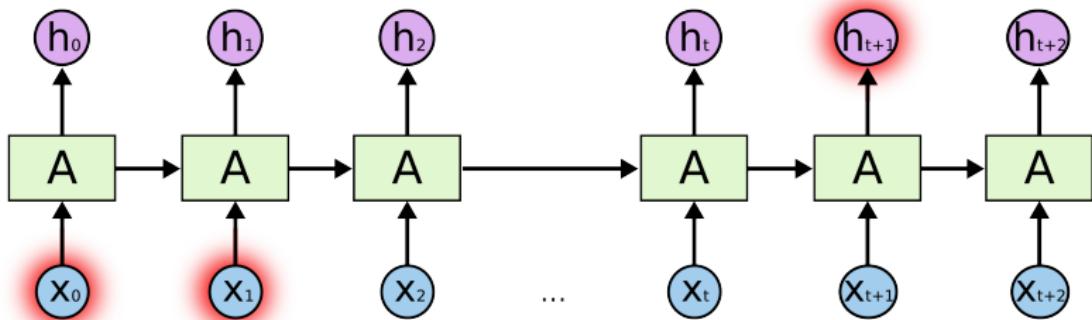
Rekurencyjne sieci neuronowe



23



Rekurencyjne sieci neuronowe



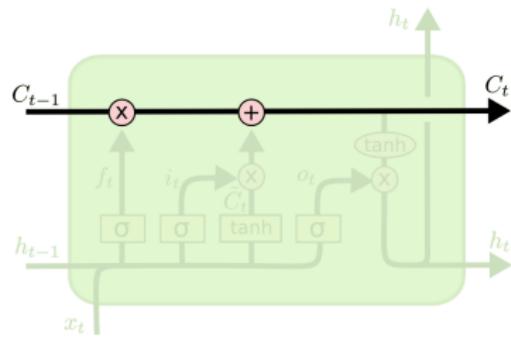
Sieci LSTM



Rekurencyjne sieci neuronowe



26



Sieci LSTM



- ▶ Bramka zapominająca – f_t ,
- ▶ bramka wejściowa – i_t ,
- ▶ bramka wyjściowa – o_t .

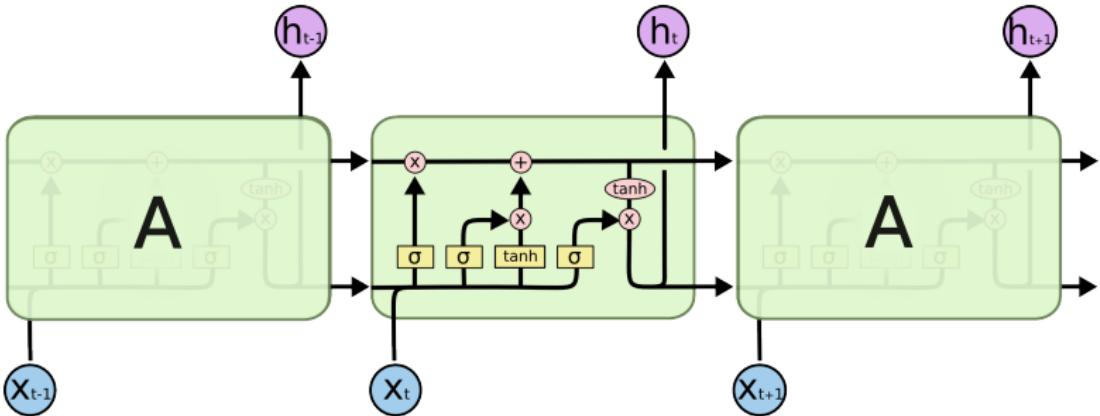
Sieci LSTM



28

- ▶ $W_{[x]}$ – macierze wag każdej z warstw komórki LSTM,
- ▶ $b_{[x]}$ – człony obciążenia każdej z warstw.

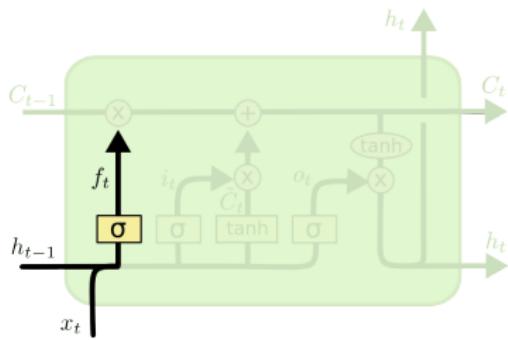
Rekurencyjne sieci neuronowe



Rekurencyjne sieci neuronowe

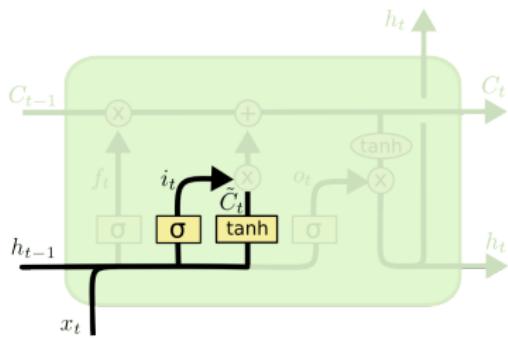


30



$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Rekurencyjne sieci neuronowe



