|  |  |  |
| --- | --- | --- |
| **پرسش‌ ۱** | **نام دانشجو** | سیدمحسن ایزدی اونجی |
| **شماره دانشجویی** | 810101317 |
| **پرسش 2** | **نام دانشجو** | سید محمد مهدی رضوی |
| **شماره دانشجویی** | 810102155 |
|  | **مهلت ارسال پاسخ** | **-** |

|  |  |  |
| --- | --- | --- |
|  | **به نام خدا**  **دانشگاه تهران**  **دانشکده‌ مهندسی برق و کامپیوتر** |  |
| **درس شبکه‌های عصبی و یادگیری عمیق**  **تمرین اول** | | |

**فهرست**

[**پرسش 1**. **تشخیص آلزایمر با استفاده از تصویر برداری مغزی (ADNI)** 1](#_Toc165297428)

[۱-۱. پیش پردازش تصاویر 2](#_Toc165297429)

[۱-2. پیاده سازی 2](#_Toc165297430)

[۱-3. معماری منتخب مقاله 4](#_Toc165297431)

[۱-4. معماری آزمایشی 1 5](#_Toc165297432)

[۱-5. معماری آزمایشی 1 6](#_Toc165297433)

[۱-6. مقایسه معماری‌ها 6](#_Toc165297434)

[۱-7. مقایسه معماری‌ها 7](#_Toc165297435)

[**پرسش ۲–بررسی تاثیرافزایش داده برعملکردشبکه‌های کانولوشنی** 10](#_Toc165297436)

[۱-۲. **پیش‌پردازش تصاویر** 10](#_Toc165297437)

[. 2-۲. پیاده‌سازی 11](#_Toc165297438)

[2-3: ResNet 12](#_Toc165297439)

[2-4: مدل VGG16 14](#_Toc165297440)

[. 2-۲. تحلیل و نتیجه‌گیری 16](#_Toc165297441)

**شکل‌ها**

شکل 1 توزیع آماری دادگان قبل و بعد از داده افزایی 1

شکل 2 نمونه‌هایی تصادفی از داده‌ها پس از داده افزایی 1

شکل 3 توزیع آماری مجموعه دادگان آموزش، آزمون و Validation 3

شکل 4 پیاده سازی معماری منتخب 3

شکل 5 نتایج دادگان تست روی معماری منتخب 4

شکل 6 دقت مجموعه Validation در معماری منتخب 4

شکل 7 دقت مجموعه داده آموزش در معماری منتخب 4

شکل 8 نمودار ROC و AUC در معماری منتخب 4

شکل 9 نمودار Precision-Recall معماری منتخب 4

شکل 10 نمودار شکل 10 مقاله مرجع 5

شکل 11 نمودارهای معماری آزمایشی شماره 1 5

شکل 12 نمودارهای معماری آزمایشی شماره 1 6

شکل 13 نمودارهای مربوط به تقسیم بندی مجموعه داده 7

شکل 14 نمودارهای مربوط به اثر Dropout 8

شکل 15 نمودارهای مربوط به اثر Initializer 9

شکل16 : تعداد تصاویر با برچسب سگ و گربه بعد از عمل اقزودن داده 10

شکل 17 : تعداد تصاویر با برچسب سگ و گربه قبل از عمل افزودن داده 10

شکل 18 : نمودار دقت و تابع ضرر قبل از افزودن داده - مدل رزنت 12

شکل 19 : دقت و تابع ضرر بعد از افزودن داده - مدل رزنت 12

شکل 20 : دقت مجموعه آموزشی ، ارزیابی و تست - قبل از افزودنگی داده - مدل رزنت 13

شکل 21 : دقت مجموعه آموزشی ، ارزیابی و تست - بعد از افزونگی داده - مدل رزنت 13

شکل 22 : نمودار دقت و تابع ضرر - قبل از افزونگی داده - مدل وی‌جی‌جی 14

شکل 23 : نمودار دقت و تابع ضرر - بعد از افزونگی داده - مدل وی‌جی‌جی 14

شکل 24 : دقت تست قبل از افزونگی داده‌ها- مدل وی‌جی‌جی 15

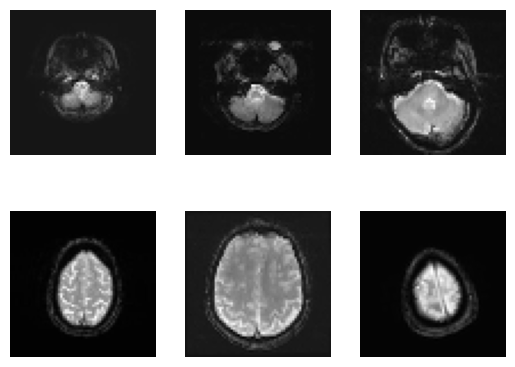
شکل 25 : دقت مجموعه آموزشی ، ارزیابی و تست بعد از افزونگی داده‌ها - مدل وی‌جی‌جی 15

**جدول‌ها**

[جدول 1 پارامترهای آموزش معماری‌های ارائه شده 2](#_Toc165297467)

# **پرسش 1**. **تشخیص آلزایمر با استفاده از تصویر برداری مغزی (ADNI)**

|  |  |
| --- | --- |
|  |  |
| شکل 1 توزیع آماری دادگان قبل و بعد از داده افزایی | |



شکل 2 نمونه‌هایی تصادفی از داده‌ها پس از داده افزایی

۱-۱. پیش پردازش تصاویر

در مرحله پیش پردازش طبق متن تمرین داده ها را به منظور تسریع فرآیند آموزش با روش Min-Max نرمال کرده‌ایم و همچنین طبق نیاز معماری‌های پیاده شده در مقاله اندازه تصاویر را به 64x64 تغییر می‌دهیم.

پس از اعمال تغییرات بیان شده از تعداد 1654 تصویر، پس از داده افزایی 8270 عکس تولید شده است. توزیع آماری تصاویر هر دسته در شکل 1 مشاهده می شود. توجه شود که در این تمرین به نمونه‌های دارای بیماری آلزایمر برچسب "1" و نمونه های دارای اختلال خفیف (MCI) برچسب "0" زده شده است. همانطور که در تصویر مشاهده می‌شود توزیع آماری پس از داده افزایی مشابه قبل از آن خواهد بود. از مجموعه جدید دادگان تعداد 6 تصویر نیز به طور تصادفی در شکل 2 رسم شده است.

