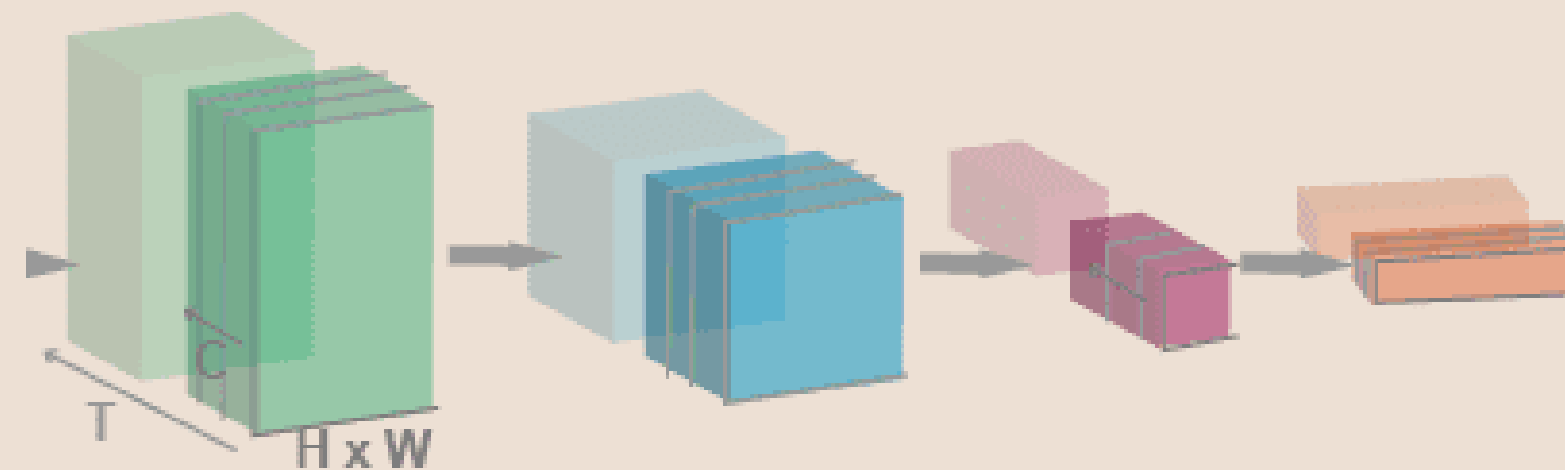


Класифікація зображень з використанням нейронних мереж

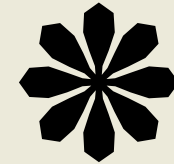
Студент: Кирпа Микита, ПМ-4
Науковий керівник: Бучко Олена Андріївна



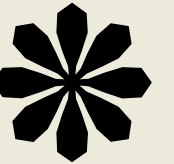
Дослідженн

я

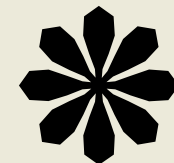
Мета



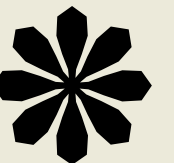
- Математичний опис складових та процесу навчання нейронних мереж. Порівняння декількох підходів до тренування та дотренування згорткових нейронних мереж. Оптимізація моделей на власних даних і порівняння отриманих результатів.



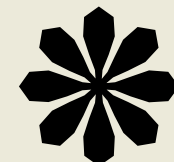
Предметна область



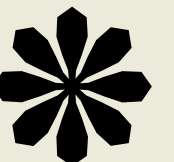
- Класифікація транспортних засобів на зображеннях для побудови якісної моделі для вирішення данної проблеми, яку можна використовувати для подальшої розробки додатків.



Об'єкт дослідження



- Дослідження архітектури та процесу тренування нейронних мереж з математичної точки зору, обробка зображень та побудова власної моделі.



Етапи дослідження



Пошук та збір даних необхідних для тренування моделей



Обробка та перетворення зібраного датасету для навчання нейронних мереж



Створення власної архітектури моделі, її оптимізація та опис математичних процесів



Оптимізація попередньо натренованого алгоритму на власних даних



Порівняння отриманих результатів

Збір даних необхідних для тренування моделей

Audi



BMW



Bentley



SEAT



Augmented SEAT



Augmented SEAT



Побудова власної архітектури згорткової нейронної мережі

$$y_{ij} = \sum_{k=0}^{m-1} \sum_{l=0}^{n-1} x_{i-s+k, j-s+l} \times w_{k,l} \text{ де:}$$

