به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

درس شبکه‌های عصبی و یادگیری عمیق

تمرین پنجم

امیرعباس رضا سلطانی	610399205
نیما نیرومند	610399199

پرسش 1. تشخیص اخبار جعلی مبتنی بر مدلهای ترنسفورمر

۱-۱. آشنایی با BERT و CT-BERT

در این سوال از روش یادگیری انتقالی استفاده می کنیم. در این روش می توانیم از ذخیره و انتقال اطلاعات بدست آمده از انجام وظیفه خاص برای انجام وظیفه جدید استفاده کنیم. از این روش زمانی استفاده می کنیم که داده های کمی برای یادگیری و ارزیابی مدل موجود است.

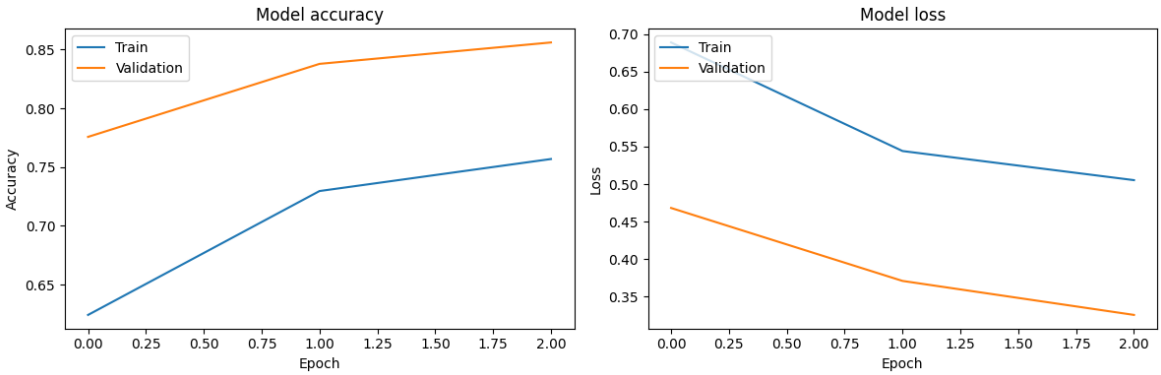
در پردازش زبان طبیعی، مدلی را روی مقدار زیادی داده متنی به صورت self supervised آموزش می دهند(pretraining). از خروجی این مدل که فهم contextual از ورودی دارد، با دو روش برای عمل یادگیری انتقالی برای مساله downstream استفاده می شود. روش Fine Tuning و روش Feature Based. در روش Feature Based، از نگاشت ویژگی های بدست آمده، به عنوان ورودی مدل دیگر (downstream) برای تصمیم گیری (طبقه بندی یا رگرسیون) استفاده می شود و صرفا مدل دوم برای وظیفه آموزش داده می شود و وزن های مدل اول ثابت است.

اما در روش Fine Tuning، پس از آموزش مدل اول روی داده های زیاد، در مرحله یادگیری برای وظیفه downstream، وزن های شبکه اول (feature extractor) نیز در کنار وزن های شبکه دوم آموزش داده می شود. فایده روش دوم، فهم عمیق تر مدل نسبت به مفهوم عبارات به کار رفته در دیتاست است.

1-2. پیاده سازی مدل با رویکرد FineTuning

داده های ورودی مربوط به اخبار شایع در توییتر در مورد کویید می باشد. هدف آموزش شبکه ای است که توانایی تشخیص درست یا غلط بودن خبر را دارد.

- در مدل اول، از شبکه BERT برای Word Embedding استفاده می کنیم. این مدل دارای 12لایه، 12 attention head و 110 میلیون پارامتر است و روی متن های انگلیسی با 2500 میلیون توکن پیش آموزش دیده است. ابتدا جملات را به صورت رشته های 128 تایی tokenize می کنیم. خروجی شبکه، ماتریسی است که عضو اول آن (CLS) طول 768تایی دارد. این رشته دارای context از کل عبارت ورودی است. خروجی را به عنوان ورودی به لایه fully connected با یک نود می دهیم و برای آن از تابع فعالساز sigmoid استفاده می کنیم و خروجی آن را به عنوان احتمال تعلق خبر به دسته واقعی تعبیر می کنیم.