۱-2. پیاده سازی

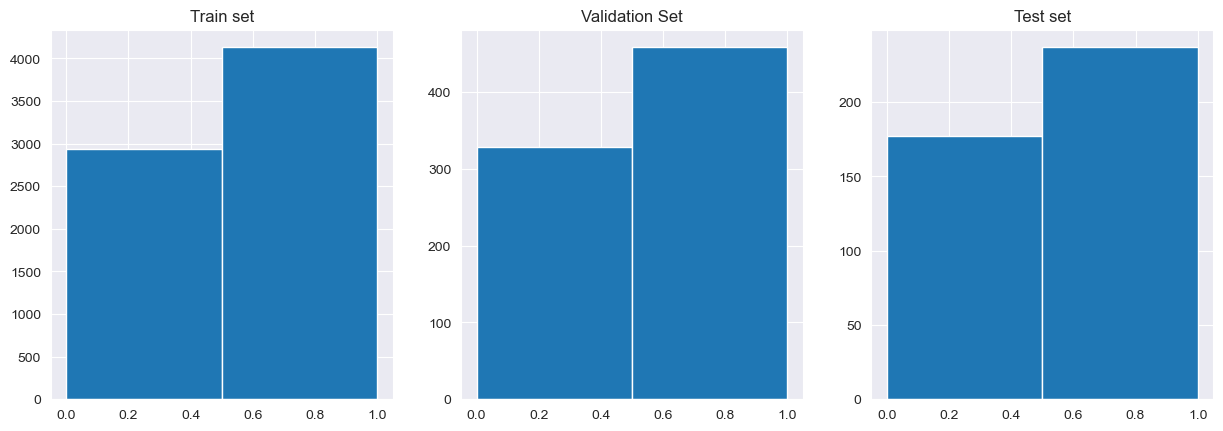
در این بخش سه معماری بیان شده در مقاله را پیاده‌سازی می‌کنیم. به عنوان تابع هزینه از categorical crossentropy و برای بهینه‌ساز از Adam استفاده می‌کنیم. همچنین از Glorot initialization به دلیل جلوگیری از شروع تابع فعال‌ساز نورون‌ها در حالت اشباع استفاده می‌شود که به طور چشمگیری سرعت همگرایی مدل را افزایش و دقت آنرا بهبود می‌دهد. در تمامی نتیجه‌گیری های انجام شده نرخ یادگیری مطابق مقاله بوده مگر در مواردی که ذکر شود.

جدول 1 پارامترهای آموزش معماری‌های ارائه شده

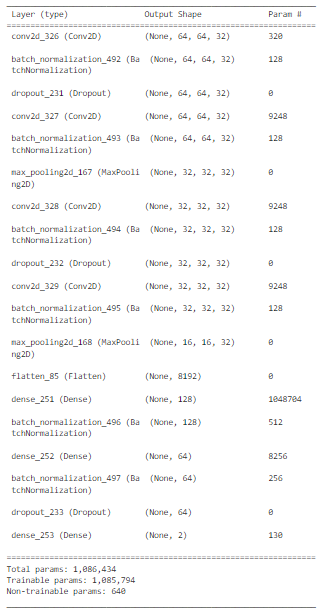
|  |  |  |
| --- | --- | --- |
|  | **مقدار** | **توضیح** |
| **Bach Size** | 256 | - |
| **Num. of Epochs** | 20 | - |
| **Learning Rate** | 0.1 | - |
| **Test size** | % 10 | 10 درصد از مجموعه داده‌ها |
| **Validation Size** | % 10 | 10 درصد از مجموعه تست |
| **Initializer** | Glorot | - |
| **Optimizer** | Adam | - |
| **Loss Function** | Cross Entropy Categorical | - |

در ادامه نتایج آموزش سه معماری بیان شده با پارامترهای بیان شده در جدول 1 را برای هر معماری رسم می‌کنیم.

**نکته: طبق متن مقاله پس از هر لایه Activation یک لایه Bach Normalization قرار داده شده است.**

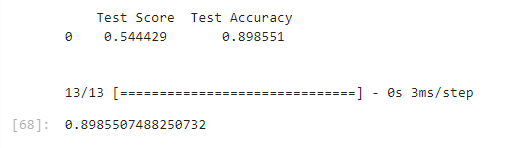


شکل 3 توزیع آماری مجموعه دادگان آموزش، آزمون و Validation



شکل 4 پیاده سازی معماری منتخب

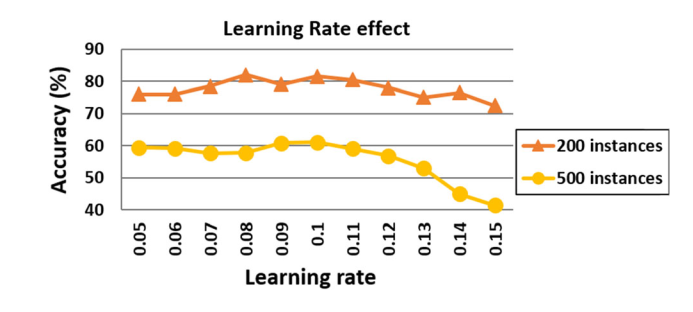
۱-3. معماری منتخب مقاله



شکل 5 نتایج دادگان تست روی معماری منتخب

|  |  |
| --- | --- |
| شکل 6 دقت مجموعه Validation در معماری منتخب | شکل 7 دقت مجموعه داده آموزش در معماری منتخب |
| شکل 8 نمودار ROC و AUC در معماری منتخب | شکل 9 نمودار Precision-Recall معماری منتخب |

در این معماری مشاهده می‌شود که نمودار ROC متمایل به گوشه بالا سمت چپ است و از خط قطری (خط چین آبی) فاصله گرفته است همچنین شاخص AUC نیز نزدیک یک است که نتیجه مطلوبی است. اما دقت روی دادگان تست به نسبت آنچه در مقاله گزارش شده است کمتر است. دلیل این مورد با استناد به تصویر شماره 9 مقاله که در شکل 10 این گزارش هم آورده شده است می‌تواند کم بودن نمونه‌ها باشد.