- y_{ij} - значення елемента вихідної матриці згортки з координатами (i, j);
- $x_{i-s+k, j-s+l}$ - значення елемента вхідної матриці згортки з координатами (i · s + k, j · s + l);
- $w_{k,l}$ - значення відповідного елемента фільтру згортки з координатами (k, l);
- m - розмір фільтру згортки по висоті;
- n - розмір фільтру згортки по ширині;
- s - значення зсуву(stride)

$$h_{x,y} = \frac{1}{K^2} \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} l_{(x+i), (y+j)}$$

- де $h_{x,y}$ - елемент вихідної матриці пулінгу, який відповідає області пулінгу (x, y) вхідної матриці
- $l_{(x+i), (y+j)}$ - елемент вхідної матриці, який знаходиться в (x + i)-ому рядку та (y + j) - ому стовпці
- K - розмір пулінг фільтру.

$$h_{x,y} = \max_{i=0, \dots, K-1, j=0, \dots, K-1} l_{(x+i), (y+j)}$$

- де $h_{x,y}$ - елемент вихідної матриці пулінгу, який відповідає області пулінгу (x, y) вхідної матриці
- $l_{(x+i), (y+j)}$ - елемент вхідної матриці, який знаходиться в (x + i)-ому рядку та (y + j) - ому стовпці
- K - розмір пулінг фільтру.

Повнозв'язний шар зазвичай слідує у кінці нейронної мережі і виконує функцію остаточного збору і виокремлення ознак визначених на попередніх шарах ЗНМ.

$$Y = WX+b$$

Де W - це ваги повнозв'язного шару, X - вектор ознак попередньо зплющений до матриці з розмірністю 1xN, і b - це випадкова похибка

Згорткові шари з моделі запропонованої у дослідженні

conv_layer_1 (Conv2D)	(None, 224, 224, 16)	448
conv_layer_2 (Conv2D)	(None, 112, 112, 32)	4640
conv_layer_3 (Conv2D)	(None, 56, 56, 64)	18496

Пулінг шари з моделі запропонованої у дослідженні

max_pooling_1 (MaxPooling2D)	(None, 112, 112, 16)	0
max_pooling_2 (MaxPooling2D)	(None, 56, 56, 32)	0
avg_pool (AveragePooling2D)	(None, 28, 28, 64)	0

Повнозв'язні шари з моделі запропонованої у дослідженні

dense_48 (Dense)	(None, 64)	3211328
dense_49 (Dense)	(None, 55)	3575

Приклад трансформації зображення після проходження крізь шари 3НМ



Тренування згорткової нейронної мережі

Матриця похідних що використовується для оновлення вагів у шарі згортки

$$\frac{dL}{dF} = Conv(X, \frac{dL}{dO}),$$

де X - це вхідні дані у шар згортки,

$\frac{dL}{dO}$ - градієнт помилки на вихідних даних наступного шару,

$Conv$ - функція згортки.

Процес оптимізації нейронної мережі з власною архітектурою

Epoch 1/15 115/115 [=====] - 47s 380ms/step - loss: 44.2438 - accuracy: 0.2154 - val_loss: 38.7444 - val_accuracy: 0.1247
Epoch 2/15 115/115 [=====] - 43s 372ms/step - loss: 36.7208 - accuracy: 0.4146 - val_loss: 37.5840 - val_accuracy: 0.2280
Epoch 3/15 115/115 [=====] - 42s 360ms/step - loss: 35.4571 - accuracy: 0.5326 - val_loss: 36.4299 - val_accuracy: 0.2996
Epoch 4/15 115/115 [=====] - 43s 371ms/step - loss: 34.3737 - accuracy: 0.6320 - val_loss: 35.2886 - val_accuracy: 0.3629
Epoch 5/15 115/115 [=====] - 42s 359ms/step - loss: 33.3836 - accuracy: 0.7075 - val_loss: 34.2996 - val_accuracy: 0.4564
Epoch 6/15 115/115 [=====] - 43s 375ms/step - loss: 32.4187 - accuracy: 0.7732 - val_loss: 34.5438 - val_accuracy: 0.4036
Epoch 7/15 115/115 [=====] - 42s 367ms/step - loss: 31.4881 - accuracy: 0.8265 - val_loss: 35.5980 - val_accuracy: 0.3236
Epoch 8/15 115/115 [=====] - 43s 372ms/step - loss: 30.6260 - accuracy: 0.8498 - val_loss: 34.6052 - val_accuracy: 0.3836
Epoch 9/15 115/115 [=====] - 42s 365ms/step - loss: 29.6851 - accuracy: 0.8866 - val_loss: 33.4204 - val_accuracy: 0.3909
Epoch 10/15 115/115 [=====] - 43s 372ms/step - loss: 28.8432 - accuracy: 0.8848 - val_loss: 33.9634 - val_accuracy: 0.3796
Epoch 11/15 115/115 [=====] - 43s 371ms/step - loss: 27.9232 - accuracy: 0.9064 - val_loss: 31.3092 - val_accuracy: 0.4760
Epoch 12/15 115/115 [=====] - 42s 365ms/step - loss: 27.0716 - accuracy: 0.9047 - val_loss: 34.3938 - val_accuracy: 0.3495
Epoch 13/15 115/115 [=====] - 42s 366ms/step - loss: 26.1509 - accuracy: 0.9223 - val_loss: 31.2132 - val_accuracy: 0.4433
Epoch 14/15 115/115 [=====] - 43s 368ms/step - loss: 25.2905 - accuracy: 0.9286 - val_loss: 31.5239 - val_accuracy: 0.4295
Epoch 15/15 115/115 [=====] - 41s 358ms/step - loss: 24.4021 - accuracy: 0.9380 - val_loss: 28.4410 - val_accuracy: 0.5193