تصویر۱ . نتایج مدل اول Feature Base

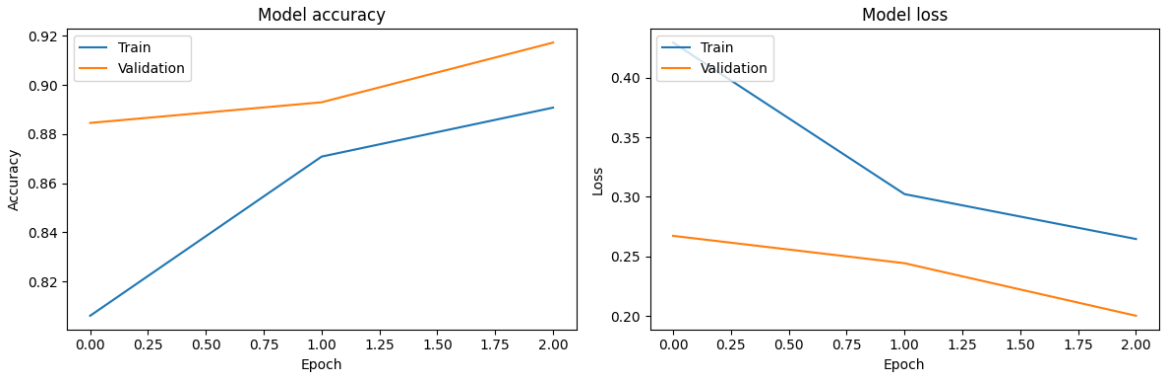
جدول۱ . نتایج مدل اول در رویکرد Feature Base

F1	loss	دقت
0.87	0.33	0.86

جدول۲ . درهمریختگی مدل اول Feature Base

	1 Predicted	0 Predicted
1 Label	762	258
0 Label	52	1068

- در مدل دوم، شبیه مدل اول عمل می کنیم با این تفاوت که خروجی BERT را به صورت ماتریسی با طول 128تایی در نظر میگیریم. این ماتریس قبل از ورود به لایه fully connected، از یک لایه GRU Bidirectional عبور می کند. خروجی آن وارد لایه fully connected با یک نود و تابع فعالساز sigmoid برای عمل طبقه بندی می شود.



تصویر۲ . نتایج مدل دوم Feature Base

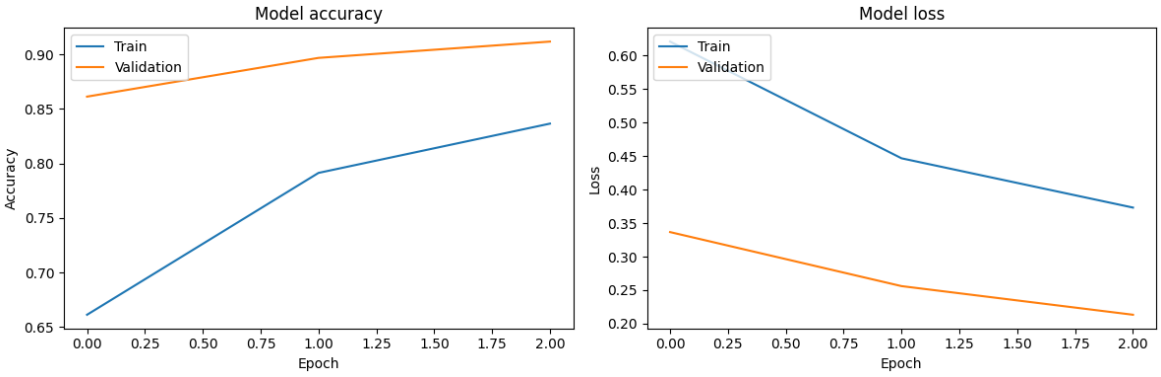
جدول۳ . نتایج مدل دوم در رویکرد Feature Base

F1	loss	دقت
0.92	0.21	0.92

جدول۴ . درهمریختگی مدل دوم Feature Base

	1 Predicted	0 Predicted
1 Label	884	136
0 Label	39	1081

- مدل سوم شبیه مدل دوم است. با این تفاوت که BERT را پس از پیش پردازش روی 2500میلیون توکن عمومی روی داده های با مفهم کویید پیش پردازش می دهیم تا درک عمیق تری نسبت به اخبار ورودی داشته باشد. این مدل CT-BERT می نامیم.