شکل 10 نمودار شکل 10 مقاله مرجع

۱-4. معماری آزمایشی 1

|  |  |
| --- | --- |
|  | |
|  |  |
|  |  |
| شکل 11 نمودارهای معماری آزمایشی شماره 1 | |

۱-5. معماری آزمایشی 1

|  |  |
| --- | --- |
|  | |
|  |  |
|  |  |
| شکل 12 نمودارهای معماری آزمایشی شماره 1 | |

۱-6. مقایسه معماری‌ها

با مقایسه نمودارهای حاصل شده واضح است که معماری منتخب چه از نظر دقت و چه از نظر شاخص‌های دیگر مانند نمودار ROC عملکرد بهتری داشته است. بین معماری های 1 و 2 هم معماری 1 به نسبت از نظر نمودار ROC عملکرد بسیار بهتری داشته است.

همچنین از نظر تعداد پارامترها، مدل پیشنهادی مقاله نسبت به مدل آزمون 2 تعداد بسیار کمتری پارامتر داشته و زمان آموزش آن نیز در عین دقت بهتر، کمتر است.

۱-7. مقایسه معماری‌ها

در انتها اثر پارامترهای آموزش بر روی شبکه را بررسی می کنیم.

|  |  |
| --- | --- |
| Test Size = 0.3 | Test Size = 0.5 |
|  |  |
|  |  |
|  |  |
| شکل 13 نمودارهای مربوط به تقسیم بندی مجموعه داده | |

همانطور که مشاهده می‌شود دقت در تقسیم 0.5 نسبت به 0.3 از 82.6 به 83.8 درصد افزایش یافته ولی در نمودار ROC تغییر چشمگیری دیده نمی‌شود.

|  |  |
| --- | --- |
| With Dropout | Without Dropout |
|  |  |
|  |  |
|  |  |
| شکل 14 نمودارهای مربوط به اثر Dropout | |

اثر Dropout در نمودارهای رسم شده در شکل 14 مشاهده می‌شود. با استفاده از Dropout دقت مدل و نمودار ROC و شاخص AUC به طور قابل توجهی افزایش پیدا کرده است.

|  |  |
| --- | --- |
| Zeros Initializer | Ones Initializer |
|  |  |
|  |  |
| شکل 15 نمودارهای مربوط به اثر Initializer | |

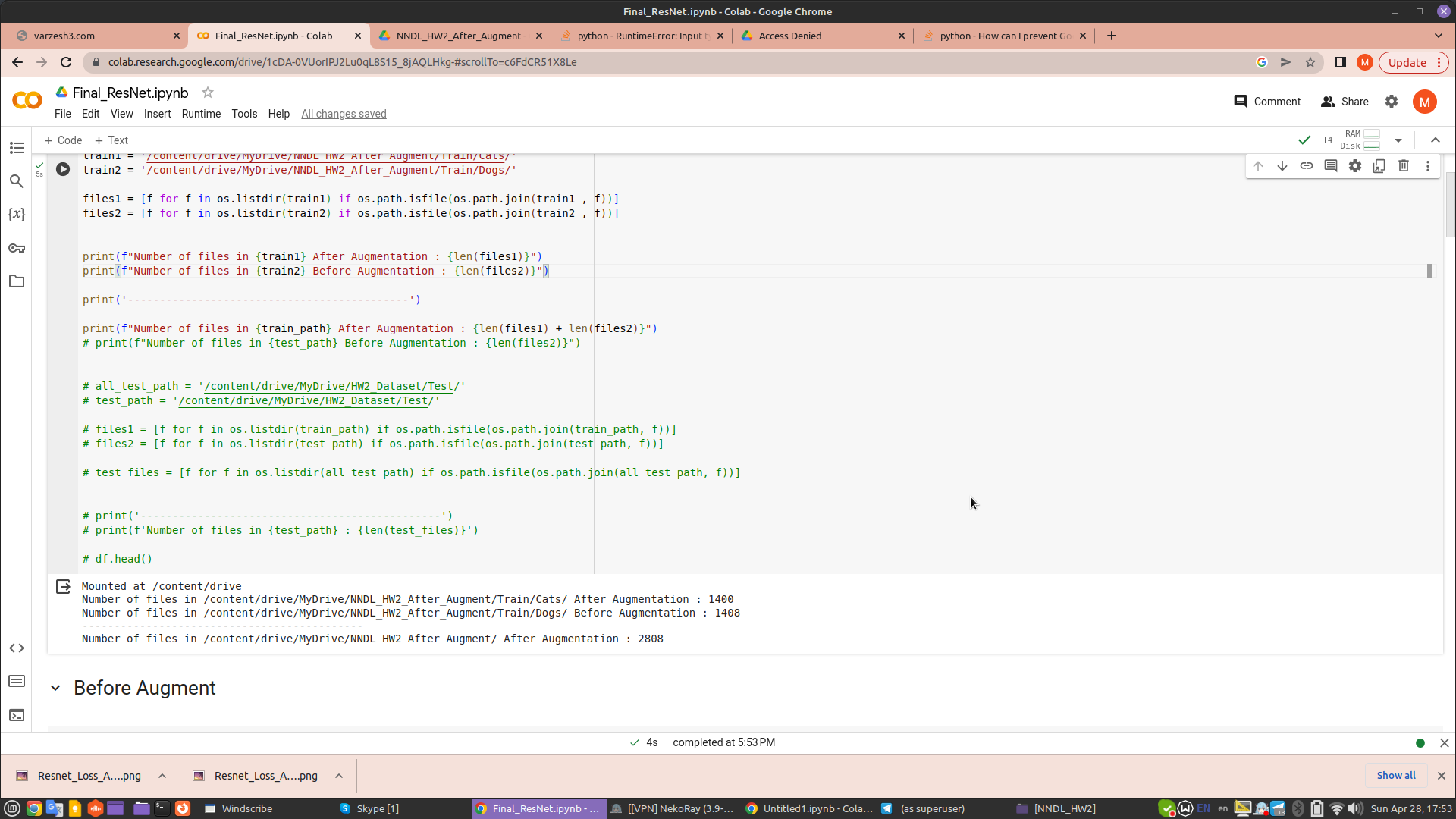
در این بخش برای مقایسه اثر Initializer ها نمودارهای مربوط به Zeros و Ones را در شکل 15 رسم کردیم. همانطور که از مقایسه نمودارها با نمودار مربوط به Glorot مشاهده می‌شود استفاده از Glorot اثر چشمگیری در افزایش دقت مدل دارد.