Матриця від вхідних даних що використовується для розповсюдження помилки і підрахунку

градієнтів

$$\frac{dL}{dX} = FullConv(F^T, \frac{dL}{dO}),$$

де F^T - транспонована матриця значень фільтру,

$\frac{dL}{dO}$ - градієнт помилки на вихідних даних наступного шару,

$FullConv$ - функція повної згортки(обернена функція згортки).

У дослідженні було використано функцію крос-ентропії, що задається формулою:

$$H(p, y) = - \sum_{i=1}^N y_i \cdot \log(p_i)$$

Де:

- $H(p, y)$ - значення функції втрат
- p - вектор прогнозованих значень для одного зображення
- y - вектор прогнозованих значень для одного зображення(буде містити 1 для індексу реального класу, і $N - 1$ нулів для інших класів)
- N - кількість класів зображень
- y_i - реальне значення класу i
- p_i - ймовірність належності до класу i

Техніка тонкого налаштування та оптимізація популярної моделі на власних даних

Процес оптимізації нейронної мережі Efficient Net B0

Epoch 1/12

115/115 [=====] - 124s 997ms/step - loss: 2.5652 - accuracy: 0.4987 - val_loss: 2.1995 - val_accuracy: 0.3862

Epoch 2/12

115/115 [=====] - 109s 939ms/step - loss: 1.2081 - accuracy: 0.6925 - val_loss: 1.3203 - val_accuracy: 0.6673

Epoch 3/12

115/115 [=====] - 112s 968ms/step - loss: 0.9849 - accuracy: 0.7415 - val_loss: 1.1101 - val_accuracy: 0.7149

Epoch 4/12

115/115 [=====] - 109s 939ms/step - loss: 0.8417 - accuracy: 0.7695 - val_loss: 1.0709 - val_accuracy: 0.7371

Epoch 5/12

115/115 [=====] - 108s 926ms/step - loss: 0.8369 - accuracy: 0.7675 - val_loss: 1.1048 - val_accuracy: 0.7436

Epoch 6/12

115/115 [=====] - 107s 927ms/step - loss: 0.7702 - accuracy: 0.7925 - val_loss: 1.0750 - val_accuracy: 0.7502

Epoch 7/12

115/115 [=====] - 108s 930ms/step - loss: 0.7441 - accuracy: 0.7919 - val_loss: 1.1071 - val_accuracy: 0.7545

Epoch 8/12

115/115 [=====] - 107s 920ms/step - loss: 0.7047 - accuracy: 0.8007 - val_loss: 1.0573 - val_accuracy: 0.7705

Epoch 9/12

115/115 [=====] - 109s 936ms/step - loss: 0.6658 - accuracy: 0.8132 - val_loss: 1.0660 - val_accuracy: 0.7662

Epoch 10/12

115/115 [=====] - 108s 931ms/step - loss: 0.6867 - accuracy: 0.8102 - val_loss: 1.0639 - val_accuracy: 0.7709

Epoch 11/12

115/115 [=====] - 106s 911ms/step - loss: 0.6588 - accuracy: 0.8135 - val_loss: 1.0370 - val_accuracy: 0.7735

