

Grupa: Kula Michał (111105)

Wybrany temat: 4A – Contrastive learning – instrukcja Jupyter Notebook

Link do repozytorium: https://github.com/DJKulix/AIPO_Projekt4A

W repozytorium znajdują się dwa pliki:

- *szablon.ipynb* – „surowy” szablon z blokami kodu do uzupełnienia przez studenta
- *przykład.ipynb* – notebook z uzupełnionymi blokami

Wstęp

Celem zadania jest stworzenie notebooka Jupyter, który w trakcie jednych zajęć laboratoryjnych (1.5h) wyjaśni ideę contrastive learningu, oraz przybliży jak stworzyć taki model i wykorzystać go do klasyfikacji przykładowych danych, testując różne modyfikacje parametrów.

W trakcie ćwiczenia student:

- zapozna się z ideą uczenia kontrastycznego,
- zaimplementuje model encoder wraz z projection head,
- wykorzysta funkcję straty NT-Xent,
- przeprowadzi proces treningu modelu.

Notebook jest podzielony na sekcje odpowiadające kolejnym etapom tworzenia modelu. Końcowym efektem jest stworzenie przykładowego modelu, w którym student może w łatwy sposób sprawdzić wpływ parametrów i transformacji na jakość wyników. Dodatkowo w notebooku zawarte są bloki kodu odpowiadające za wizualizację kolejnych etapów pracy.

Zakres merytoryczny

- Wyjaśnienie idei contrastive learning, powodów użycia i zastosowań

<https://medium.com/biased-algorithms/contrastive-learning-explained-9343b3a67513>

<https://www.v7labs.com/blog/contrastive-learning-guide>

- Kontrastowanie par danych podobnych (positive) i różnych (negative)
- Augmentacja danych, wpływ jakości augmentacji na jakość reprezentacji

<https://proceedings.mlr.press/v119/chen20j/chen20j.pdf> - str. 4 – przykłady transformacji

- Contrastive loss

<https://towardsdatascience.com/nt-xent-normalized-temperature-scaled-cross-entropy-loss-explained-and-implemented-in-pytorch-cc081f69848/>

Dane

W notebooku wykorzystywany jest zbiór CIFAR-10 do treningu contrastive learning oraz ewaluacji linear probe. Zbiór zawiera 60 000 obrazów w 10 klasach (32×32 piksele). W notebooku jest także zadanie do samodzielnego wykonania polegające na wykorzystaniu innego zbioru danych np. STL-10.

Zadania do samodzielnego wykonania

- Uzupełnienie bloków kodu w oparciu o polecenia
- Modyfikacja augmentacji danych
- Eksperymenty z funkcją straty
- Eksperymenty treningowe

Środowisko i zależności

Notebook zakłada minimalne środowisko, które można zainstalować uruchamiając pierwsze dwa bloki kodu (do uruchomienia wymagany jest pakiet *pip*):

- Python
- PyTorch
- torchvision
- numpy
- matplotlib
- scikit-learn (t-SNE)

Działanie notebooka zostało przetestowane w Google Colab na ustawieniach runtime Python 3 w najnowszej wersji i T4 GPU dla obsługi CUDA oraz w IDE PyCharm w wersji 2025.1.3.1.

Plan notebooka (struktura zajęć)

1. Wprowadzenie i cele zajęć
2. Teoria contrastive learning
3. Definicja transformacji i ich wpływ na model
4. Podgląd augmentacji
5. Definicja modelu (encoder, MLP i normalizacja)
6. Przykład funkcji contrastive loss w oparciu o NT-Xent, wpływ parametru temperatury, trening modelu
7. Ewaluacja reprezentacji (t-SNE / linear probe), statystyki i wykresy
8. Zadania do samodzielnego wykonania

Podsumowanie

Średni czas wykonania notebooka powinien wynieść ok. 1.5h, w zależności od wydajności komputera i prędkości działania modelu treningowego. Po wykonaniu student powinien zrozumieć na czym polega uczenie kontrastywne i kiedy jest stosowane, a także powinien rozumieć rolę augmentacji i znaczenie wykorzystanych funkcji. Notebook może stanowić punkt wyjścia do dalszych eksperymentów np. przez zastosowanie innego zbioru albo modelu.

Bibliografia

<https://medium.com/biased-algorithms/contrastive-learning-explained-9343b3a67513>

<https://medium.com/biased-algorithms/contrastive-learning-explained-9343b3a67513>

<https://www.v7labs.com/blog/contrastive-learning-guide>

<https://proceedings.mlr.press/v119/chen20j/chen20j.pdf>

<https://towardsdatascience.com/nt-xent-normalized-temperature-scaled-cross-entropy-loss-explained-and-implemented-in-pytorch-cc081f69848/>