# **پرسش ۲–بررسی تاثیرافزایش داده برعملکردشبکه‌های کانولوشنی**

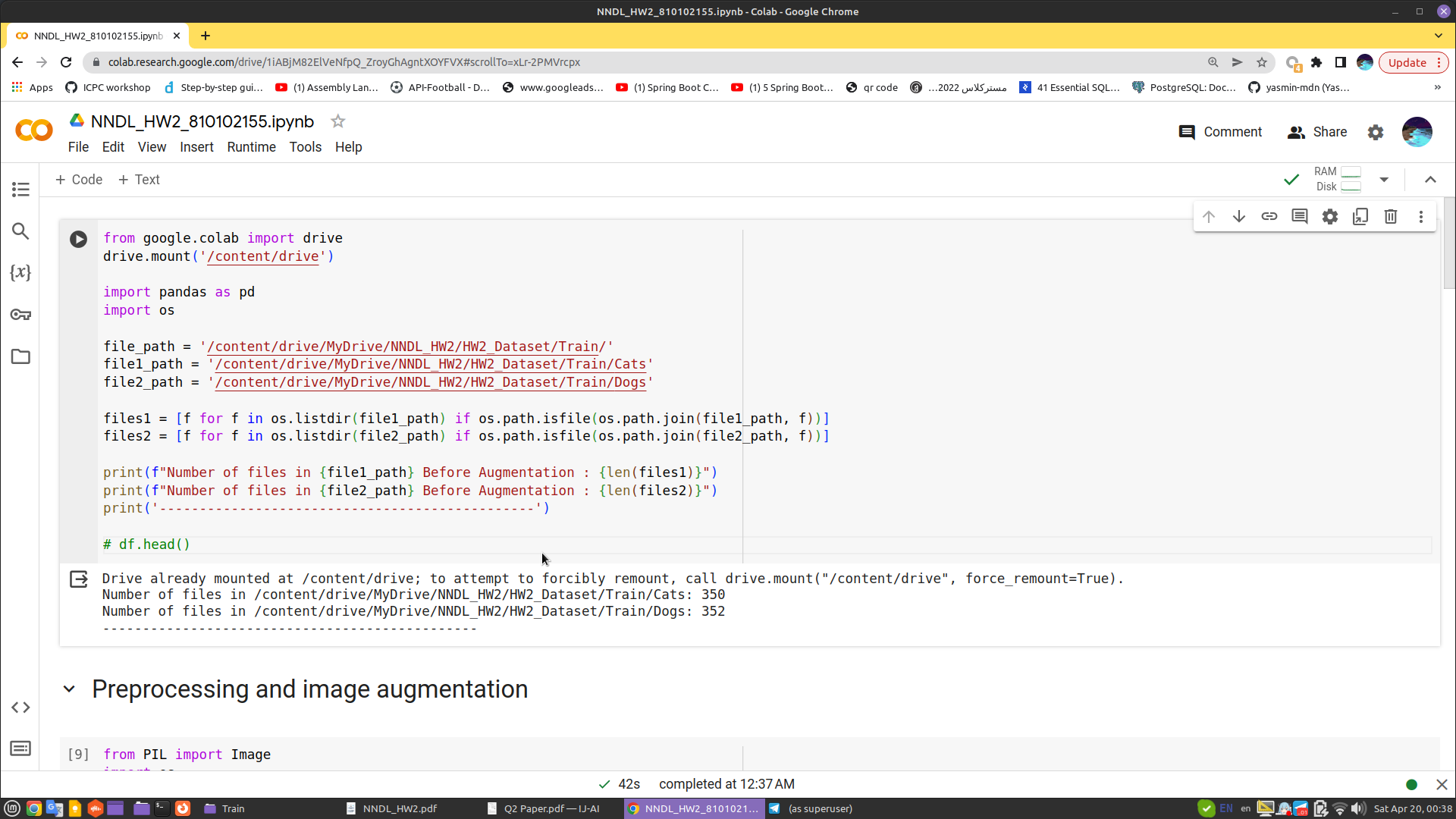
## ۱-۲. **پیش‌پردازش تصاویر**

با استفاده از 3 عملکرد ذکرشده در مقاله به فرآیند تولید عکس برای بهبود عملکرد مدل پرداخته‌ایم.

این عملیات شامل چرخش افقی کامل ، بزرگنمایی یا تغییر اندازه تصویر و چرخش با کمی زاویه دادن می‌باشد.



شکل16 : تعداد تصاویر با برچسب سگ و گربه بعد از عمل اقزودن داده



شکل 17 : تعداد تصاویر با برچسب سگ و گربه قبل از عمل افزودن داده

## . 2-۲. پیاده‌سازی

برای پیاده‌سازی مدل‌های زیر حقیقتا در ابتدای کار با مشکلات زیادی رودرو بودیم . عدم آشنایی کامل با اجرای کد بر روی واحد گرافیکی در گوگل کولب کمی آزاردهنده بود. ابتدا کدهایمان بر روی CPU قرارمی‌دادیم و تا صبح می‌گذاشتیم اجرا شود که معمولا حاصلی نداشت.

در نهایت از مدل کراس برای پیاده‌سازی هر دو مدل ذکرشده در مقاله استفاده کردیم. اما متاسفانه برای مدل رزنت به هیچ عنوان مدل به خوبی نمی‌توانست آموزش ببیند . طبیعتا من نتوانستم به خوبی شبکه رزنت را مدیریت کنم. کراس برای شبکه وی‌جی‌جی دقت‌‌های قابل‌قبولی را نمایش می‌داد.

بنابراین گزارش بنده برای مدل وی‌جی‌جی از کراس و مدل رزنت از پایتورچ می‌باشد.