Epoch 12/12

115/115 [=====] - 108s 932ms/step - loss: 0.6634 - accuracy: 0.8144 - val_loss: 1.1026 - val_accuracy: 0.7593

Порівняння результатів власної мережі та моделі Efficient Net на валідаційній вибірці

$$\frac{1}{K} \sum_{n=0}^K \frac{TP}{(TP + TN)},$$

де K - це кількість класів даних,

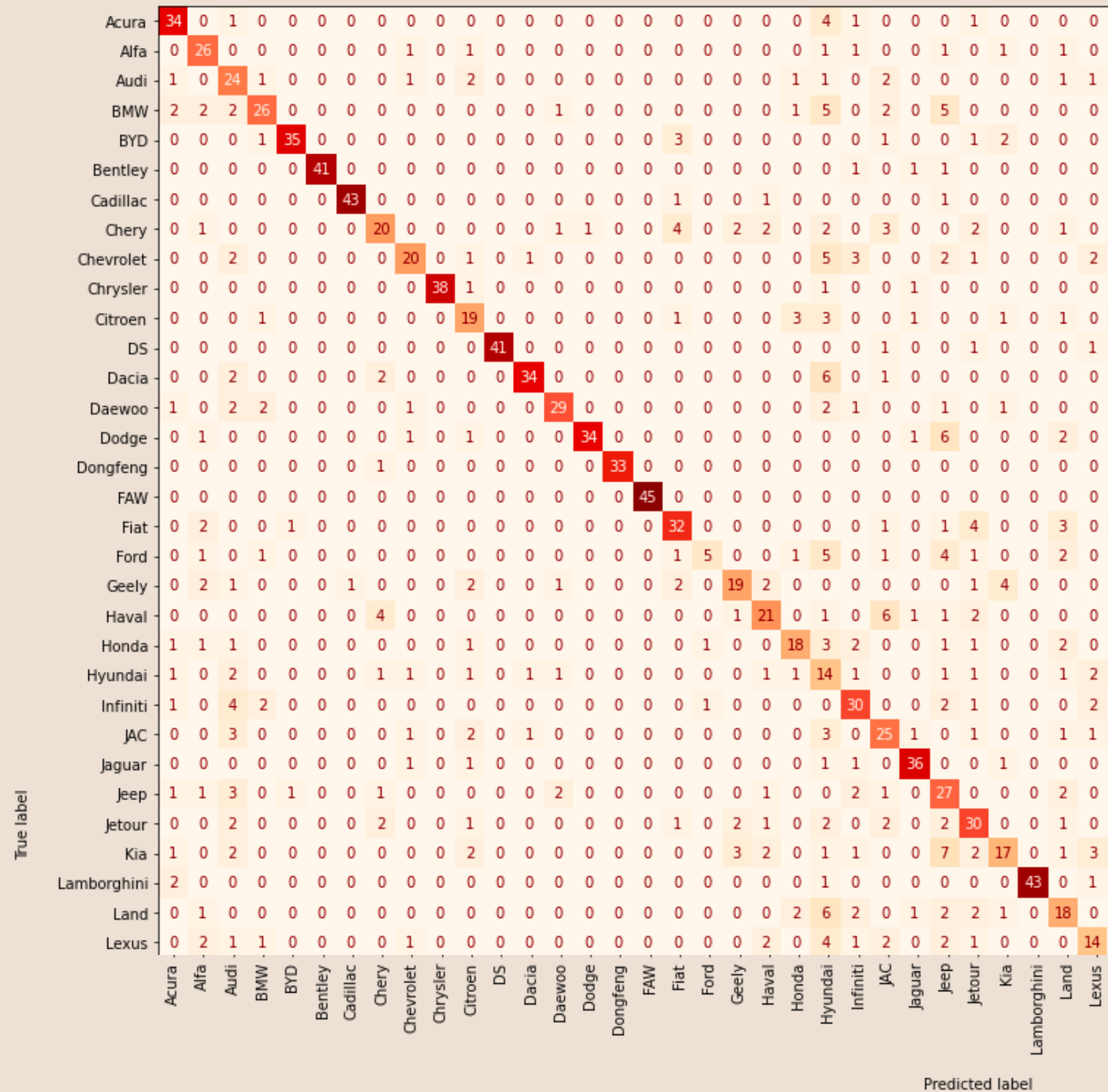
TP - це кількість вірно класифікованих зображень певного класу,

TN - кількість невірно класифікованих зображень певного класу.

