Lab 06 - Sieci konwolucyjne

Convolutional neural networks. Temat był omawiany na wykładzie. Komentarz dotyczący wydajności - warstwa konwolucyjna będzie miała weights=filter\_height×filter\_width×input\_channels×output\_filters+output\_filters wag do nauczenia, więc tego jest całkiem sporo, chociaż mniej niżbyśmy mieli sieć w pełni połączoną.

Dana jest sieć do rozpoznawania cyfr pisanych. Dane do uczenia bierzemy z zestawu MNIST, który jest publicznie dostępny. Przykład wykorzystujący właśnie ten zestaw do uczenia jest dostępny na stronie dokumentacji <https://www.tensorflow.org/datasets/keras_example>

Przykład z dokumentacji ma problem z zbiorem danych. Poprawka to

pip3 uninstall tensorflow\_datasets; pip3 install tensorflow\_datasets==4.9.2

**Zadanie wstępne**

Uruchom lokalnie wspomniany przykład. Powinno uczyć sieć warstwową. Zapoznaj się z tym jak ona jest skonstruowana i jeśli coś jest niejasne, to zachęcam do zadawania pytań. Nie kasuj tego rozwiązania, przyda się później.

**Zadanie 1**

Przerób powyższy przykład w taki sposób, aby zamiast zwykłych warstw w pełni połączonych były także warstwy konwolucyjne. Szablon rozwiązania powinien zostać tak jak z zadania wstępnego.

Wykorzystamy warstwy Conv2D oraz MaxPooling2D. Temat jest dość typowy i bardzo szeroko omawiany w literaturze + tutorialach, na przykład <https://www.tensorflow.org/tutorials/images/cnn> (i oczywiście wykład). Proszę zwrócić uwagę na kształty (shape) wejścia i wyjścia z sieci. Zachęcam także do wywoływania co jakiś czas funkcji

*print(model.summary())*  
Pozwoli to na wygodną analizę tego co się stworzyło.

**Zadanie 2**

Jak do tej pory, nasze przykłady najczęściej nie stanowią gotowego rozwiązania, które można do czegoś użyć. Zmieńmy ten stan. Niech nasza sieć będzie potrafiła pobrać obrazek z pliku i odpowie jedną wartością - cyfrą która jest widoczna na rysunku. Podpowiem:

*tf.keras.utils.load\_img*

**Zadanie 3**

Porównaj (na ile to możliwe w ograniczonym czasie zajęć) jak sprawują się sieci warstwowe kontra konwolucyjne dla tego samego zadania i takiej samej liczby iteracji. Wystarczy porównanie wyników dla danych testowych.