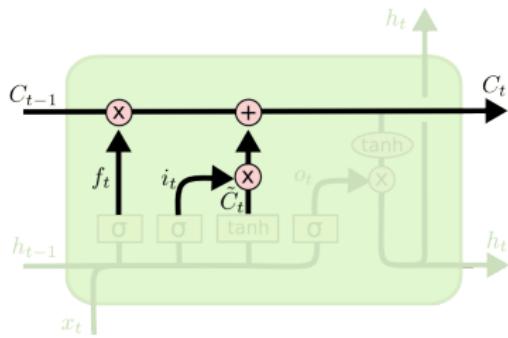
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

Rekurencyjne sieci neuronowe



32

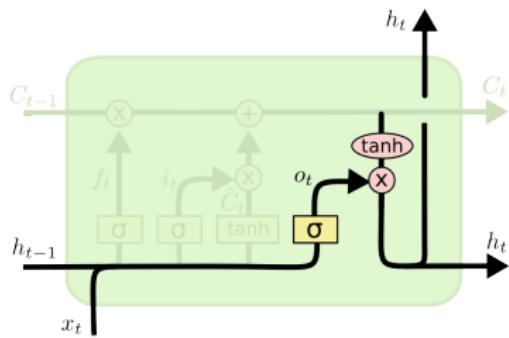


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Rekurencyjne sieci neuronowe



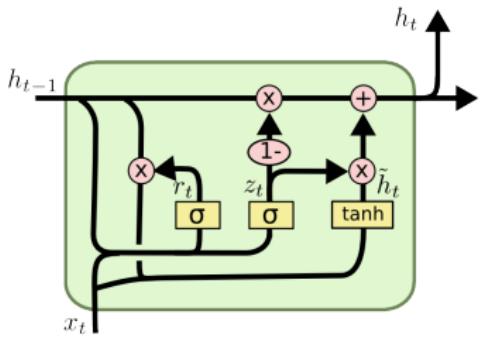
33



$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

Rekurencyjne sieci neuronowe



$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Model

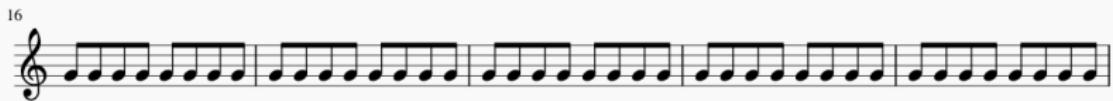
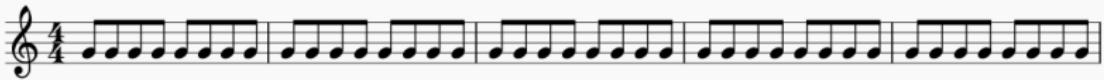


- ▶ Wejście: pliki MIDI,
- ▶ parsowanie ścieżek (music21),
- ▶ utworzenie słownika nut i akordów,
- ▶ uczenie sieci LSTM (bez wartości rytmicznych),
- ▶ generowanie dźwięków i zapis do MIDI.

No i mamy pierwszy wynik!

Wybitne dzieło muzyki współczesnej,
bez kompromisów na polu formy i treści.





Co poszło nie tak?



38

1. Mały zbiór uczący,
2. zbyt krótkie uczenie sieci.

Co poszło nie tak?



36

1. Mały zbiór uczący,
2. zbyt krótkie uczenie sieci.

Ale:

1. jedna iteracja na większym zbiorze zajmuje półtorej godziny,
2. potrzebujemy około 150-200 iteracji.

Weźmy GPU!



Czas uczenia spadł z 10 dni
do jednego, super!



Nie.



colab

+ TPU

Czas uczenia spadł
do 20 sekund na epokę.



Teraz super.



Jak teraz z wynikami?



Co można ulepszyć?



Co można ulepszyć?



48

- ▶ Uczenie sieci razem z wartościami rytmicznymi,

Co można ulepszyć?



49

- ▶ Uczenie sieci razem z wartościami rytmicznymi,
- ▶ oddzielna sieć do generowania rytmu,

Co można ulepszyć?



50

- ▶ Uczenie sieci razem z wartościami rytmicznymi,
- ▶ oddzielna sieć do generowania rytmu,
- ▶ użycie algorytmu do generowania rytmu,

Co można ulepszyć?



51

- ▶ Uczenie sieci razem z wartościami rytmicznymi,
- ▶ oddzielna sieć do generowania rytmu,
- ▶ użycie algorytmu do generowania rytmu,
- ▶ częściowo losowe generowanie rytmu,

Co można ulepszyć?