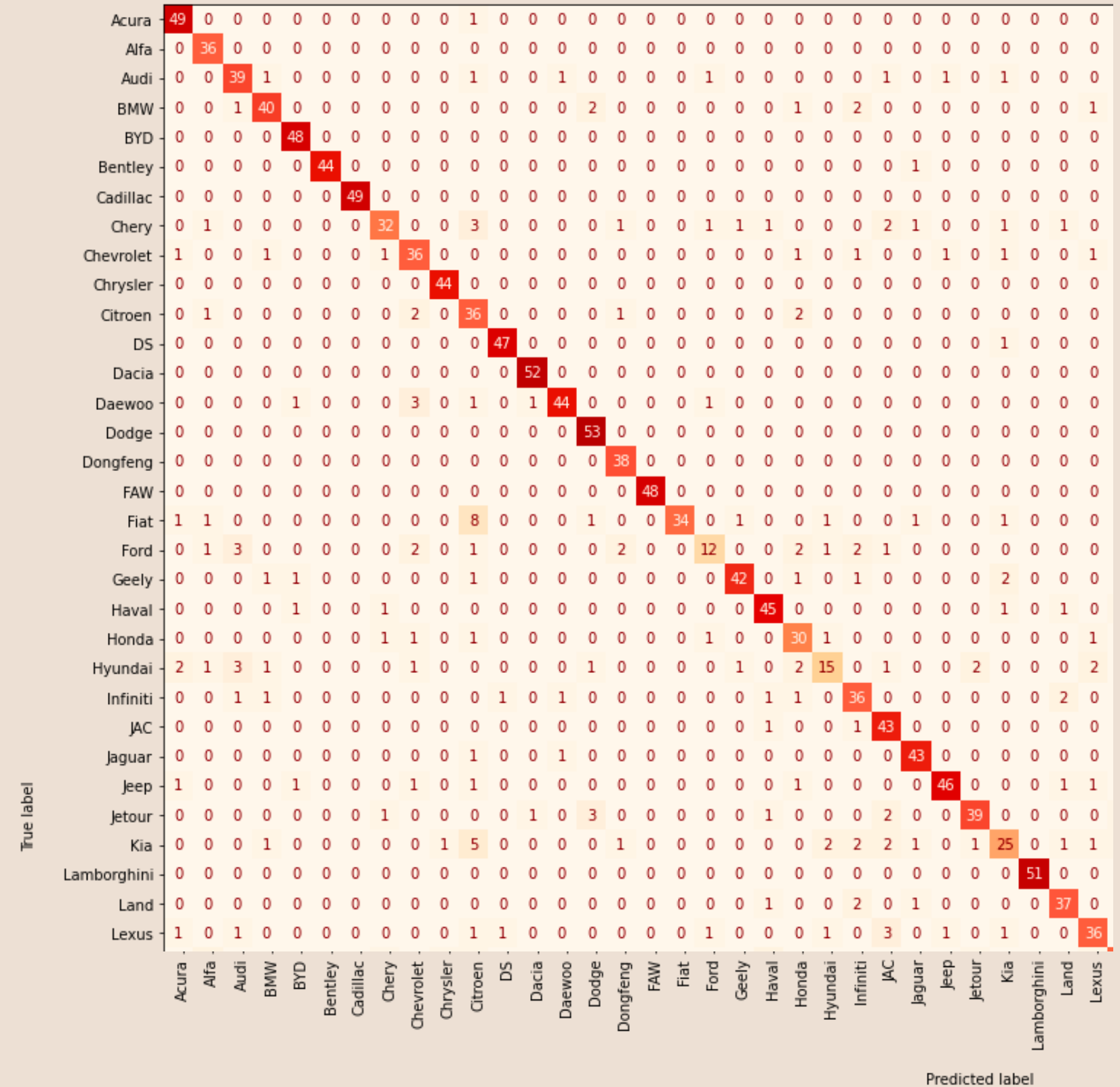
- Зважена точність для моделі з власною архітектурою дорівнює 0.53
- Зважена точність для дотренованої моделі Efficient Net дорівнює 0.77

Порівняння результатів власної мережі та моделі Efficient Net на валідаційній вибірці

Результати власної моделі



Результати моделі Efficient Net



Висновки

- Наразі створення власних архітектур ЗНМ є дуже складним і ресурсозатратним процесом
- Для використання у сучасних додатках набагато краще і швидше оптимізувати вже існуючі моделі
- При наявності великих розрахункових ресурсів та часу можна будувати власні моделі

Дякую за увагу!