برای پایتورچ به این صورت از فرامتغیرها استفاده کردم :

optimizer = optim.SGD(model.parameters(), lr=learning\_rate ,momentum=momentum)

برای فریزکردن پارامترها در ابتدای آموزش از متغیر زیر کمک می‌گیریم :

for param in model.parameters():

        param.requires\_grad = False

train\_loader=DataLoader(train\_dataset, batch\_size=batch\_size ,shuffle=True)

val\_loader = DataLoader(val\_dataset, batch\_size=batch\_size, shuffle=False)

test\_loader = DataLoader(test\_dataset, batch\_size=batch\_size,shuffle=False)

برای مدل کراس به شرح زیر است :

برای فریز کردن وزن‌ها در مدل کراس از متغیر زیر استفاده خواهیم کرد :

# Freeze all layers except the new FC layers

for layer in base\_model.layers:

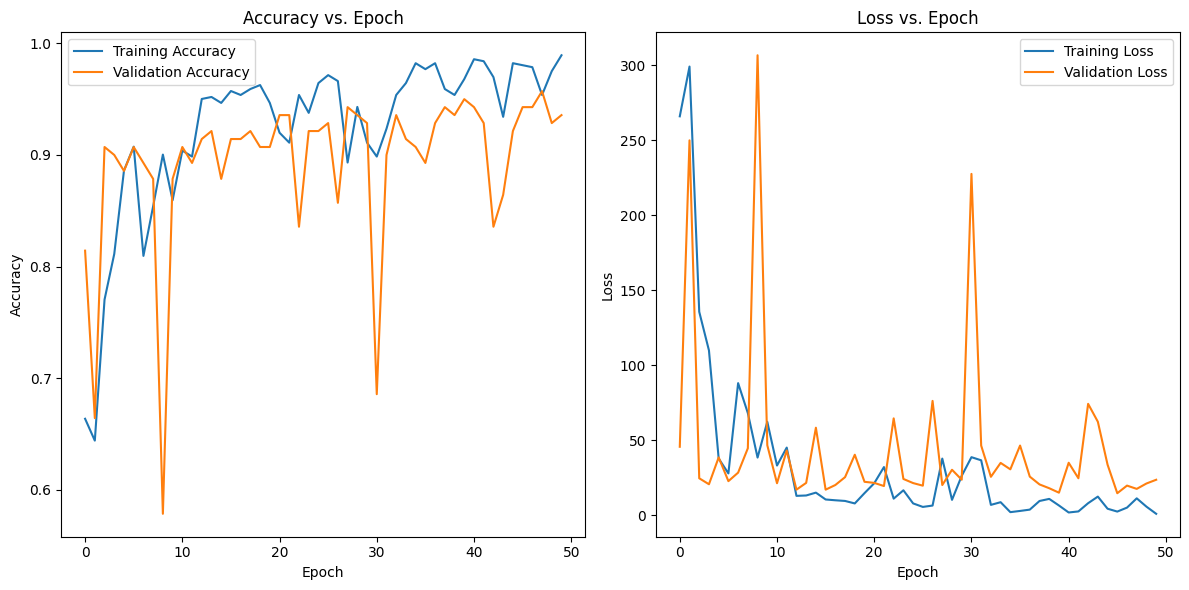
    layer.trainable = False

نرخ یادگیری و مومنتوم را به این صورت به مدل استوکستیک گرادیان دیسنت معرفی می‌کنیم:

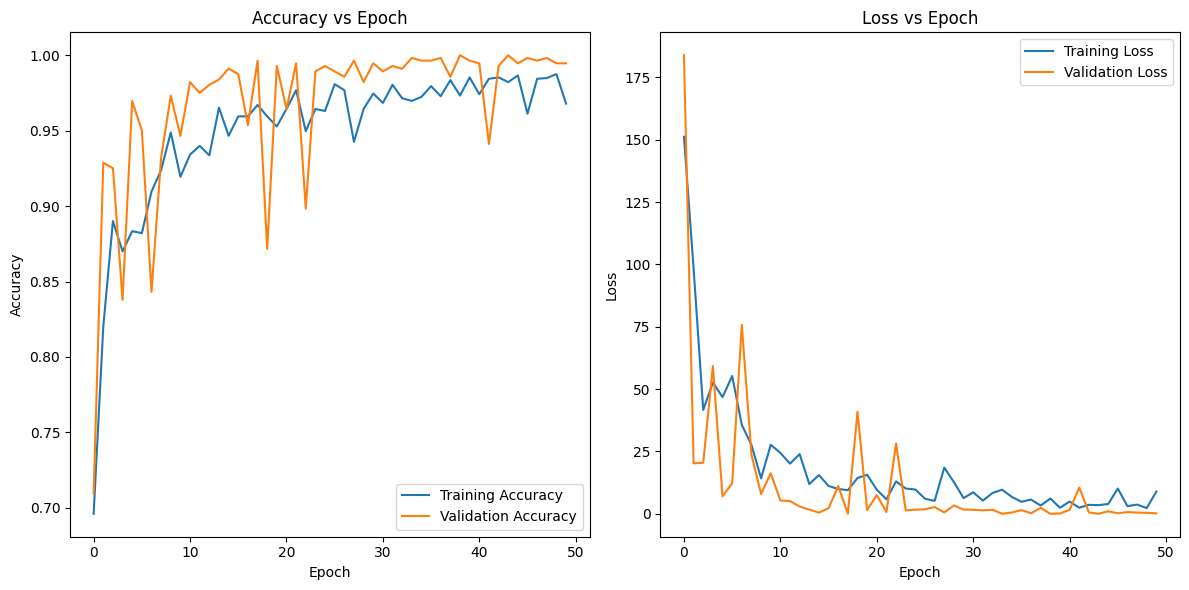
# Compile the model

opt = SGD(learning\_rate=initial\_lr, momentum=momentum)