تصویر ۳. نتایج مدل سوم Feature Base

جدول ۵. نتایج مدل دوم در رویکرد Feature Base

F1	loss	دقت
0.92	0.21	0.91

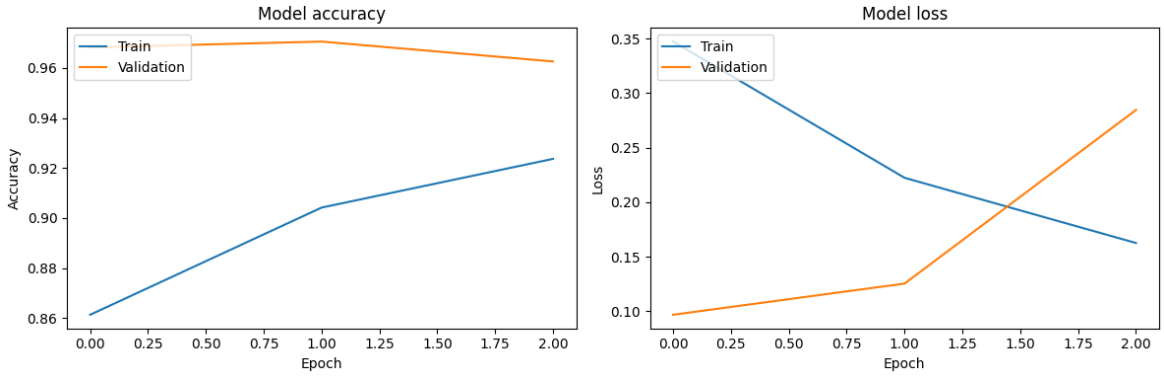
جدول ۶. درهمریختگی مدل سوم Feature Base

	1 Predicted	0 Predicted
1 Label	893	127

0 Label	63	1057

3-1. پیاده سازی مدل با رویکرد Feature-Based

- مدل اول:



تصویر۴ . نتایج مدل اول Fine Tuning

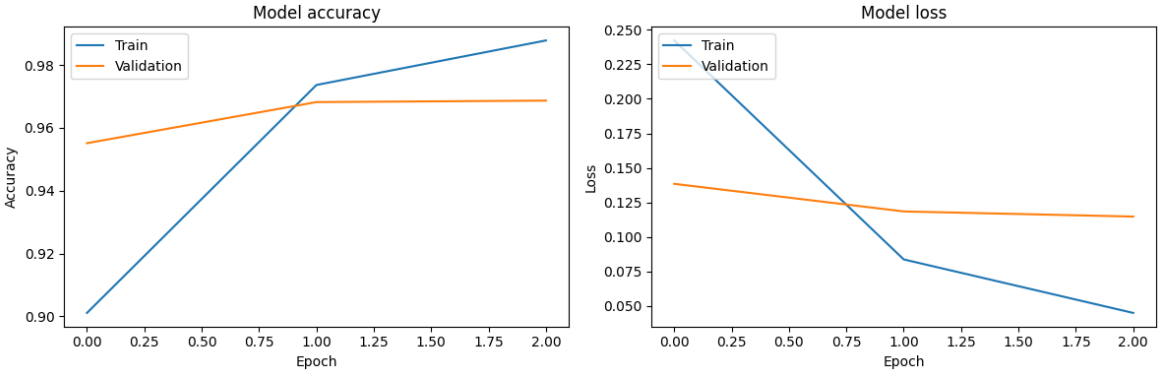
جدول۷ . نتایج مدل اول در رویکرد Fine Tuning

F1	loss	دقت
0.96	0.31	0.95

جدول۸ . درهمریختگی مدل اول Feature Base

	1 Predicted	0 Predicted
1 Label	968	52
0 Label	17	1103

- مدل دوم:



تصویر۵ . نتایج مدل دوم Fine Tuning

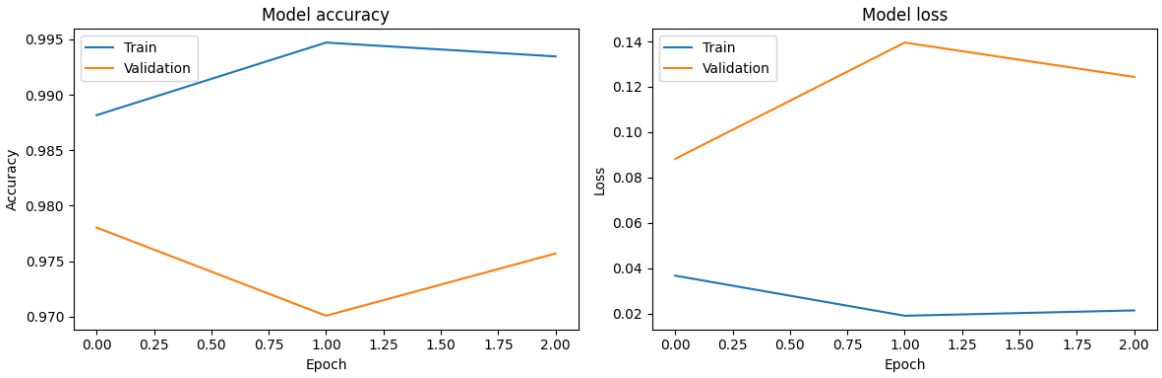
جدول۱۰ . نتایج مدل دوم در رویکرد Fine Tuning

