



دانشکده مهندسی کامپیوتر

دکتر محمدی

بهار ۱۴۰۰

تمرین هفتم

یادگیری عمیق

مجتبی نافذ 96431335

۱. در این سوال قصد داریم مسئله دسته‌بندی نوع خودرو بر اساس تصویر را شبیه‌سازی کنیم. در این سوال از مجموعه داده [Cars](#) استفاده خواهیم کرد که شامل ۱۶۱۸۵ تصویر از ۱۹۶ دسته خودرو است.



این نمونه کد نحوه کار با این مجموعه داده را نشان می‌دهد. همانطور که در این لینک مشخص است، شبکه [Xception](#) یکی از شبکه‌های موفق در حوزه دسته‌بندی تصویر بوده است. در این تمرین، مراحل زیر را انجام دهید و نتایج بدست آمده را به طور دقیق گزارش، مقایسه و تحلیل کنید (تحلیل نتایج و انجام آزمایش‌های گوناگون بسیار مهم است).

الف) این شبکه را با استفاده از وزن‌های تصادفی آموزش بدهید.

ب) از مدل پیش‌آمورخته بر روی مجموعه داده ImageNet به عنوان Feature Extractor استفاده کنید.

پ) چندین لایه انتهایی شبکه را fine-tune کنید.

ت) برای بهترین مدل، Grad-CAM را برای چند تصویر نمونه نمایش دهید و تحلیل کنید.

الف) در این حالت که در اپیک‌های ابتدایی به لحاظ دقت تقریباً وضعیت افتضاحی داشته اما مشاهده می‌کنیم در نهایت به دقت ۵۷ درصد در تست رسیده که به نسبت بقیه دقت خوبی به شمار می‌رود و روی داده‌های آموزشی هم که overfit کرد.

مزیت این روش آن است که دقیقاً همین مسئله را مدل می‌کند و راه حل خود را کامل با مسئله منطبق می‌کند تعداد پارامترهای قابل یادگیری آن خیلی بالاست (۲۱ میلیون) پس کند اجرا خواهد شد و هر اپیک تقریباً ۲۲۷ ثانیه طول می‌کشد. این مدل نیازمند داده‌ی بالاست تا تعمیم‌دهی خوبی را خروجی دهد. اما در این جا به دلیل application specific بودن از بقیه‌ی مدل‌ها دقت بالاتری را به ما نشان می‌دهد.



Total params: 21,263,084
Trainable params: 21,208,556
Non-trainable params: 54,528