## 2-3: ResNet



شکل 18 : نمودار دقت و تابع ضرر قبل از افزودن داده - مدل رزنت



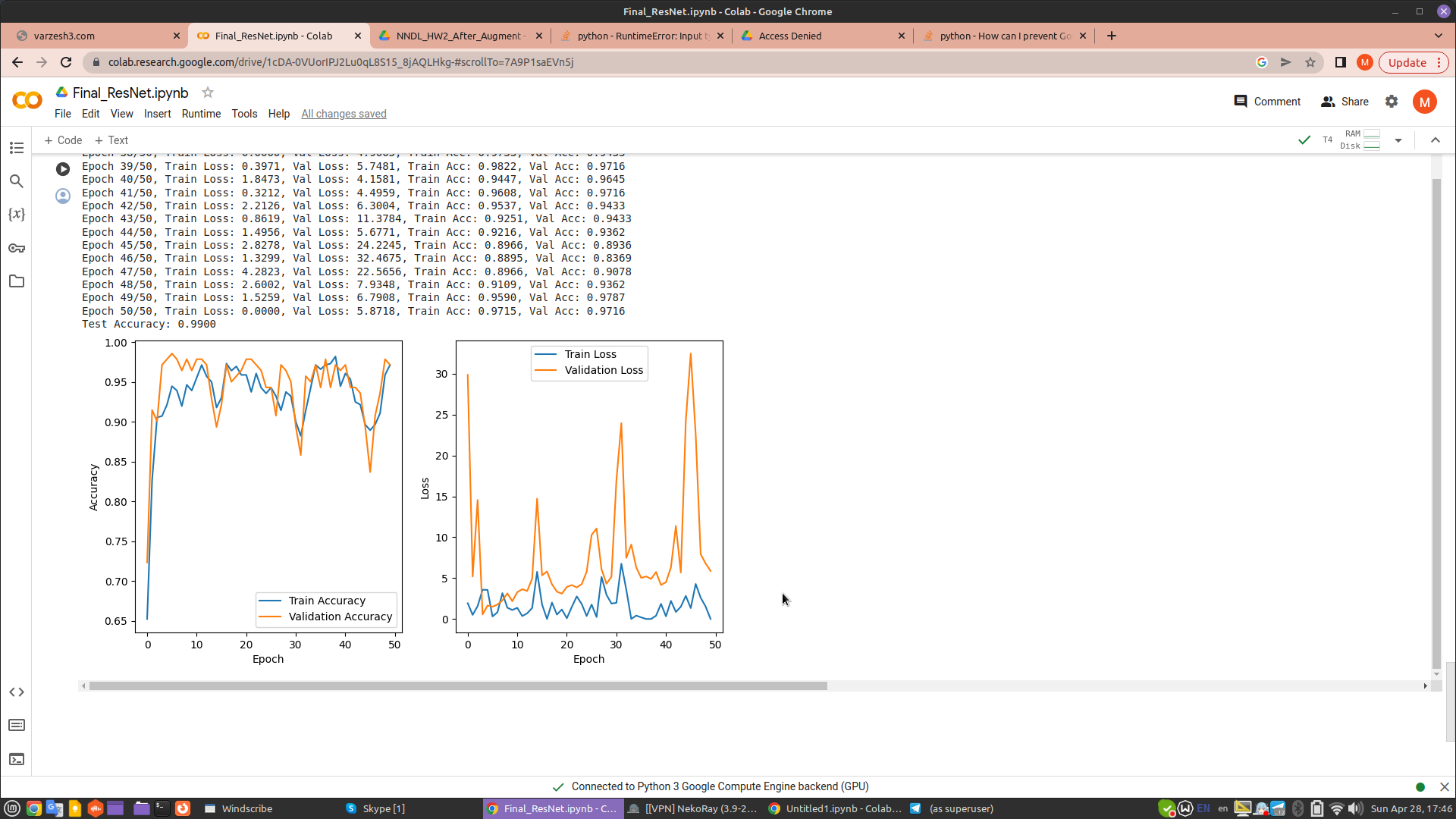
شکل 19 : دقت و تابع ضرر بعد از افزودن داده - مدل رزنت

همانطور که از تصاویر نیز مشخص است ، تاثیر افزایش داده‌ها بر روی عملکرد شبکه‌عصبی کانولوشنی ما مثبت بوده است ، اگرچه هر دو مدل فوق (قبل و بعد افزونگی) از یک الگوی کاهش تابع ضرر پیروی می‌کنند ، اما به مراتب ضرر در مدل بعد از افزونگی کمتر است.(مقیاس نمودار عمودی در قبل از افزونگی بیشتر است.)

همچنین ضرر مجموعه ارزیابی با نوسان کمتری نسبت به حالت قبل از افزونگی تغییر می‌کند.