F1	loss	دقت
0.97	0.11	0.96

جدول۱۱ . درهمریختگی مدل دوم Feature Base

	1 Predicted	0 Predicted
1 Label	925	95
0 Label	3	1117

- مدل سوم:



تصویر۶ . نتایج مدل سوم Fine Tuning

جدول۱۲ . نتایج مدل سوم در رویکرد Fine Tuning

F1	loss	دقت
0.98	0.14	0.97

جدول۱۳ . درهمریختگی مدل سوم Fine Tuning

	1 Predicted	0 Predicted
1 Label	975	45
0 Label	12	1108

4-1. تحلیل نتایج

در هر رویکرد یادگیری انتقالی، مدل سوم با استفاده از CT-BERT و مدل دوم با استفاده از BERT دقت تقریبا مشابه و بهتر از مدل دیگر کسب کردند. همچنین F1-Score آن ها نیز تقریبا برابر است. در رویکرد Fine Tuning، مدل سوم با استفاده CT-BERT، دقت بهتری از دو مدل دیگر (که از BERT استفاده کردند) کسب کرده است. همچنین F1-Score بالاتری نسبت به دو مدل دیگر دارد. بنابراین می توان نتیجه گرفت که استفاده از مدل BERT با پیش پردازش بر روی داده های مرتبط با داده های ورودی برای وظیفه خاص (آموزش مدل CT-BERT) و سپس آموزش آن برای انجام یک وظیفه خاص، نتایج بهتری دارد. زیرا درک بهتر (جدایی پذیری بیشتر برای مفاهیم دورتر و نزدیکی بهتر برای مفاهیم های نزدیک تر) از جملات مربوط به کوویید دارد و با آموزش مجدد روی داده های مربوط به یک وظیفه خاص مربوط به کوید، جدایی پذیری بالاتری در خروجی ایجاد می کند و عمل پیش بینی برای بخش های بعدی شبکه ساده تر می شود.

در رویکرد Feature-Based، مدل دوم (BERT+GRU+FC) دقت و F1-Score بهتری از مدل اول (BERT+FC) دارد. علت آن استفاده از GRU است که قدرت حافظه را به شبکه اضافه می کند و به عنوان یک لایه feature extractor عمل می کند. GRU الگوهایی که در توالی کلمات خروجی از BERT است را درک کرده و به لایه بعدی می دهد. این درک، قابلیت جدایی پذیری بالاتری به داده های دو کلاس می دهد (داده های هم کلاس نزدیک تر و داده ها از کلاس های متفاوت دورتر).

هر مدل، در رویکرد FineTuning نتایج بهتری نسبت به مدل متناظر در FeatureBase گرفته است. علت آن، افزایش اطلاعات لایه Embedding در مورد متن های ورودی مساله با توجه به هدف مساله است. در واقع لایه Embedding درکنار اطلاعات قبلی پیش آموزش دیده، اطلاعات جدیدی درمورد متون یاد می گیرد که برای تشخیص اخبار در لایه های بعدی مفید خواهد بود.

جدول ۱۴. نمونه تشخیص های غلط

مدل	پیش بینی	برچسب	خبر
اول	real	fake	An audio file by an alleged worker at a health institution in Rio de Janeiro. She says that healthcare workers on public institutions in Rio are forced to state whether a patient has COVID-19 or not even before he sees a doctor. This was allegedly being done to artificially inflate the number of cases
اول	real	fake	The Chinese government announced that "garlic is a preventative food for the the novel coronavirus".
دوم	fake	real	On the 15/03 NCDC directly contacted a Twitter user who mentioned his friend who returned from UK had runny nose but could not reach authorities for testing. Within 12 hours of communication with us via DM a sample was collected. We're committed to doing our best https://t.co/fccdGij3uG
دوم	fake	real	IndiaFightsCorona #COVID19 Recoveries exceed active # cases by more than 2.1 million. There has been more than 4 times jump in the average weekly recoveries from the first week of July to last week of August. https://t.co/HRyyaWhdJR
سوم	real	fake	RT @factchecknet: In the absence of clarity and a rash of #misinformation the #COVID19 pandemic has created a …breeding ground for #prejudi
سوم	fake	real	An even better piece of news: states reported fewer than 500 deaths. That hadn't happened since March. Yes there will probably be a larger number of deaths reported tomorrow as lagging weekend data gets posted but it is a significant pandemic milestone. https://t.co/DfIu3I5agZ

