

Практическое задание №4 8382

Черницын Павел

Вариант 6

Задание

Необходимо реализовать нейронную сеть вычисляющую результат заданной логической операции. Затем реализовать функции, которые будут симулировать работу построенной модели. Функции должны принимать тензор входных данных и список весов. Должно быть реализовано 2 функции:

1. Функция, в которой все операции реализованы как поэлементные операции над тензорами
2. Функция, в которой все операции реализованы с использованием операций над тензорами из NumPy

Для проверки корректности работы функций необходимо:

1. Инициализировать модель и получить из нее веса ([Как получить веса слоя](#), [Как получить список слоев модели](#))
2. Прогнать датасет через не обученную модель и реализованные 2 функции. Сравнить результат.
3. Обучить модель и получить веса после обучения
4. Прогнать датасет через обученную модель и реализованные 2 функции. Сравнить результат.

Примечание: так как множество всех наблюдений ограничен, то обучение проводить можно на всем датасете без контроля.

(a and not b) or (c xor b)

Выполнение

Была реализована нейросеть вычисляющая результат заданной логической операции.

```
model = Sequential()
model.add(Dense(10, activation='relu', input_shape=(3,)))
model.add(Dense(10, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
```

Затем были написаны вспомогательные функции:

def naive_matrix_matrix_dot(x, y): - скалярное произведение матриц

def naive_vector_dot(x, y): - скалярное произведение векторов

def naive_relu(x): - функция relu для x

def sigmoid(x): - функция sigmoid для x

Затем были написаны функции, симулирующие работу построенной модели.

`def naive_simulation(layers, input):` - поэлементные операции над тензорами

`def np_simulation(layers, input):` - обработка тензоров через NumPy

Датасет был прогнан через не обученную модель:

```
Numpy untrained simulation:
[[0.5
  [0.48522631]
  [0.5227453 ]
  [0.54527707]
  [0.41769233]
  [0.40842322]
  [0.38995879]
  [0.43819118]]
Naive untrained simulation:
[[0.5
  [0.48522631]
  [0.5227453 ]
  [0.54527707]
  [0.41769233]
  [0.40842322]
  [0.38995879]
  [0.43819118]]
```

Датасет был прогнан через обученную модель:

```
Untrained model predict:
[[0.5
  [0.4852263 ]
  [0.5227453 ]
  [0.54527706]
  [0.41769233]
  [0.40842322]
  [0.3899588 ]
  [0.43819118]]
Numpy trained simulation:
[[0.22180459]
 [0.88889871]
 [0.96059947]
 [0.01790155]
 [0.97137948]
 [0.99154577]
 [0.91817433]
 [0.11276371]]
Naive untrained simulation:
[[0.22180459]
 [0.88889871]
 [0.96059947]
 [0.01790155]
 [0.97137948]
 [0.99154577]
 [0.91817433]
 [0.11276371]]
Trained model predict:
[[0.22180459]
 [0.88889873]
 [0.9605995 ]
 [0.01790148]
 [0.9713795 ]
 [0.9915458 ]
 [0.9181743 ]
 [0.11276367]]
```

Значения близки к верным.