به نام خدا

دانشگاه صنعتی امیرکبیر

دانشکده مهندسی کامپیوتر

پاسخ تمرین سری چهارم شبکه‌های عصبی

استاد:

دکتر صفابخش

دانشجو:

حلیمه رحیمی

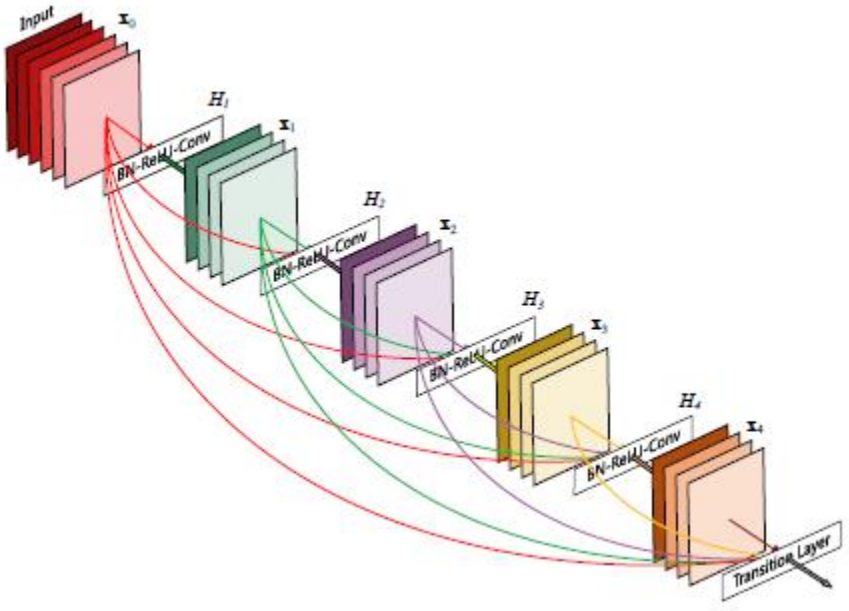
شماره دانشجویی:

۹۹۱۳۱۰۴۳

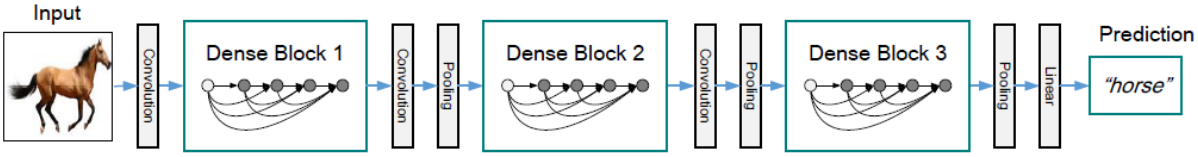
بهار ۱۴۰۰

۱- شبکه DenseNet شامل بلوک‌هایی است که در درون آنها، خروجی‌های هر لایه به لایه‌های بعد از آنها داده می‌شود؛ به طور مثال ورودی لایه‌ی چهارم در یک بلوک، خروجی‌های لایه‌های اول، دوم و سوم می‌باشد که به شکل عمقی به یکدیگر پیوسته اند (Concatenate).

ابتدای این شبکه یک لایه‌ی کانولوشنی برای کشف ویژگی وجود دارد که خروجی‌هایش وارد اولین بلوک می‌گردد. ساختار بلوک‌های Dense به شکل زیر است. در درون این بلوک‌ها اندازه نقشه‌های ویژگی یکسان است که باعث می‌شود پیونددهی آنها آسان باشد.

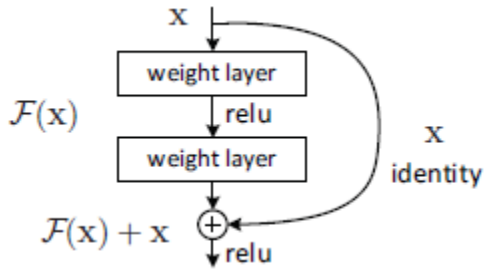


هر لایه در بلوک، شامل لایه‌ی Batch Normalization، Relu و Convolution می‌باشد. به علاوه تعداد فیلترهای هر لایه‌ی درون بلوک را درجه‌ی رشد (Growth Rate) می‌نامند.



