Projekt (40pkt) - NLP, semestr zimowy 2018/2019

(wersja ostateczna zadania pojawi się do 23 grudnia 2018r.)

Termin oddawania projektu: 18 stycznia 2019r.

Link do zbioru danych i embeddingów znajduje się tutaj.

Zaimplementuj sieć neuronową, która rozwiązuje zadanie Natural Language Inference na podzbiorze Stanford Natural Language Inference (SNLI). Zadanie polega na klasyfikacji pary zdań (przesłanka, hipoteza) do jednej z 3 kategorii („entailment" - z pierwszego zdania wynika drugie, „contradiction" - drugie zdanie przeczy pierwszemu, „neutral" - zdania są niezależne).

Zadania:

- (12pkt) Preprocessing (Zarówno podzbiór SNLI jak i embeddingi znajdują się na google drive - link powyżej):
  - Ze zbiorów SNLI wybierz tylko te zdania, w których większość adnotatorów wybrała jedną klasę (czyli gold_label ≠ „-"). Wypisz 10 zdań wyrzuconych.
  - Stokenizuj zdania i zamień wszystkie litery na małe (ponieważ embeddingi słów zakładają, że to zrobiliśmy). Wypisz liczbę różnych słów, które zostały.
  - Jako enkodera słów użyj embeddingów wytrenowanych metodą GloVe na zbiorze Wikipedia 2014 (6B tokenów, 400k vocabulary, 200D) (ściągamy te embeddingi, uczenie ich zajęłoby zbyt dużo czasu). Do macierzy embeddingów wybierz tylko te słowa, które występują jednocześnie w GloVe i SNLI.
  - Słowa, które występują w SNLI, ale nie znajdują się w GloVe, traktuj jako <unk>, dla którego wektor zainicjalizuj losowo. Podobnie stwórz dwa dodatkowe wektory, <bos>, <eos>, które reprezentują początek i koniec zdania (zmodyfikuj zbiór SNLI w taki sposób, żeby każde zdanie zaczynało się od tokenu <bos>, kończyło tokenem <eos>, a dla nieznanych słów zawierało token <unk>)
- (15pkt) Zdefiniuj model i wytrenuj go (monitorując co epokę accuracy na zbiorze treningowym oraz testowym):
  - Słowa powinny być enkodowane za pomocą enkodera słów (używamy word embeddingów jak powyżej, nie uczymy)
  - Zdania powinny być zaenkodowane za pomocą enkodera zdań (LSTM, który jest uczony podczas treningu - można użyć jednego LSTM, aby zaenkodować przesłankę i hipotezę, osobno) - używamy hidden size równy 100.
  - W wyniku powyższego enkodowania dostaniemy 2 wektory, które są reprezentacjami zdań (dla przesłanki, dla hipotezy). Te wektory konkatenujemy i przepuszczamy przez dwie warstwy sieci neuronowej (pierwsza z hidden size 100 i funkcją aktywacji relu, następnie warstwa softmaxowa)
  - Używamy batchów (przykładowo batch size 32)
  - Proszę dodać wykres zmian lossów i accuracy per epoch oraz wypisać po 3 przykłady dobrze i źle zaklafyfikowane (1 per klasa)
- (8pkt) Reprodukowalność:
  - Napisz funkcję, która zapisuje wagi najlepszego modelu po każdej epoce treningu.
  - Napisz funkcję, która ładuje model z zapisanych wag, i funkcję, która testuje model na zbiorze testowym.
- (5pkt) Wynik na zbiorze testowym (to be defined)

W pierwszej linijce rozwiązania proszę o oświadczenie, że zadanie zostało wykonane samodzielnie.