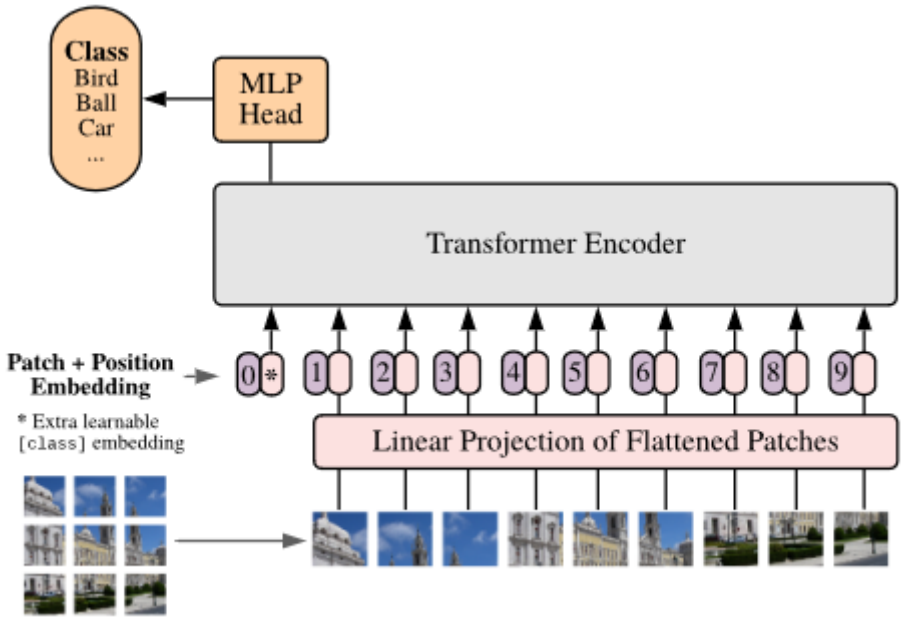
این طور که پیداست، می توان حدس زد که اگر خبر از طرف منبعی بیان شده باشد و آن منبع به صورت "some resource says:" یا "A news:" در ابتدای خبر بیان شود، مدل ها آن را درست تشخیص می دهد. همچین در صورتی که لینک خبر در انتها ظاهر شود، خبر اشتباه تشخیص داده می شود.

پرسش ۲ –به کارگیری مدلهای ترنسفرمر در طبقه بندی تصاویر

۲-۱. آشنایی با ترنسفورمرهای تصویر

الف) ViT یا همون vision transformer یک مدل است که برای پردازش تصویر استفاده می‌شود و به جای CNN ها سعی می‌شود از لایه‌های transformer استفاده کند. از آنجایی که در ابتدا ترنسفرمرها را برای داده‌های متنی تعریف کردیم و به صورت دنباله‌ای آن را توکن و سپس تعبیه(embedd) می‌کردیم حال پس برای عکساهم باید دنبال توکن کردن باشیم که همچنین مفهوم دنباله‌ای را نیز داشته باشد و برای این کار مثلا می‌توانیم فرض کنیم که قطعات کوچک عکس می‌توانند دارای یک مفهوم باشند(مانند یک کلمه در متن که از تعدادی حرف است) پس در ابتدا عکس‌ها را به قطعات کوچک(مثلا ۱۶ در ۱۶ تقسیم می‌کنیم.) و همانطور که گفتیم در ادامه این قطعات را به یک بردار تبدیل می‌کنیم و سپس مانند ترسفور برای متن‌ها موقعیت مکانی قطعات را نیز لحاظ می‌کنیم و سپس این بردارها را به شبکه ترسفرمر می‌دهیم و دیگر تغییر خاصی نسبت به حالت متنی نداریم.

ب)



شکل ۱. ساختار ViT

Patch Embedding : تصاویر را به قطعه‌های کوچک‌تر تقسیم می‌کند و سپس هر قطعه را به یک بردار تبدیل می‌کند.

Position Embedding: اطلاعات موقعیت مکانی را به بردارهای قطعات اضافه می‌کند.

Linear Projection of Flattened Patchs: قطعات تصویر به صورت مسطح تبدیل می‌کند و به یک بردار ویژگی تبدیل می‌شوند.