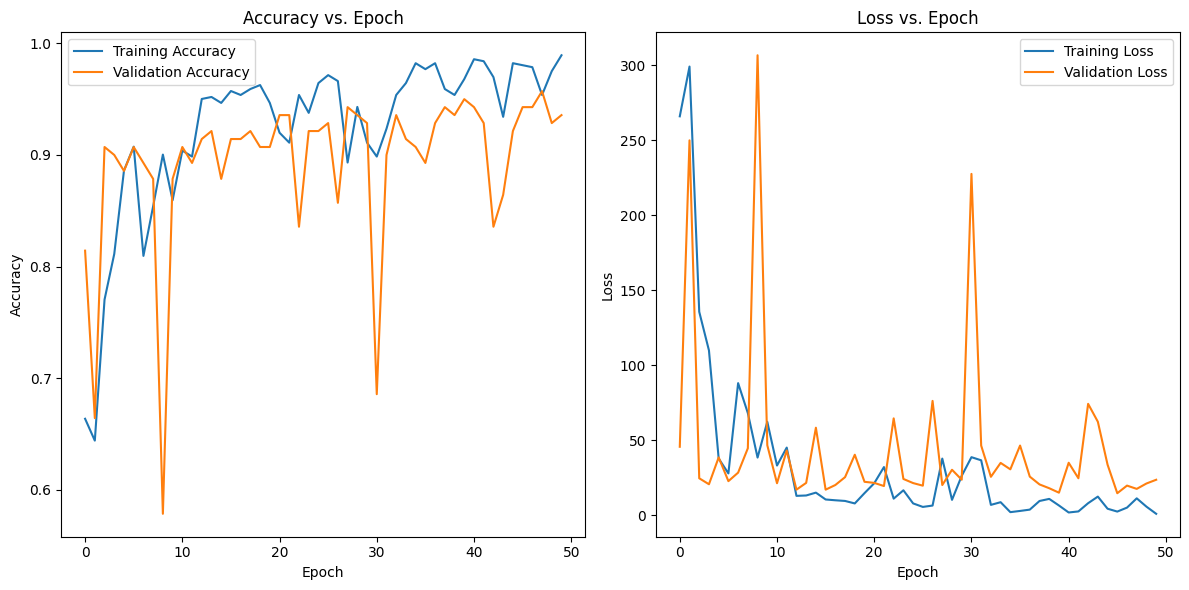


شکل 20 : دقت مجموعه آموزشی ، ارزیابی و تست - قبل از افزودنگی داده - مدل رزنت

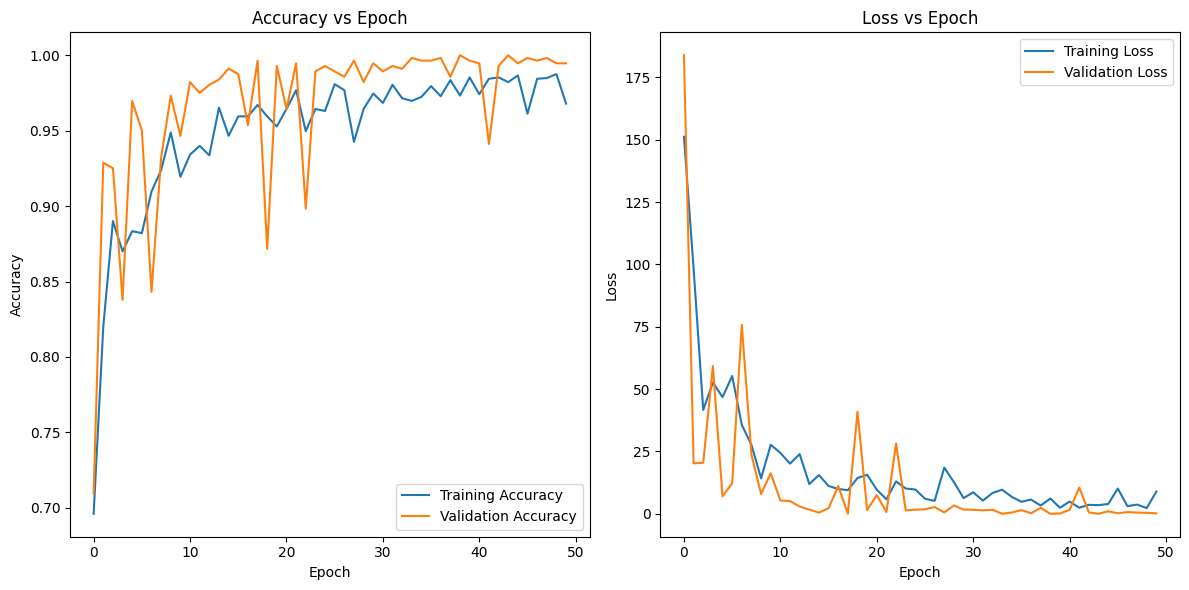


شکل 21 : دقت مجموعه آموزشی ، ارزیابی و تست - بعد از افزونگی داده - مدل رزنت

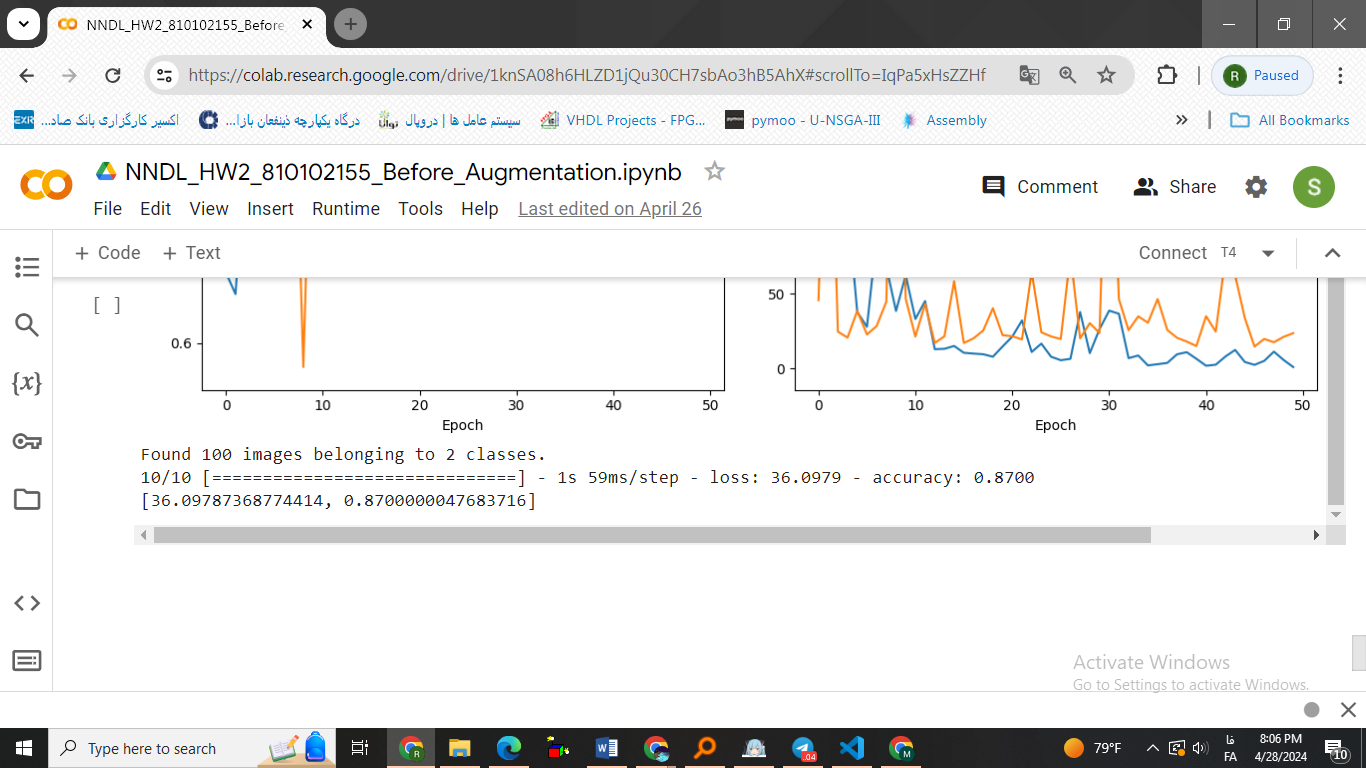
## 2-4: مدل VGG16



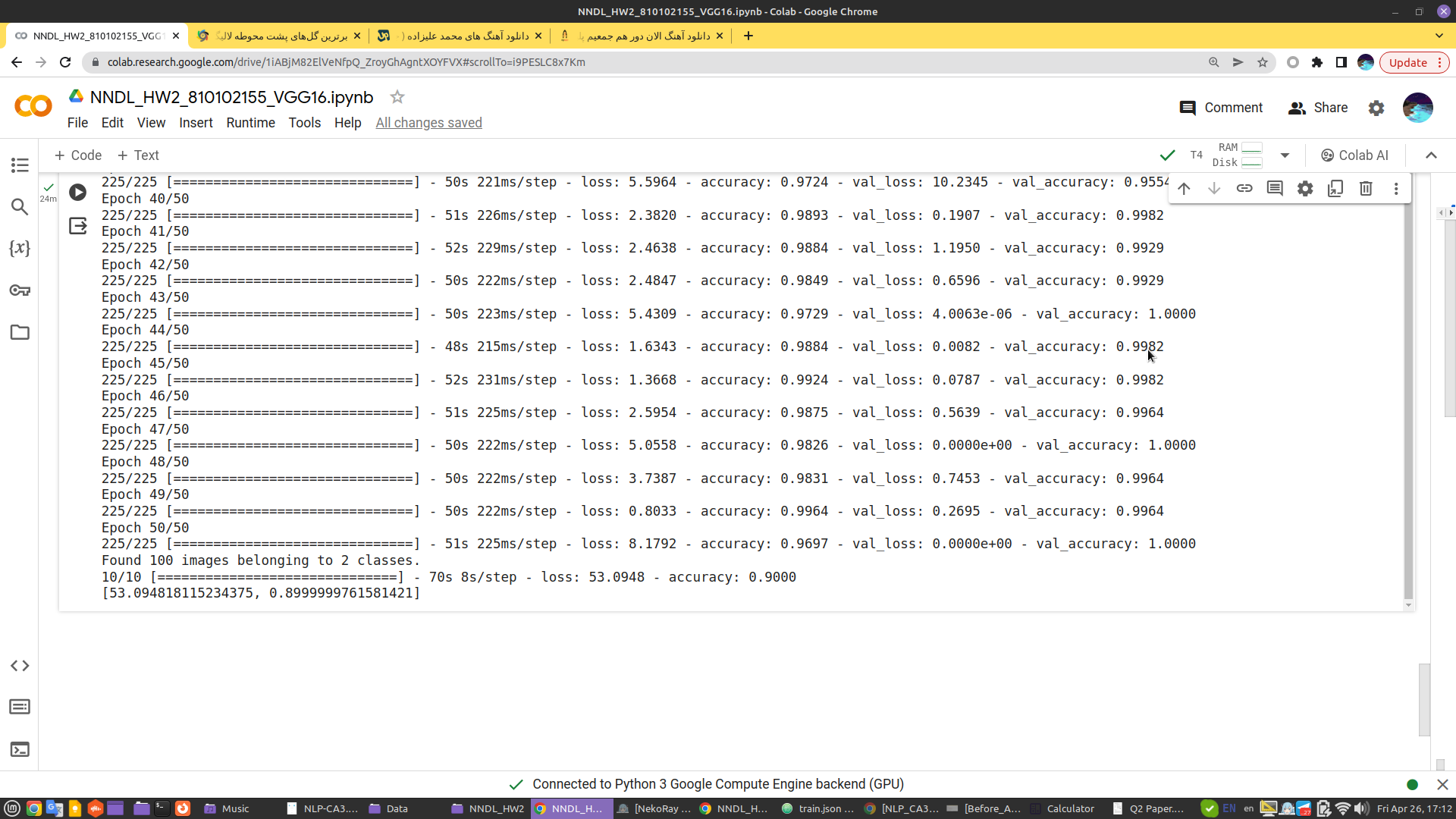
شکل 22 : نمودار دقت و تابع ضرر - قبل از افزونگی داده - مدل وی‌جی‌جی



شکل 23 : نمودار دقت و تابع ضرر - بعد از افزونگی داده - مدل وی‌جی‌جی



شکل 24 : دقت تست قبل از افزونگی داده‌ها- مدل وی‌جی‌جی



شکل 25 : دقت مجموعه آموزشی ، ارزیابی و تست بعد از افزونگی داده‌ها - مدل وی‌جی‌جی

## . 2-۲. تحلیل و نتیجه‌گیری

جدول 1 : دقت‌های مدل‌های ذکرشده

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ResNet Model | | VGG16 Model | |  |
| With Augmentation | Without Augmentation | With Augmentation | Without Augmentation |  |
| 97% | 96% | 96% | 98% | Train |
| 97% | 96% | 100% | 93% | Validation |
| 99% | 93% | 90% | 87% | Test |

به نظرمی‌رسد که بهبود در میزان دقت مدل توسط مدل رزنت به مراتب بیشتر از مدل وی‌جی‌جی بوده‌است.