52

- ▶ Uczenie sieci razem z wartościami rytmicznymi,
- ▶ oddzielna sieć do generowania rytmu,
- ▶ użycie algorytmu do generowania rytmu,
- ▶ częściowo losowe generowanie rytmu,
- ▶ zmiana podejścia do kodowania nut.

Jak inaczej kodować ścieżkę dźwiękową?



53

- ▶ Melodia składa się z interwałów, a nie konkretnych dźwięków,
- ▶ możemy kodować różnice pomiędzy kolejnymi akordami, a nie akordy,
- ▶ pozostaje problem rytmu.

Jak kodować zmianę?



$$D \rightarrow [FAC] \rightarrow [EG]$$

Jak kodować zmianę?



55

$$D \rightarrow [FAC] \rightarrow [EG]$$

$$[3, 7, -2] \rightarrow [-1, 2]$$

Algorytmiczne generowanie rytmu



56

Problem

Mając dane n i k , gdzie n jest liczbą jednostek metrycznych w sekwencji, a k liczbą nut, które mają być rozmiieszczone w sekwencji, znaleźć taki wektor binarny o długości n , w którym k jedynek będzie rozmiieszczonych możliwie rzadko między zerami.

Algorytmiczne generowanie rytmu



57

Przykład

Dla $n = 9$ i $k = 3$:

[1 0 0 1 0 0 1 0 0]

Algorytmiczne generowanie rytmu



58

Przykład 2

Dla $n = 13$ i $k = 5$:

[1 1 1 1 1 0 0 0 0 0 0 0]

Algorytmiczne generowanie rytmu



59

Przykład 2

Dla $n = 13$ i $k = 5$:

[1 1 1 1 1 0 0 0 0 0 0 0]
[1 0][1 0][1 0][1 0][1 0][0][0][0]

Algorytmiczne generowanie rytmu



Przykład 2

Dla $n = 13$ i $k = 5$:

[1 1 1 1 1 0 0 0 0 0 0 0 0]
[1 0][1 0][1 0][1 0][1 0][0][0][0]
[1 0 0][1 0 0][1 0 0][1 0][1 0]

Algorytmiczne generowanie rytmu



Przykład 2

Dla $n = 13$ i $k = 5$:

[1 1 1 1 1 0 0 0 0 0 0 0 0]
[1 0][1 0][1 0][1 0][1 0][0][0][0]
[1 0 0][1 0 0][1 0 0][1 0][1 0]
[1 0 0 1 0][1 0 0 1 0][1 0 0]

Algorytmiczne generowanie rytmu



62

Przykład 2

Dla $n = 13$ i $k = 5$:

[1 1 1 1 1 0 0 0 0 0 0 0 0]
[1 0][1 0][1 0][1 0][1 0][0][0][0]
[1 0 0][1 0 0][1 0 0][1 0][1 0]
[1 0 0 1 0][1 0 0 1 0][1 0 0]

[1 0 0 1 0 1 0 0 1 0 1 0 0]

Algorytmiczne generowanie rytmu



63

Algorytm Euklidesa

Dla $m = 8$ i $k = 5$:

$$\begin{aligned}euklides(8, 5) &= euklides(5, 3) = euklides(3, 2) = euklides(2, 1) = \\&= euklides(1, 0) = 1\end{aligned}$$

m, k – liczba zer i jedynek

$$m = n - k$$

Losowe generowanie rytmu



1. Losowanie długości sekwencji,
2. losowanie offsetów,
3. losowanie czasu trwania dźwięków,
4. losowanie liczby powtórzeń sekwencji rytmicznej.

Jak to brzmi?





SPRAWDŹMY TO!

Potrzebujemy silnego czynnika
losowego na wielu poziomach.



Co dalej?



68

- ▶ Bardziej zaawansowane generowanie rytmu (np. w oparciu o losowo parametryzowany algorytm Euklidesa),
- ▶ powiązanie interwałów z rytmem,
- ▶ wprowadzenie (meta)parametrów i uczenie w oparciu o teorię muzyki.

Interpretacja ma znaczenie.



Interpretacja ma znaczenie.



Interpretacja ma znaczenie.



Bibliografia



- [1] A. Geron
Uczenie maszynowe z użyciem Scikit-Learn i TensorFlow
- [2] C. Olah
Understanding LSTM Networks, <http://colah.github.io/>
- [3] S. Merity, N. S. Keskar, R. Socher
Regularizing and Optimizing LSTM Language Models
- [4] G. Toussaint
The Euclidean Algorithm Generates Traditional Musical Rhythms
- [5] S. Skúli
How to Generate Music using a LSTM Neural Network in Keras,
<https://towardsdatascience.com/>

Kod źródłowy, wraz z wygenerowaną muzyką
i wytrenowanymi modelami, można znaleźć pod adresem:

<https://github.com/jangromko/omatko2019>