در میان این بلوک‌ها یک لایه‌ی انتقال وجود دارد که شامل لایه‌ی کانولوشنی و ادغام می‌باشد و اندازه‌ی نقشه‌های ویژگی را کاهش می‌دهد. می‌توان از لایه‌ی کانولوشنی ۱X۱ با عنوان لایه‌ی گلوگاهی برای انتقال استفاده کرد که سرعت را بیشتر می‌کند. همچنین می‌توان تعداد نقشه‌های ویژگی را کنترل کرد (کاهش داد یا تغییری نداد، Compression).

شبکه ResNet شامل بلوک‌هایی است که در درون آنها، خروجی‌های هر لایه با ورودی آن جمع شده و به لایه‌ی بعدی داده می‌شود. ساختار بلوک را در زیر می‌توان دید.



تعداد پارامترهای شبکه‌ی DenseNet بسیار کمتر از تعداد پارامترها شبکه‌ی ResNet است درحالیکه نتیجه‌ی بهتری دارد. در واقع شبکه‌ی DenseNet ویژگی‌ها را از لایه‌های مختلف جمع می‌کند. درحالیکه شبکه‌های معمول کانولوشن اطلاعات تصویر را در هر لایه از لایه‌های قبلی بدست می‌آورد. در لایه‌های پیشین اطلاعاتی هست که می‌توان همچنان به کار برد و کوچک و کوچک‌تر کردن تصویر تاثیر آنها را در تشخیص کاهش می‌دهد. ResNet با آنکه از اطلاعات لایه‌ی پیشین خود نیز بهره می‌برد همچنان به خوبی DenseNet که از اطلاعات تمامی لایه‌های پیشین خود در بلوک استفاده می‌کند عمل نخواهد کرد.

۲- تمامی تصاویر را به اندازه‌ی ۱۲۸x۱۲۸ درآوردم. این اندازه را بر اساس انتخاب کردم. دلیل اینکه لازم است تصاویر یک اندازه باشند، یکی بودن تعداد نورون‌های ورودی است. برای تقسیم داده‌ها ایندکس‌های تصادفی تولید کردم و داده‌ها را همانطور که خواسته شده به ۷۰، ۲۰ و ۱۰ درصد تقسیم کردم.

۳- برای انتقال یادگیری، به سه طریق عمل می‌شود:

۱. مدل از پیش آموزش دیده را به طور کامل دریافت می‌کنیم و در حالیکه تمامی وزن‌های آن ثابت اند، از آن برای پیشبینی استفاده می‌کنیم.
۲. تمام مدل به جز بخش دسته‌بندی آن را دریافت می‌کنیم و از آن تنها به عنوان کاشف ویژگی بهره می‌بریم. در این حالت بخش دسته‌بند را که خودمان طراحی کرده‌ایم به انتهای مدل پایه افزوده و با ثابت کردن وزن‌های مدل پایه، دسته‌بند را آموزش می‌دهیم. در اینجا تنها وزن‌های بخش دسته‌بند مدل بروز می‌شوند.
۳. مدل را بر روی دیتاست خود تنظیم دقیق می‌کنیم؛ به این صورت که تمام مدل به جز بخش دسته‌بند را دریافت کرده و دسته‌بند طراحی شده توسط خودمان را به آن اضافه می‌کنیم. در ابتدا وزن لایه‌های مدل پایه را ثابت کرده و تنها دسته‌بند را آموزش می‌دهیم. حال به جایی رسیده ایم که دسته‌بند وزن‌های اولیه مناسبی دارد. به این دلیل که اصولا لایه‌های پایین‌تر ویژگی‌های بدوی تری را می‌آموزند و همچنین یادگیری آنها بیشتر طول می‌کشد، آنها را ثابت نگه داشته و لایه‌های بالاتر بعلاوه دسته‌بند خود را با درجه یادگیری کم، آموزش می‌دهیم. لازم به ذکر است اگر خواستیم می‌توانیم برخی لایه‌های مدل پایه را که به آنها نیازی نمی‌بینیم حذف کنیم.