Transformer Encoder: شامل چندین لایه ترنسفرمر است.

MLP Head: در نهایت از ویژگی‌های استخراج شده چندین لایه فولی کانکتت می‌زنیم تا ویژگی ها استخراج شوند.

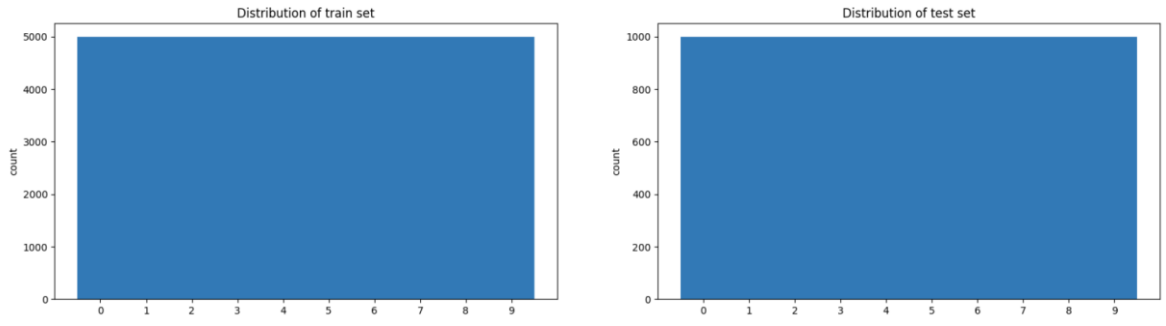
ج) از ایرادات می‌توان این را اشاره کرد که CNN ها موقعیت مکانی هم در نظر می‌گرفتند اما در این مدل تا حدی این اتفاق نمی‌افتد و داده‌ها چون به صورت دنباله‌ای داده می‌شوند سخت می‌شود این را لحاظ کرد همچنین از نظر زمانی نیز طولانی‌تر و به داده‌های بیشتری نیاز دارند.

پس با توجه به این اشکلات احتمالا ایده‌ای که بتونه یکی کمک کننده باشد این است که به صورت ترکیبی از هردوی CNN و ViT بهره ببریم تا از خوبی‌ها هر کدام بتونیم استفاده کنیم.

۲-۲. لود و پیش پردازش دیتاست

در این قسمت از مجموعه داده‌ی آماده CFAR10 استفاده می‌کنیم که شامل ۵۰۰۰ عکس برای مجوعه آموزشی و ۱۰۰۰ عکس برای مجموعه آزمایشی است همچنین عکسا ۳۲ در ۳۲ و رنگی (۳*۳۲*۳۲) هستند که دارای ۱۰ برچسب هستند و در ۱۰ کلاس طبقه‌بندی شده‌اند

