

Cel laboratorium

Celem laboratorium było zapoznanie się z funkcjonalnością pakietów TensorFlow i Keras, implementacja prostego perceptronu wielowarstwowego. Następnie wykonanie eksperymentów badających jakość uczenia oraz klasyfikacji dla różnych ustawień.

Wyniki

Wpływ funkcji aktywacji na jakość uczenia

Eksperyment z przetestowaniem funkcji aktywacji został wykonany z ustawieniami podobnymi do tych przedstawionych w podstawowym tutorialu TensorFlow.

Funkcja aktywacji	Jakość uczenia	Jakość klasyfikacji
sigmoid	0.8676	0.9129
hard_sigmoid	0.4420	0.4788
tanh	0.1071	0.1768
linear	0.0943	0.0893
relu	0.1480	0.0980
softmax	0.4760	0.8462

Najwyższą jakość uczenia miała funkcja aktywacji 'sigmoid', która jest powszechnie wykorzystywana przy implementacji prostszych modeli, takich jak MlpExtreme.

Jednak należy pamiętać, że dobór funkcji aktywacji jest zależny od problemu.

Najgorzej poradziła sobie funkcja aktywacji 'linear'.

Funkcje 'softmax' oraz 'hard_sigmoid' poradziły sobie umiarkowanie, jednak dla funkcji 'softmax' jakość klasyfikacji jest znacznie większa od 'hard_sigmoid'.

Wpływ liczby epok na jakość klasyfikacji

Eksperyment z liczbą epok został przetestowany dla ustawień z poprzedniego eksperymentu, jednak funkcja aktywacji została ustawiona na 'sigmoid' (ponieważ miała ona najwyższą jakość uczenia).

Liczba epok	Jakość uczenia	Jakość klasyfikacji
10	0.9141	0.9244
100	0.9157	0.9267
1000	0.9146	0.9246

Najwyższa jakość klasyfikacji jest dla 100 epok uczenia. Oznacza to, że nie zawsze większa liczba epok uczenia przekłada się na lepszą jakość uczenia oraz klasyfikacji. W trakcie uczenia można było zaobserwować wzrastającą jakość klasyfikacji wraz z kolejnymi epokami, jednak po kilku epokach jakość klasyfikacji przestała wzrastać znacząco.

Wpływ optymalizatora na jakość uczenia

Eksperyment z liczbą epok został przetestowany dla ustawień z poprzedniego eksperymentu, jednak liczba epok została ustawiona na wartość równą 10, aby uniknąć dużego czasu oczekiwania na wyniki.

Optymalizator	Jakość uczenia	Jakość klasyfikacji
adam	0.9142	0.9251
sgd	0.8995	0.9167
adadelata	0.3831	0.4611
adagrad	0.8433	0.8662
rmsprop	0.9150	0.9264

Najlepszym optymalizatorem dla tego problemu okazał się optymalizator 'rmsprop'.

Wpływ kroku na jakość uczenia

Eksperyment z liczbą epok został przetestowany dla ustawień z poprzedniego eksperymentu, jednak optymalizator został ustawiony na 'rmsprop', ze względu na największą jakość uczenia oraz klasyfikacji w poprzednim eksperymencie.

Krok uczenia	Jakość uczenia	Jakość klasyfikacji
0.0001	0.9011	0.9157
0.001	0.9144	0.9267
0.005	0.9072	0.9239
0.01	0.8987	0.9132
0.05	0.8717	0.9051
0.1	0.8691	0.9096
0.2	0.8648	0.8831
0.3	0.8655	0.8931
0.4	0.8629	0.8966
0.5	0.8608	0.9010

0.6	0.8619	0.9013
0.7	0.8616	0.8816
0.8	0.8623	0.8999

Najwyższa jakość uczenia była dla kroku uczenia równego 0.001. Oznacza to, że należy rozsądnie dobrać krok uczenia w zależności od algorytmu i problemu, aby model nie stracił na jakości uczenia. Nie powinien on być zbyt mały ani zbyt duży. Można zaobserwować, że wraz ze wzrostem kroku uczenia zmniejsza się jakość uczenia.

Predykcja na przykładowym obrazie testowym

Do predykcji stworzono model z funkcją aktywacji 'sigmoid' oraz optymalizatorem 'adam', ze względu na ich najwyższe dokładności uczenia oraz klasyfikacji wykazane w poprzednich eksperymentach.

Po wykonaniu uczenia na danych uczących wykorzystywanych w poprzednich eksperymentach z 10 epokami uczenia, dokładność klasyfikacji wyniosła 0%, Obrazek nie został dobrze rozpoznany.

Podobieństwo figury na obrazie do liczby 1 oraz 7 jest dość spore, szczególnie przy takim obrocie testowego obrazu.

Wnioski

Im lepsza jest jakość danych, tym lepiej dopasuje się model. Dobór odpowiednich parametrów do danego modelu zależy od postawionego problemu oraz specyfiki przekazywanych mu danych. W przypadku danych dobrej jakości, zmiana większości parametrów nieznacznie wpływa na poprawę jakości uczenia oraz klasyfikacji.