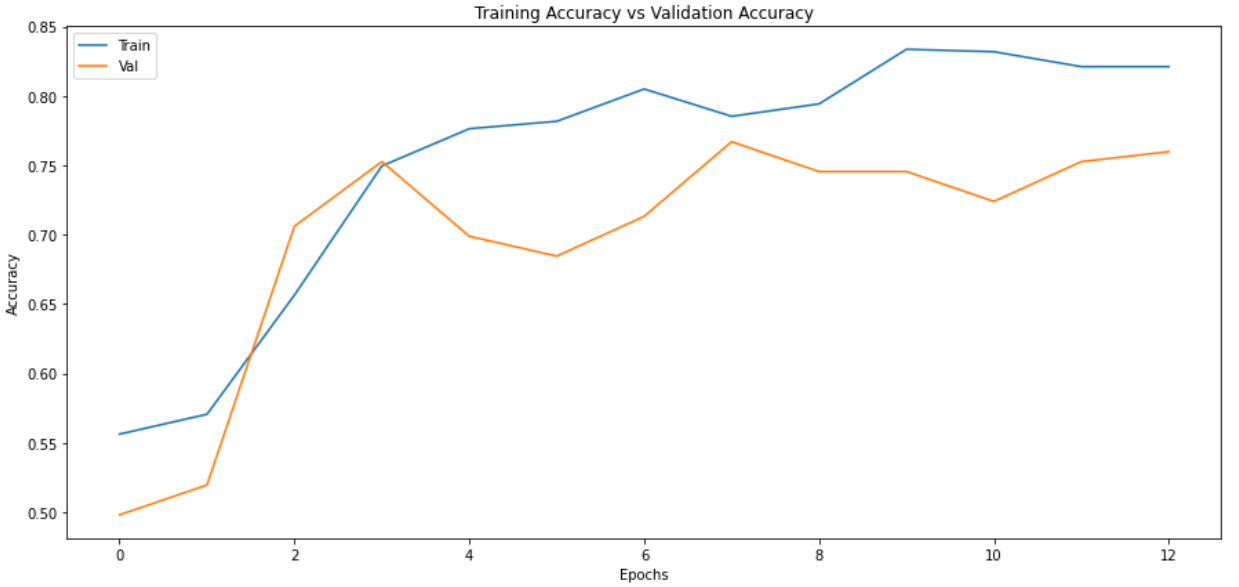
۴- از مدل ResNet101 و DenseNet121 در کتابخانه تنسورفلو استفاده کردم.

مطابق بخش سوم در پاسخ سوال قبلی عمل کردم.

رزنت:

پس از اضافه کردن لایه‌های دسته‌بند به ترتیب با ۵۱۲ نورون، لایه DropOut با ۰,۲، لایه با ۲۵۶ نورون، لایه DropOut با ۰,۲ و ۲ نورون خروجی، نتیجه‌ی آموزش به شکل زیر است:

```
loss: 0.4387 - accuracy: 0.7853 - val_loss: 0.5251 - val_accuracy: 0.7670
```



نتیجه‌ی آزمون:

```
loss: 0.4988 - accuracy: 0.7863
```

	Actual Positive	Actual Negative
Pred. Positive	60	10
Pred. Negative	18	43

تنها وزن‌های دو لایه‌ی انتهایی را از حالت ثابت خارج کردم. تعداد بیشتر لایه‌ها موجب نتیجه‌ی بدتر می‌شد. دلیل آن می‌تواند تعداد داده‌های کم باشد که نمی‌تواند لایه‌ها را به درستی آموزش دهد و در صورت آموزش زیاد دچار بیش برازش خواهیم شد. دو لایه‌ی انتهایی مدل رزنت ۱۱۰ شامل لایه‌های زیر می‌شود:

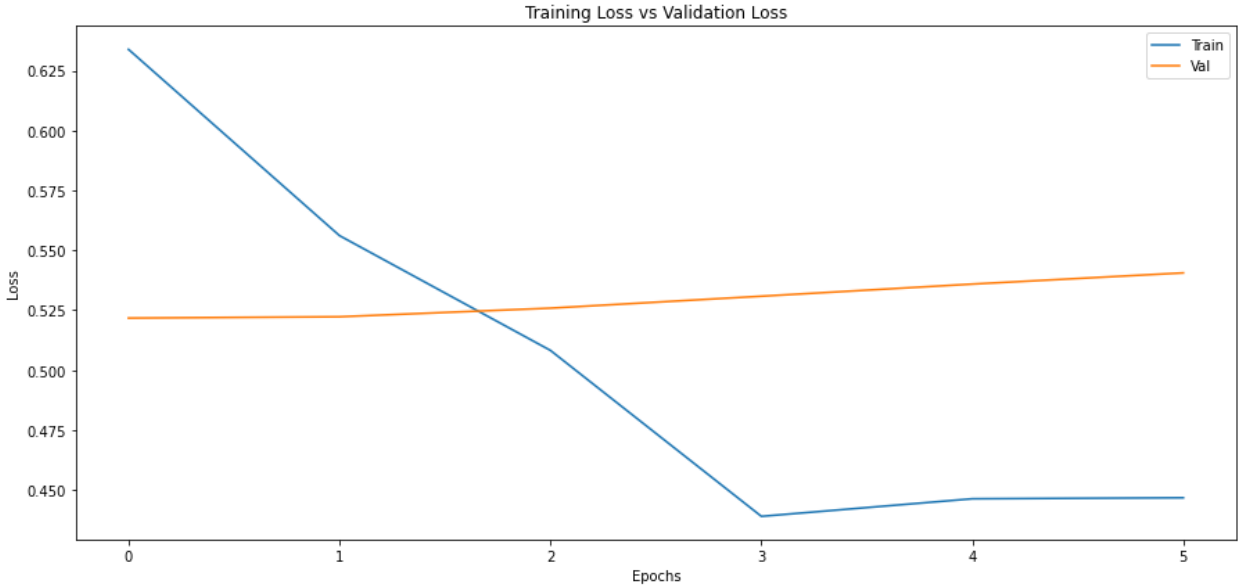
```
conv5_block3_2_conv (Conv2D)    (None, 4, 4, 512)    2359808     conv5_block3_1_relu[0][0]
conv5_block3_2_bn (BatchNormali (None, 4, 4, 512)    2048        conv5_block3_2_conv[0][0]
conv5_block3_2_relu (Activation (None, 4, 4, 512)    0           conv5_block3_2_bn[0][0]
conv5_block3_3_conv (Conv2D)    (None, 4, 4, 2048)   1050624     conv5_block3_2_relu[0][0]
conv5_block3_3_bn (BatchNormali (None, 4, 4, 2048)   8192        conv5_block3_3_conv[0][0]
conv5_block3_add (Add)          (None, 4, 4, 2048)   0           conv5_block2_out[0][0]
                                                                 conv5_block3_3_bn[0][0]
conv5_block3_out (Activation)   (None, 4, 4, 2048)   0           conv5_block3_add[0][0]
```