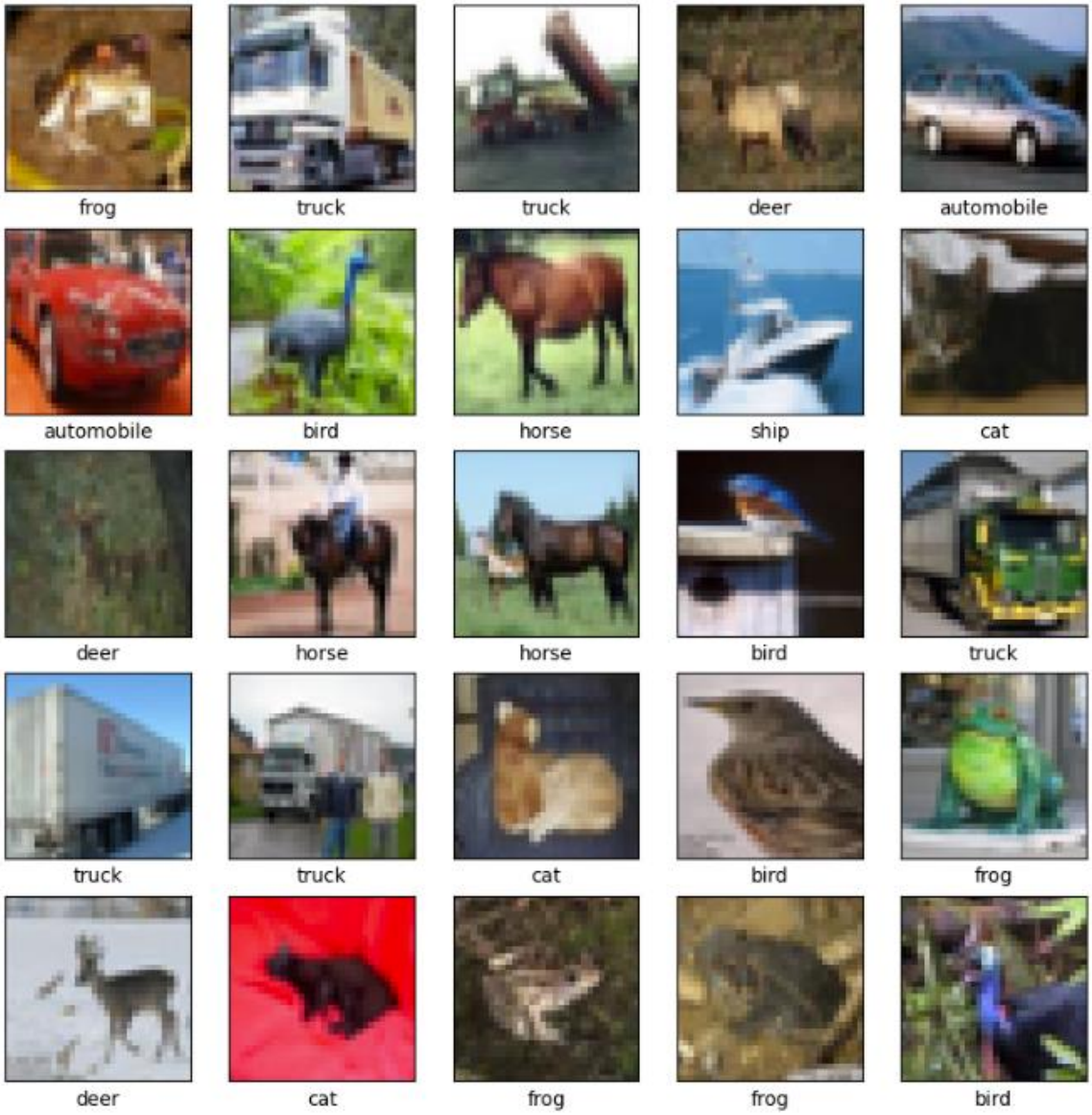
حال در هردو داده آموزشی و تست توزیع کلاس‌ها را بررسی می‌کنیم.



شکل ۱. توزیع آماری دسته‌ها در مجموعه آموزش و آزمایش

و همانطور که می‌بینیم در هردوی حالت آزمایش و تست داده‌ها به صورت یکنواخت و یکسان تقسیم شده‌اند(در آموزش از هر کلاس ۵۰۰ داده و در تست از هر داده ۱۰۰تا) پس توزیع داده‌ها متوزان است.

حال برای آشنایی بیشتر با مجموعه داده‌ها تصویر تعدادی را خروجی می‌دهیم.(توجه شود که کلاس‌های ما برچسب ۰ تا ۹ دارند و در عکسا به صورت دستی نام لیبل واقعی را نوشتیم.)

