

## Rozpoznawanie cyfr na obrazach z użyciem zbioru danych MNIST

1. Na samym początku zapoznaję się z dokumentacją TensorFlow.
2. Następnie ładuję bibliotekę TensorFlow oraz zbiór MNIST zawierający cyfry.  
Dzielę cały zbiór cyfr na zbiór uczący i testowy.
3. Normalizuję w każdym obrazie dane tak aby wartości mieściły się w zakresie  $[0,1]$ .
4. Dzięki bibliotece TensorFlow tworzę model sieci Sekwencyjnej gdzie:
  - Pierwsza warstwa "flatten" przemienia obraz na jednowymiarowy wektor o wielkości 784.
  - Druga warstwa to warstwa "dense", co oznacza, że każdy neuron w tej warstwie łączy się z każdym neuronem w poprzedniej warstwie. Zastosowana funkcja aktywacji "relu".
  - Trzecia warstwa losowo wyłącza 20% neuronów co zapobiega przeuczeniu modelu.
  - Czwarta warstwa ponownie jest warstwą "dense" jednakże mamy tylko 10 neuronów, ponieważ mamy cyfry od 0 do 9. Zastosowana funkcja "softmax" wylicza prawdopodobieństwo danej cyfry.
5. Konfiguracja modelu uczeniowego. Ustawiam optimizer na "adam", który przez dokumentację uznawany jest za szybki i efektywny algorytm uczący, a loss na "sparse\_categorical\_crossentropy".

Sparse\_categorical\_crossentropy mierzy stratę między przewidywanymi prawdopodobieństwami (wyjściem modelu), a rzeczywistymi etykietami klas. Strata wynosi zero, jeśli model jest pewien, że klasa jest prawidłowa.

Ustawiam wyjście jako "accuracy" która wyświetla procentowy wynik skuteczności modelu.

6. Ładuję zbiór uczący i testowy. Ustawiam ilość epok na 5 i startuję model.  
Skuteczność modelu jest na poziomie ponad 97%.