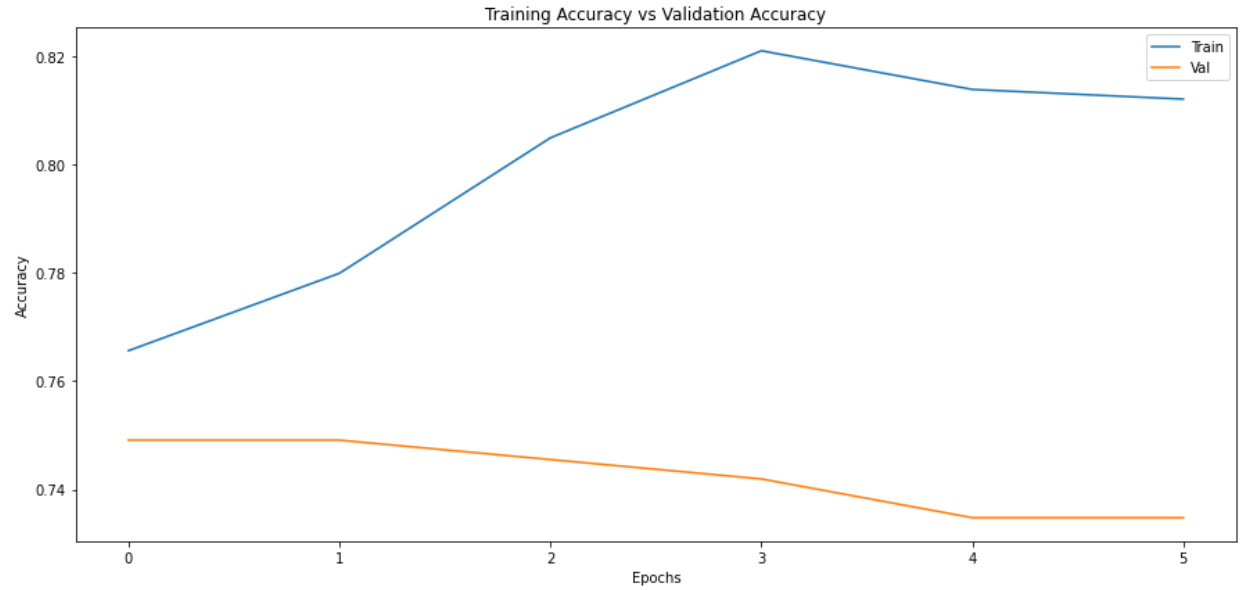
لازم به ذکر است که از callback بر روی دقت مجموعه اعتبارسنجی با ۵ تکرار حداکثر استفاده کردم.

در نهایت، آموزش دوباره با درجه‌ی یادگیری ۰٫۰۰۰۰۰۱:

```
Epoch 1/40
9/9 [==============================] - 10s 387ms/step - loss: 0.6338 - accuracy: 0.7657 - val_loss: 0.5217 - val_accuracy: 0.7491
Epoch 2/40
9/9 [==============================] - 2s 181ms/step - loss: 0.5561 - accuracy: 0.7800 - val_loss: 0.5223 - val_accuracy: 0.7491
Epoch 3/40
9/9 [==============================] - 2s 185ms/step - loss: 0.5083 - accuracy: 0.8050 - val_loss: 0.5259 - val_accuracy: 0.7455
Epoch 4/40
9/9 [==============================] - 2s 185ms/step - loss: 0.4390 - accuracy: 0.8211 - val_loss: 0.5309 - val_accuracy: 0.7419
Epoch 5/40
9/9 [==============================] - 2s 185ms/step - loss: 0.4464 - accuracy: 0.8140 - val_loss: 0.5359 - val_accuracy: 0.7348
Epoch 6/40
9/9 [==============================] - 2s 185ms/step - loss: 0.4468 - accuracy: 0.8122 - val_loss: 0.5406 - val_accuracy: 0.7348
epoch: 5: early stopping
```

قابل مشاهده است که بهبودی نیافته است.





نتیجه‌ی آزمون:

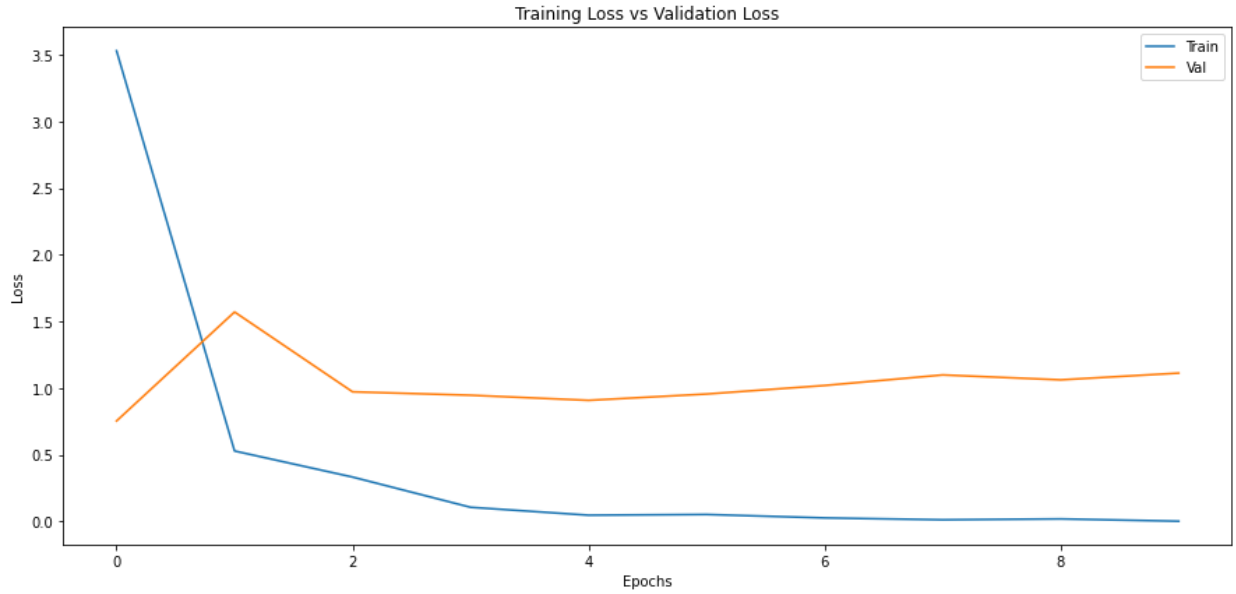
```
loss: 0.4981 - accuracy: 0.7710
```

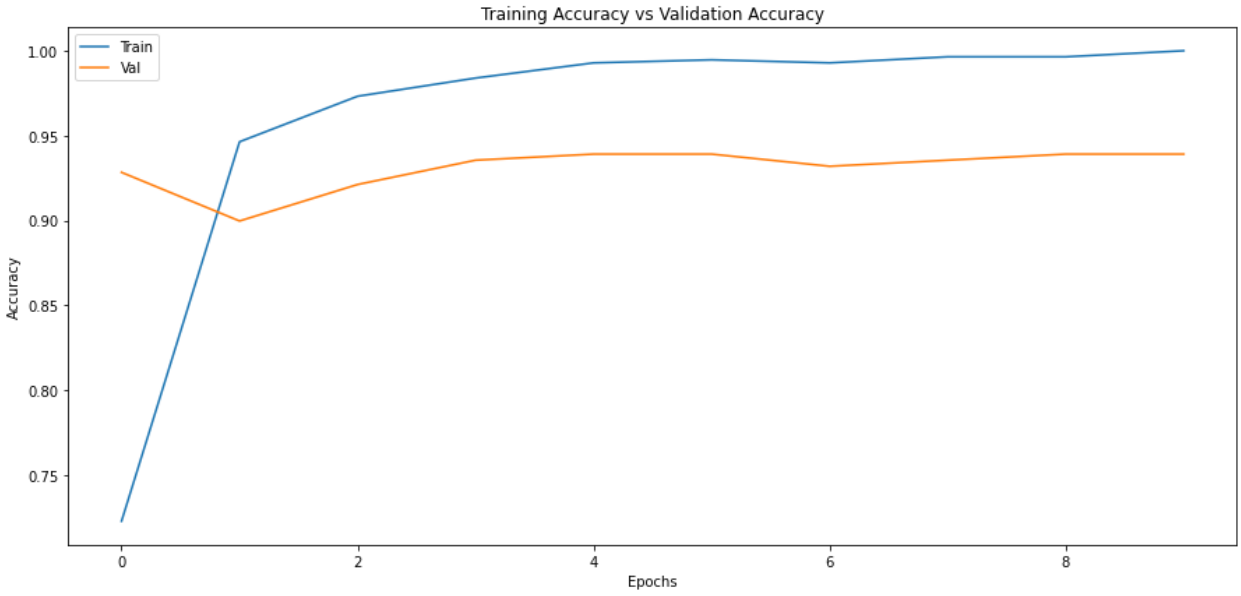
	Actual Positive	Actual Negative
Pred. Positive	58	12
Pred. Negative	18	43

دنس نت:

پس از اضافه کردن لایه‌های مشابه رزنت نتیجه به صورت زیر است:

```
loss: 0.0459 - accuracy: 0.9928 - val_loss: 0.9077 - val_accuracy: 0.9391
```





