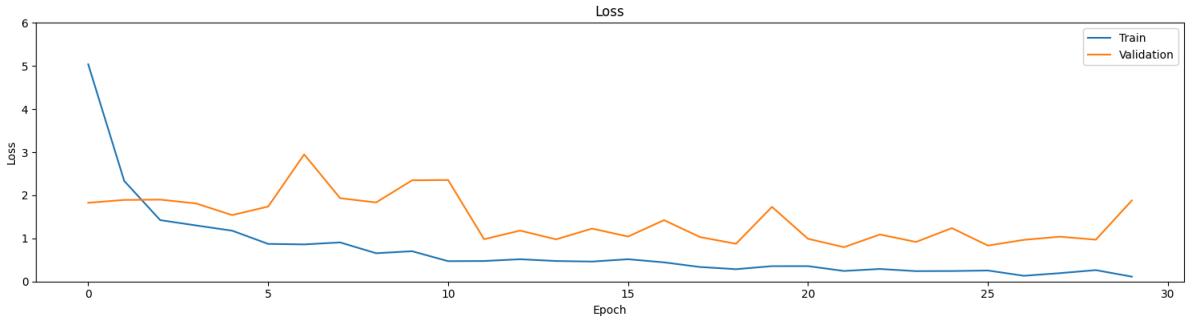
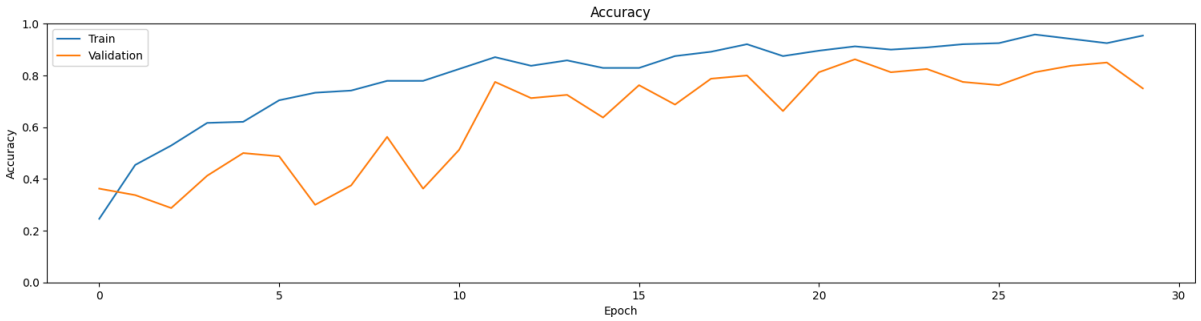
Rozpoznawanie scen w filmach - klasyfikacja scen w materiałach wideo

Aleksandra Talabska, Szymon Jasiński

1. Wyniki testowe i treningowe

Po 30 epokach treningowych model osiąga satysfakcjonujące wartości accuracy dla zbioru walidacyjnego. Funkcja loss zbioru walidacyjnego maleje.





2. Wybór modelu

Architektura modelu bazuje na warstwach konwolucyjnych 3D. Zastosowano osobne warstwy konwolucyjne dla wymiaru przestrzennego i czasowego. Funkcja aktywacji ReLU zapewnia nieliniowość i pomaga w ekstrakcji cech. Warstwa MaxPooling wydobywa dominującą cechę z feature map i zmniejsza wymiar przestrzenny. Dzięki BatchNormalization możliwe są sprawniejsze obliczenia. Po pierwszej sekwencji powyższych warstw stosuje się drugą z większą liczbą filtrów w warstwach konwolucyjnych. Następnie warstwa Flatten przekształca tensor 4D w wektor 1D. Kolejna mniejsza warstwa Dense agreguje cechy na wyższym poziomie. Problemy z overfittingiem doprowadziły do wykorzystania warstwy Dropout z wartością 0,5.

Learning rate wynoszący 0,001 pozwala na szybki trening modelu. Założono 50 epok treningowych, ale z wykorzystaniem EarlyStopping (patience 8) trenowanie zostało zakończone po 30 epokach.

3. Strategia podziału danych

Dane podzielono na 3 zbiory: treningowy, walidacyjny i testowy. Dla każdej klasy wybrano po 30, 10 i 10 plików wideo do zbiorów odpowiednio treningowego, walidacyjnego i testowego.

4. Opis danych wejściowych

Dataset UCF101 (action recognition) zawiera filmy zebrane z YouTube, które przedstawiają ludzi wykonujących dane akcje. Charakteryzuje się dużymi wariacjami w ruchu kamery, wyglądzie i pozie obiektów, skali obiektów, punkcie widzenia, złożonym tle oraz warunkach oświetleniowych. Ze względów praktycznych z wyjściowego zbioru danych wybrano jedynie pliki z 8 pierwszych klas. Po preprocessingu każdy materiał wideo jest reprezentowany przez 4 kolorowe klatki o wymiarach 224 x 224 pikseli.

5. Analiza wyników

Accuracy na zbiorze testowym wyniosło około 70%, a loss około 1.99. Poniżej w tabelce po lewej stronie przedstawione są wartości precision i recall na zbiorze testowym dla poszczególnych klas. Po prawej stronie znajduje się porównanie wyników na confusion matrix: wysokie wartości na przekątnej wskazują wysoką dokładność modelu.

W dalszych krokach: do ulepszenia wydajności modelu można eksperymentować z innymi architekturami, np. wykorzystać więcej bloków warstw konwolucyjnych z większą liczbą filtrów, wykorzystać głębsze sieci typu ResNet lub attention mechanisms. Można też spróbować wykorzystać inne funkcje aktywacji lub learning rate i liczbę epok.

ClassName	Precision [%]	Recall [%]
ApplyEyeMakeup	66,7	60
ApplyLipstick	80	80
Archery	85,7	60
BabyCrawling	64,3	90
BalanceBeam	80	40
BandMarching	47,6	100
BaseballPitch	100	70
BasketballDunk	85,7	60