

Zespół 5: Michał Motak, Kamil Hosaja, Maciej Wowra

W ramach dotychczasowych prac nad projektem zaimplementowano konwolucyjną sieć neuronową UNet, na podstawie oryginalnej pracy przedstawiającej tę architekturę [1]. Implementacja wykonana została w oparciu o bibliotekę **Keras** w języku Python (kod umieszczony został w serwisie GitHub pod nazwą UNET.py). Architektura zaimplementowanej sieci jest zgodna z wyżej cytowaną pracą.

Dane (obrazy oraz maski) otrzymane na podstawie kodu organizatorów challenge (kod z folderu *starter_code* → *load_case()*, obraz typu **NiftyImage** został przekształcony na numpy array za pomocą funkcji *get_data()* z biblioteki **Nibabel**) zostały podzielone na przekroje i zapisane funkcją *imsave()* z biblioteki **matplotlib** na dysku, w celu późniejszego dostępu, w formacie .png (nie zapisywano wszystkich dostarczonych przykładów).

W celu wczytania danych do modelu wykorzystano klasę *ImageDataGenerator()* oraz metodę *flow_from_directory()* z biblioteki **Keras**, która pozwala na pobieranie ich bezpośrednio z folderu, skalowanie do poszukiwanego rozmiaru, przestawianie kolejności danych wysyłanych w porcjach uprzednio podzielonych przez metodę, w zależności od ziarna. Maski zostały przekształcone do postaci *one-hot encoding*, za pomocą własnoręcznie zaimplementowanego kodu.

Podczas trenowania modelu wykorzystywany był optymalizator Adam w podstawowej wersji (learning rate = 0.001) oraz wartości *steps_per_epoch* = (ilość przykładów treningowych // BATCH_SIZE) i *validation_steps* = (ilość przykładów testowych // BATCH_SIZE). Trening modelu wykonywany był z wykorzystaniem GPU zapewnianego przez Google Colaboratory.

Walidacja zaproponowanej metody polegała na porównaniu otrzymanych masek (R) z maskami wzorcowymi (G). Wybrano trzy metryki i na etapie prototypu zaimplementowano pierwszą z nich.

- 1) Dice coefficient (DICE) - oznacza podobieństwo masek (?)

$$DICE(R, G) = \frac{2|R \cap G|}{|R| + |G|}$$

- 2) volumetric overlap error (VOE) - stosunek części wspólnej do sumy rezultatu i maski wyrażona w procentach. Wartość 0 oznacza idealną segmentację.

$$VOE(R, G) = (1 - \frac{|R \cap G|}{|R \cup G|}) \cdot 100\%$$

- 3) relative volume difference (VD) - wskazuje na podobieństwo objętości masek. Nie uwzględnia ich odwzorowania, tylko objętość. Pokazuje czy metoda ma tendencję do przesegmentowania (wartość dodatnia) albo podsegmentowania (wartość ujemna).

$$VD(R, G) = \frac{|R| - |G|}{|G|} \cdot 100 \%$$

Model wytrenowany z zaimplementowaną metryką DICE oraz z funkcją kosztu DICE_loss nie dawał zadowalających wyników - przewidywane maski zawierały tylko jedną wartość (niepoprawnie zaimplementowana obecnie funkcja DICE()). Model wytrenowany z funkcją kosztu 'categorical_crossentropy' dawał lepsze, choć dalej złe wyniki (były przewidywane jakieś pola jednak nie należały one ani do nerek ani do guza).

[1] O. Ronneberger, P. Fischer, i T. Brox, „U-Net: Convolutional Networks for Biomedical Image Segmentation”, *arXiv:1505.04597 [cs]*, maj 2015, Udostępniono: cze. 07, 2020.

[Online]. Dostępne na: <http://arxiv.org/abs/1505.04597>.

[2] Q. Dou i in., „3D deeply supervised network for automated segmentation of volumetric medical images”, *Medical Image Analysis*, t. 41, s. 40–54, paź. 2017, doi:

[10.1016/j.media.2017.05.001](https://doi.org/10.1016/j.media.2017.05.001).