Epoch 1/30
255/255 [=====] - 265s 905ms/step - loss: 5.3772 - accuracy: 0.0089 - val_loss: 5.2775 - val_accuracy: 0.0086
Epoch 2/30
255/255 [=====] - 228s 895ms/step - loss: 5.1866 - accuracy: 0.0092 - val_loss: 5.2504 - val_accuracy: 0.0099
Epoch 3/30
255/255 [=====] - 227s 892ms/step - loss: 5.1305 - accuracy: 0.0122 - val_loss: 5.6303 - val_accuracy: 0.0103
Epoch 4/30
255/255 [=====] - 228s 893ms/step - loss: 5.0308 - accuracy: 0.0145 - val_loss: 9.2839 - val_accuracy: 0.0129
Epoch 5/30
255/255 [=====] - 228s 893ms/step - loss: 4.9003 - accuracy: 0.0234 - val_loss: 17.9470 - val_accuracy: 0.0092
Epoch 6/30
255/255 [=====] - 228s 893ms/step - loss: 4.6757 - accuracy: 0.0339 - val_loss: 11.4351 - val_accuracy: 0.0123
Epoch 7/30
255/255 [=====] - 229s 897ms/step - loss: 4.4095 - accuracy: 0.0488 - val_loss: 7.5720 - val_accuracy: 0.0225
Epoch 8/30
255/255 [=====] - 228s 896ms/step - loss: 4.1277 - accuracy: 0.0781 - val_loss: 4.6703 - val_accuracy: 0.0485
Epoch 9/30
255/255 [=====] - 228s 896ms/step - loss: 3.7400 - accuracy: 0.1217 - val_loss: 4.1814 - val_accuracy: 0.0904
Epoch 10/30
255/255 [=====] - 229s 896ms/step - loss: 3.3237 - accuracy: 0.1897 - val_loss: 4.3712 - val_accuracy: 0.1127
Epoch 11/30
255/255 [=====] - 228s 894ms/step - loss: 2.8785 - accuracy: 0.2680 - val_loss: 3.9043 - val_accuracy: 0.1744
Epoch 12/30
255/255 [=====] - 227s 891ms/step - loss: 2.4800 - accuracy: 0.3455 - val_loss: 4.5862 - val_accuracy: 0.1761
Epoch 13/30
255/255 [=====] - 228s 895ms/step - loss: 1.9590 - accuracy: 0.4723 - val_loss: 2.9421 - val_accuracy: 0.3146
Epoch 14/30
255/255 [=====] - 227s 889ms/step - loss: 1.5749 - accuracy: 0.5617 - val_loss: 2.6851 - val_accuracy: 0.3931
Epoch 15/30
255/255 [=====] - 228s 893ms/step - loss: 1.2588 - accuracy: 0.6400 - val_loss: 2.2633 - val_accuracy: 0.4526
Epoch 16/30
255/255 [=====] - 227s 891ms/step - loss: 1.0495 - accuracy: 0.6899 - val_loss: 3.1965 - val_accuracy: 0.3568
Epoch 17/30
255/255 [=====] - 227s 892ms/step - loss: 0.8784 - accuracy: 0.7355 - val_loss: 2.0433 - val_accuracy: 0.5042
Epoch 18/30
255/255 [=====] - 227s 892ms/step - loss: 0.7050 - accuracy: 0.7882 - val_loss: 2.4904 - val_accuracy: 0.4618
Epoch 19/30
255/255 [=====] - 228s 893ms/step - loss: 0.5436 - accuracy: 0.8306 - val_loss: 2.2170 - val_accuracy: 0.5053
Epoch 20/30
255/255 [=====] - 227s 889ms/step - loss: 0.4841 - accuracy: 0.8474 - val_loss: 2.2965 - val_accuracy: 0.5273
Epoch 21/30
255/255 [=====] - 227s 892ms/step - loss: 0.3896 - accuracy: 0.8761 - val_loss: 2.2597 - val_accuracy: 0.5210
Epoch 22/30
255/255 [=====] - 228s 895ms/step - loss: 0.3053 - accuracy: 0.9027 - val_loss: 2.6107 - val_accuracy: 0.5059
Epoch 23/30
255/255 [=====] - 227s 891ms/step - loss: 0.2834 - accuracy: 0.9099 - val_loss: 2.1911 - val_accuracy: 0.5384
Epoch 24/30
255/255 [=====] - 227s 891ms/step - loss: 0.2493 - accuracy: 0.9229 - val_loss: 2.8128 - val_accuracy: 0.5098
Epoch 25/30
255/255 [=====] - 227s 891ms/step - loss: 0.2558 - accuracy: 0.9248 - val_loss: 3.0223 - val_accuracy: 0.4804
Epoch 26/30
255/255 [=====] - 227s 890ms/step - loss: 0.2085 - accuracy: 0.9337 - val_loss: 2.5544 - val_accuracy: 0.5227
Epoch 27/30
255/255 [=====] - 227s 889ms/step - loss: 0.1989 - accuracy: 0.9374 - val_loss: 2.4633 - val_accuracy: 0.5488
Epoch 28/30
255/255 [=====] - 227s 890ms/step - loss: 0.1894 - accuracy: 0.9446 - val_loss: 2.0489 - val_accuracy: 0.5886
Epoch 29/30
255/255 [=====] - 227s 892ms/step - loss: 0.1842 - accuracy: 0.9430 - val_loss: 2.6425 - val_accuracy: 0.5392
Epoch 30/30
255/255 [=====] - 228s 894ms/step - loss: 0.1926 - accuracy: 0.9340 - val_loss: 2.1102 - val_accuracy: 0.5787

(ب)

با این که انتظار داشتیم بهتر از این ها دقت بگیریم مشاهده شده که دقت کمتر از حالت رندوم شد. هم دقت روی آموزشی و هم تست.