این مساله از آنجا می‌توان با اطمینان بالاتری به آن اشاره کرد که میزان تعداد دفعاتی که در آزمایش‌های تیم‌ما ، مدل وی‌جی‌جی در **قبل و بعد افزونگی داده هیچ تغییری در میزان دقت تست مشاهده نمی‌شود** ، بالا می‌باشد.

اما مدل رزنت تاثیر قابل توجهی در میزان بهبود دقت حالت تست در آزمایش‌های ما مشاهده‌می‌شود.

حال بهتر است که به تاثیر معماری این دو شبکه در میزان جواب‌های فوق بپردازیم :

مدل وی‌جی‌جی اساسا مدل ساده‌تری به نسبت مدل رزنت هست که این سادگی حتی منجر به ایجاد گرادیان نزدیک به صفر در این شبکه خواهد شد در بسیاری از موارد.

تفاوت در عملکرد هنگام اضافه کردن داده برای آموزش مدل‌های ResNet و VGG16:

ResNet با داشتن اتصالات باقیمانده، آموزش شبکه‌های عمیق را به صورت آسانتر ممکن می‌کند. اضافه کردن داده برای بهبود تعمیم و عملکرد مدل‌های ResNet، به خصوص هنگام آموزش بر روی مجموعه داده‌های بزرگ، مفید است.

رزنت به شبکه این قابلیت را اضافه خواهدکرد که با اضافه‌کردن اتصالات پرش که از یک یا چند لایه عبور می‌کند به گرادیان‌ها این اجازه را می‌دهد که راحت‌تر در شبکه جریان‌پیدا کنند . این راه‌حل کمک‌می‌کند که شبکه به مشکل ناپدیدشدن گرادیان‌ها دچار نشود و ساده‌تر با لایه‌های بیشتر آموزش ببیند.