نتیجه‌ی آزمون:

```
loss: 0.5231 - accuracy: 0.9771
```

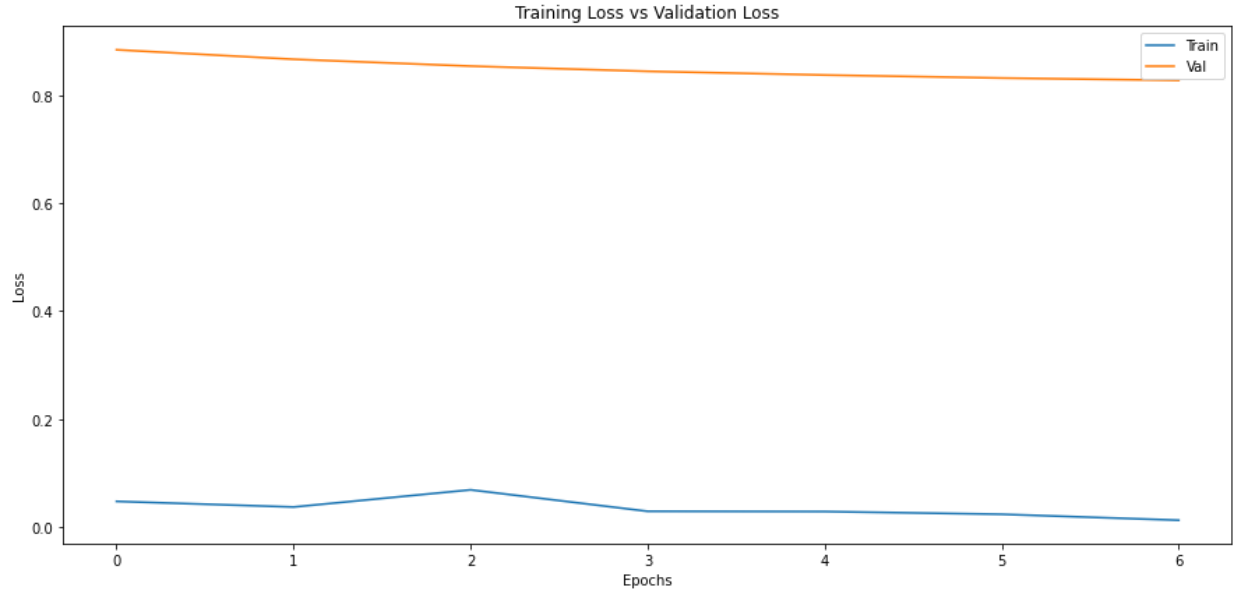
	Actual Positive	Actual Negative
Pred. Positive	69	1
Pred. Negative	2	59