و علت آن این بود که مدل ما در این مسئله تمایز ماشین ها را باید تشخیص دهد

و این مدل با مدل imagenet که کلی تصاویر حیوانات و غیره را هم دارد شروع به کار می کند و در واقع لایه های کانولوشنی خوب برای مسئله ی imagenet برای مسئله cars خوب نیست ،

با این که احتمالاً تعمیم دهی بالاتری به نسبت قبلی خواهد داشت. اما دقت خوبی ندیدیم.

در این جا چون پارامترهای لایه ی کانولوشنی را آپدیت نکردیم مقدار زمان مورد نیاز برای هر اپیک به ۱۹۵ ثانیه کاهش یافته. (تعداد پارامترهای قابل آموزش به ۴۰۱۰۰۰ تا کاهش یافت)



Total params: 21,263,084
Trainable params: 401,604
Non-trainable params: 20,861,480

```
Epoch 1/30
255/255 [=====] - 199s 774ms/step - loss: 4.7031 - accuracy: 0.0639 - val_loss: 3.5714 - val_accuracy: 0.1618
Epoch 2/30
255/255 [=====] - 196s 769ms/step - loss: 3.2471 - accuracy: 0.2495 - val_loss: 3.2486 - val_accuracy: 0.2277
Epoch 3/30
255/255 [=====] - 196s 769ms/step - loss: 2.8044 - accuracy: 0.3307 - val_loss: 3.0712 - val_accuracy: 0.2617
Epoch 4/30
255/255 [=====] - 196s 768ms/step - loss: 2.4963 - accuracy: 0.4012 - val_loss: 2.9353 - val_accuracy: 0.2906
Epoch 5/30
255/255 [=====] - 196s 768ms/step - loss: 2.2810 - accuracy: 0.4556 - val_loss: 2.8805 - val_accuracy: 0.2981
Epoch 6/30
255/255 [=====] - 196s 768ms/step - loss: 2.0882 - accuracy: 0.4889 - val_loss: 2.7766 - val_accuracy: 0.3245
Epoch 7/30
255/255 [=====] - 196s 769ms/step - loss: 1.9690 - accuracy: 0.5271 - val_loss: 2.7284 - val_accuracy: 0.3389
Epoch 8/30
255/255 [=====] - 196s 769ms/step - loss: 1.8274 - accuracy: 0.5671 - val_loss: 2.6494 - val_accuracy: 0.3513
Epoch 9/30
255/255 [=====] - 195s 768ms/step - loss: 1.6831 - accuracy: 0.6021 - val_loss: 2.6876 - val_accuracy: 0.3431
Epoch 10/30
255/255 [=====] - 196s 769ms/step - loss: 1.6210 - accuracy: 0.6111 - val_loss: 2.6157 - val_accuracy: 0.3574
Epoch 11/30
255/255 [=====] - 196s 768ms/step - loss: 1.5416 - accuracy: 0.6268 - val_loss: 2.6036 - val_accuracy: 0.3638
Epoch 12/30
255/255 [=====] - 196s 770ms/step - loss: 1.4789 - accuracy: 0.6510 - val_loss: 2.5974 - val_accuracy: 0.3685
Epoch 13/30
255/255 [=====] - 196s 769ms/step - loss: 1.3597 - accuracy: 0.6783 - val_loss: 2.5512 - val_accuracy: 0.3783
