

**LSTM w TensorFlow dla *Pana Tadeusza***  
**Adama Mickiewicza**  
(zadanie przygotowane wraz z Michałem Zającem)

Zbuduj i wytrenuj model rekursywnej sieci neuronowej zgadujący kolejne znaki w tekście *Pana Tadeusza*. Wymagane perplexity na zbiorze testowym to **4,99**.

Scalony tekst [10 książ](#) jest naszym zbiorem treningowym (ponad 370k znaków), [11 księga](#) jest zbiorem walidacyjnym, a [12 księga](#) zbiorem testowym.

Po każdej epoce trenowania powinna zostać wygenerowana próbka tekstu z modelu zainicjalizowanego kontekstem: *Jam jest Jacek*. Wymagana jest również wizualizacja działania modelu jak w poprzednich projektach.

Termin: **11 grudnia** za **1,5 punktu**

Termin: 18 grudnia za 0,75 punktu

Dodatkowo ogłaszamy konkurs na jak najniższe perplexity. Termin rozstrzygnięcia konkursu to **13 grudnia** godz. **23:59**.

Poniżej podajemy listę wymagań dla projektowanego modelu:

- komórka może mieć dowolną architekturę ale polecana jest zwykła LSTM
- wielowarstwowa architektura, gdzie liczba warstw jest hiperparametrem modelu
- obowiązują *shared embeddings* (*współdzielone zanurzenia*), a zatem macierz przetwarzająca one-hot wektory alfabetu na gęstą reprezentację powinna być transpozycją macierzy tłumaczącej wyjście modelu na logitsy
- dropout z ustaloną maską dla jednego zejścia gradientowego (poziomy wewnątrz warstwy jak i pionowy między warstwami oraz po *input embedding* i przed *output embedding*)
- Weight decay, czyli regularyzacja L2 wag modelu
- Activation Regularization oraz Temporal Activation Regularization (<https://arxiv.org/abs/1708.02182>)
- EMA
- losowa długość okienka czasowego dla pojedynczego zejścia gradientowego
- dynamiczna ewaluacja
- w tym projekcie NIE przeprowadzamy głosowania zewnętrznego

**Uwaga: Wszystkie powyższe elementy powinny być zaimplementowane, natomiast niekoniecznie muszą być włączone w ostatecznej wersji modelu!**

A tutaj podajemy pomysły, które mogą pomóc wygrać konkurs:

- DropConnect, czyli dropout na wagach modelu (<https://arxiv.org/abs/1708.02182>)
- ZoneOut (<https://arxiv.org/abs/1606.01305>)
- learning rate schedule dla ADAMa (mimo wszystko)

- Bayesowski model