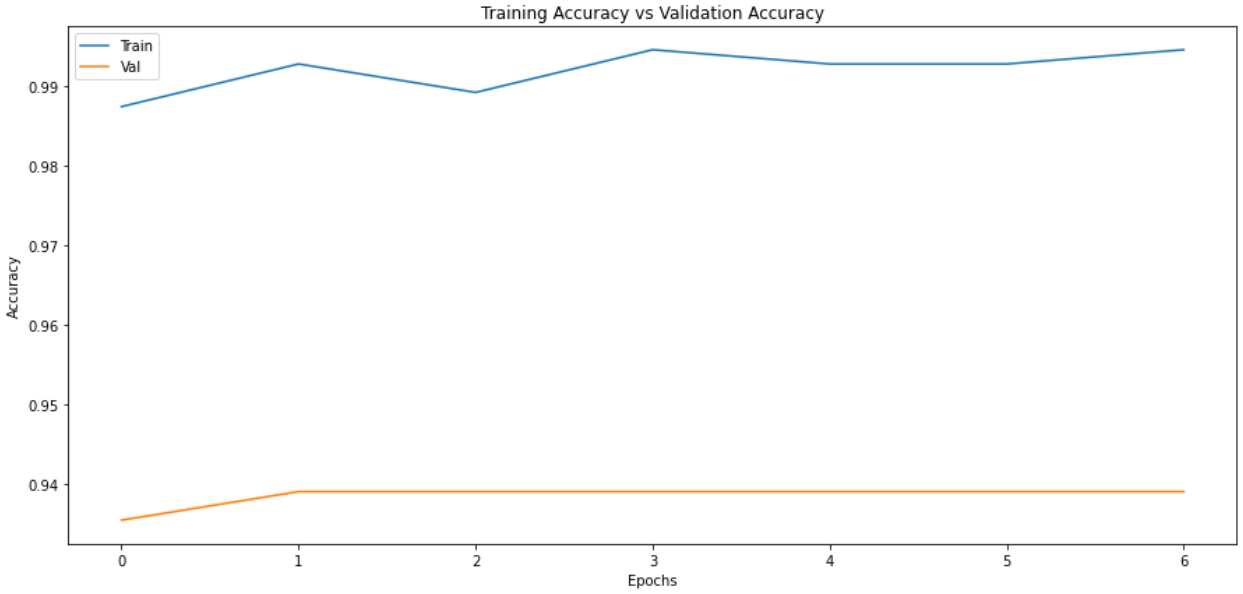
تنها وزن‌های بلوک ۱۶ از بلوک دنس آخر بعلاوه‌ی لایه نرمال‌سازی دسته‌ای و فعالیت Relu را از حالت ثابت خارج کردم.

```
conv5_block16_0_bn (BatchNormal (None, 4, 4, 992)    3968        conv5_block15_concat[0][0]
conv5_block16_0_relu (Activatio (None, 4, 4, 992)    0           conv5_block16_0_bn[0][0]
conv5_block16_1_conv (Conv2D)   (None, 4, 4, 128)    126976      conv5_block16_0_relu[0][0]
conv5_block16_1_bn (BatchNormal (None, 4, 4, 128)    512         conv5_block16_1_conv[0][0]
conv5_block16_1_relu (Activatio (None, 4, 4, 128)    0           conv5_block16_1_bn[0][0]
conv5_block16_2_conv (Conv2D)   (None, 4, 4, 32)     36864       conv5_block16_1_relu[0][0]
conv5_block16_concat (Concatena (None, 4, 4, 1024)   0           conv5_block15_concat[0][0]
                                                                 conv5_block16_2_conv[0][0]
bn (BatchNormalization)         (None, 4, 4, 1024)   4096        conv5_block16_concat[0][0]
relu (Activation)               (None, 4, 4, 1024)   0           bn[0][0]
```

با درجه یادگیری ۰،۰۰۰۰۱ و آموزش دوباره،

```
Epoch 1/40
9/9 [==============================] - 8s 342ms/step - loss: 0.0473 - accuracy: 0.9875 - val_loss: 0.8844 - val_accuracy: 0.9355
Epoch 2/40
9/9 [==============================] - 1s 109ms/step - loss: 0.0369 - accuracy: 0.9928 - val_loss: 0.8669 - val_accuracy: 0.9391
Epoch 3/40
9/9 [==============================] - 1s 109ms/step - loss: 0.0687 - accuracy: 0.9893 - val_loss: 0.8542 - val_accuracy: 0.9391
Epoch 4/40
9/9 [==============================] - 1s 107ms/step - loss: 0.0290 - accuracy: 0.9946 - val_loss: 0.8445 - val_accuracy: 0.9391
Epoch 5/40
9/9 [==============================] - 1s 109ms/step - loss: 0.0285 - accuracy: 0.9928 - val_loss: 0.8376 - val_accuracy: 0.9391
Epoch 6/40
9/9 [==============================] - 1s 108ms/step - loss: 0.0234 - accuracy: 0.9928 - val_loss: 0.8321 - val_accuracy: 0.9391
Epoch 7/40
9/9 [==============================] - 1s 107ms/step - loss: 0.0124 - accuracy: 0.9946 - val_loss: 0.8277 - val_accuracy: 0.9391
epoch: 6: early stopping
```





نتیجه‌ی آزمون:

```
loss: 0.4927 - accuracy: 0.9771
```

	Actual Positive	Actual Negative
Pred. Positive	69	1
Pred. Negative	2	59

همانطور که انتظار می‌رود نتایج دنس نت بهتر از رزنت است. با این حال هر دو پس از آموزش دسته‌بند، و در مرحله‌ی تنظیم دقیق، بهبود نمی‌یابند. بیش برازش نیز در این مرحله کاملا مشهود است.