Epoch 14/30
255/255 [=====] - 197s 774ms/step - loss: 1.3391 - accuracy: 0.6849 - val_loss: 2.5385 - val_accuracy: 0.3847
Epoch 15/30
255/255 [=====] - 195s 766ms/step - loss: 1.2604 - accuracy: 0.7050 - val_loss: 2.5541 - val_accuracy: 0.3736
Epoch 16/30
255/255 [=====] - 196s 768ms/step - loss: 1.2090 - accuracy: 0.7134 - val_loss: 2.5396 - val_accuracy: 0.3837
Epoch 17/30
255/255 [=====] - 195s 767ms/step - loss: 1.1375 - accuracy: 0.7320 - val_loss: 2.5160 - val_accuracy: 0.3902
Epoch 18/30
255/255 [=====] - 196s 768ms/step - loss: 1.1121 - accuracy: 0.7358 - val_loss: 2.4955 - val_accuracy: 0.3950
Epoch 19/30
255/255 [=====] - 195s 767ms/step - loss: 1.0568 - accuracy: 0.7562 - val_loss: 2.4932 - val_accuracy: 0.3929
Epoch 20/30
255/255 [=====] - 195s 768ms/step - loss: 1.0047 - accuracy: 0.7666 - val_loss: 2.5161 - val_accuracy: 0.3886
Epoch 21/30
255/255 [=====] - 195s 767ms/step - loss: 0.9771 - accuracy: 0.7758 - val_loss: 2.4767 - val_accuracy: 0.4073
Epoch 22/30
255/255 [=====] - 195s 767ms/step - loss: 0.9369 - accuracy: 0.7894 - val_loss: 2.5000 - val_accuracy: 0.3988

Epoch 23/30
255/255 [=====] - 195s 767ms/step - loss: 0.9008 - accuracy: 0.7940 - val_loss: 2.4712 - val_accuracy: 0.4058
Epoch 24/30
255/255 [=====] - 196s 768ms/step - loss: 0.8899 - accuracy: 0.7985 - val_loss: 2.5175 - val_accuracy: 0.3951
Epoch 25/30
255/255 [=====] - 197s 772ms/step - loss: 0.8413 - accuracy: 0.8106 - val_loss: 2.4863 - val_accuracy: 0.4036
Epoch 26/30
255/255 [=====] - 195s 767ms/step - loss: 0.7939 - accuracy: 0.8205 - val_loss: 2.4607 - val_accuracy: 0.4089
Epoch 27/30
255/255 [=====] - 195s 767ms/step - loss: 0.7941 - accuracy: 0.8168 - val_loss: 2.4586 - val_accuracy: 0.4097
Epoch 28/30
255/255 [=====] - 195s 766ms/step - loss: 0.7641 - accuracy: 0.8253 - val_loss: 2.4930 - val_accuracy: 0.3988
Epoch 29/30
255/255 [=====] - 195s 765ms/step - loss: 0.7283 - accuracy: 0.8334 - val_loss: 2.4709 - val_accuracy: 0.4126
Epoch 30/30
255/255 [=====] - 195s 765ms/step - loss: 0.7147 - accuracy: 0.8459 - val_loss: 2.4759 - val_accuracy: 0.4105
```

ج) در این حالت زمان مورد نیاز برای آموزش بیشتر خواهد شد. از **feature extraction** خواهد شد چون لایه ها بیشتری آموزش دیدند. و ۵۰ میلیون پارامتر قابل آموزش دارد

در این حالت دقت بیشتر از **feature extraction** گرفتیم چون لایه های قبلی هم متناسب با خود مسئله ی **cars** شدند و **application specific** تر آموزش دید و به دقت حدود 55 رسید. اما در عوض ۵ میلیون پارامتر قابل آموزش دارد و نکته مهم و بد ماجرا اینجاست که **overfit** ما شدیدتر شد چون **application specific** تر شدیم و تصاویر را حفظ کردیم.

Total params: 21,263,084
Trainable params: 5,150,404
Non-trainable params: 16,112,680

Epoch	Time	loss	accuracy	val_loss	val_accuracy
Epoch 1/30	218s 846ms/step	4.3436	0.1141	2.6570	0.3150
Epoch 2/30	212s 832ms/step	1.7877	0.5333	2.0574	0.4601
Epoch 3/30	211s 827ms/step	1.0013	0.7230	1.9496	0.5021
Epoch 4/30	211s 829ms/step	0.6168	0.8418	1.9512	0.5004
Epoch 5/30	210s 826ms/step	0.4136	0.8878	2.1006	0.5198
Epoch 6/30	210s 823ms/step	0.3061	0.9144	1.9405	0.5330
Epoch 7/30	212s 834ms/step	0.2469	0.9304	1.9744	0.5595
Epoch 8/30	211s 828ms/step	0.2127	0.9407	2.0646	0.5462
Epoch 9/30	210s 826ms/step	0.1737	0.9507	2.1770	0.5385
Epoch 10/30	213s 836ms/step	0.1539	0.9582	2.2154	0.5293
Epoch 11/30	209s 821ms/step	0.1590	0.9568	2.4513	0.5100
Epoch 12/30	208s 818ms/step	0.1396	0.9637	2.2037	0.5532
Epoch 13/30	209s 819ms/step	0.1358	0.9595	2.3063	0.5381
Epoch 14/30	212s 831ms/step	0.1429	0.9619	2.5671	0.5165
Epoch 15/30	211s 828ms/step	0.1336	0.9608	2.3293	0.5514
Epoch 16/30	211s 826ms/step	0.1222	0.9645	2.4531	0.5481
Epoch 17/30	210s 822ms/step	0.1254	0.9625	2.5027	0.5389
Epoch 18/30	210s 823ms/step	0.0887	0.9763	2.3137	0.5660
Epoch 19/30	210s 823ms/step	0.0887	0.9758	2.7922	0.5099
Epoch 20/30	209s 821ms/step	0.1227	0.9649	2.4232	0.5575
Epoch 21/30	209s 819ms/step	0.1006	0.9733	2.4323	0.5386

```
Epoch 22/30
255/255 [=====] - 209s 819ms/step - loss: 0.1105 - accuracy: 0.9701 - val_loss: 2.7684 - val_accuracy: 0.5175
Epoch 23/30
255/255 [=====] - 208s 817ms/step - loss: 0.0848 - accuracy: 0.9742 - val_loss: 2.6578 - val_accuracy: 0.5404
Epoch 24/30
255/255 [=====] - 208s 817ms/step - loss: 0.0801 - accuracy: 0.9770 - val_loss: 2.3863 - val_accuracy: 0.5705
Epoch 25/30
255/255 [=====] - 208s 816ms/step - loss: 0.0808 - accuracy: 0.9748 - val_loss: 2.4445 - val_accuracy: 0.5525
Epoch 26/30
255/255 [=====] - 208s 817ms/step - loss: 0.0778 - accuracy: 0.9777 - val_loss: 2.5672 - val_accuracy: 0.5579
Epoch 27/30
255/255 [=====] - 208s 817ms/step - loss: 0.0706 - accuracy: 0.9803 - val_loss: 2.5963 - val_accuracy: 0.5427
Epoch 28/30
255/255 [=====] - 208s 818ms/step - loss: 0.0766 - accuracy: 0.9786 - val_loss: 2.6505 - val_accuracy: 0.5486
Epoch 29/30
255/255 [=====] - 209s 822ms/step - loss: 0.1036 - accuracy: 0.9701 - val_loss: 2.8043 - val_accuracy: 0.5289
Epoch 30/30
255/255 [=====] - 209s 822ms/step - loss: 0.0728 - accuracy: 0.9798 - val_loss: 2.7760 - val_accuracy: 0.5365
```

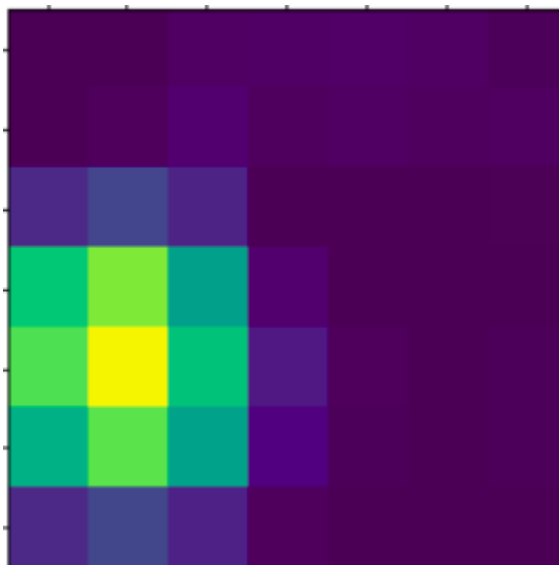
ج:

بهترین مدل در این جا همان مدل با وزن رندوم بود

من به علت حواس پرتی grad came را بر روی مدل fine tuning اجرا کردم و چون دیر شد و وقت نبود همین را

ارسال نمودم

هدف این مسئله کشیدن grad-cam بود که مشکلی روش آن فرقی ندارد و مدل با وزن رندوم ساده تر هم هست.



Grad-CAM



Heatmap

همانطور که مشخص است ما تقریباً مفهومی های خوبی را یاد گرفته ایم اما در کل هنوز خیلی کار دارد و باید مدل بهتر شود.

بهتر بود هر چه بیشتر به مساحت ماشین حساس شود. و کم تر به خارج از ماشین حساس شود