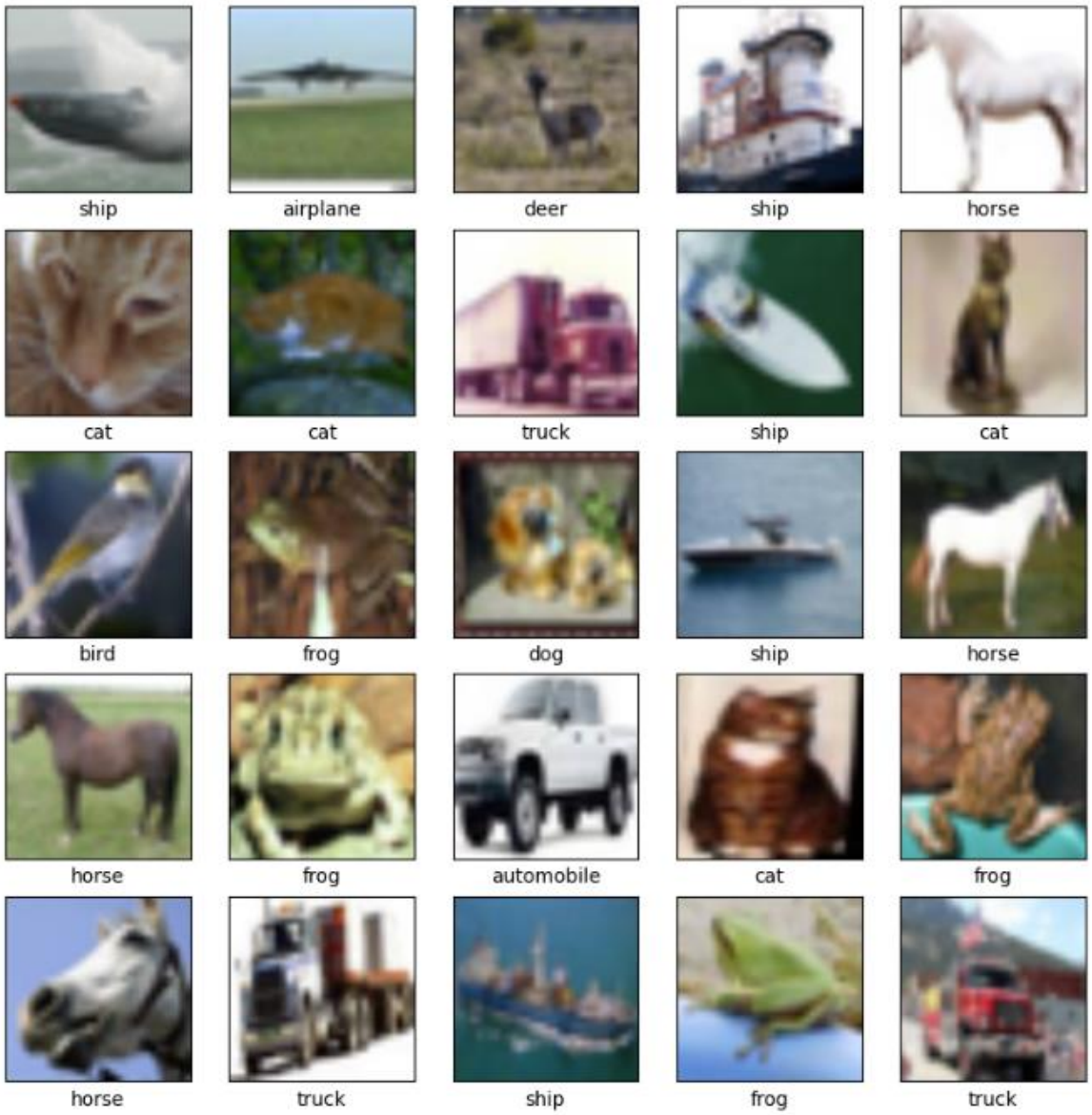


شکل ۲. نمونه‌ای از تصاویر دیتاست قبل از پیش پردازش

حال از آنجایی که مقادیر درایه‌های داخل آرایه عکسای ما ۰ تا ۲۲۵ هستند برای اینکه نرمال کنیم و اعداد را در بازه ۰ تا ۱ داشته باشیم از min-max normalization استفاده می‌کنیم و همه مقادیر آرایه‌ها را بر ۲۵ تقسیم می‌کنیم. حال از آنجایی که transfer learning استفاده می‌کنیم و از مدل‌های آموزش دیده‌ای

که استفاده می‌کنیم ورودیشون با سایز ۲۲۴ در ۲۲۴ است پس سایز عکس‌هامون را تغییر می‌دهیم و با استفاده از روش bilinear آن‌ها را تغییر سایز می‌دهیم.

حال تعدادی از عکس‌ها را بعد از این تغییرات خروجی می‌دهیم



شکل ۳. نمونه‌ای از تصاویر دیتاست قبل از پیش پردازش

۲-۳. شبکه کانولوشنی fine-tuning

الف) از مدل پیش آموخته شده VGG19 استفاده می‌کنیم که بر روی مجموعه Imagenet1k

یادگیری شده حال قسمت فولی کانکتت را حذف می‌کنیم و می‌خواهیم از ویژگی‌های استخراج شده توسط این مدل را استفاده کنیم.حال برای اینکه بعضی از لایه‌های مدل را دوباره آموزش بدیم طبق مقاله لایه‌های block5_conv1 به بعد را unfreeze می‌کنیم تا دوباره مدل را آموزش دهیم و fine-tuning کنیم حال برای تسک خودمان طبق گفته مقاله ابتدا لایه ها فلت می‌کنیم تا سایز داده‌ها یک بعدی شه حال از بردارهای به دست آمده را به یک لایه فولی کانکتت ۲۵۶ با نورون وصل می‌کنیم. حال برای جلوگیری از بیش برداش Dropout با مقدار احتمال ۵۰ درصد اضافه کنم حال برای خروجی از آنجایی که ۱۰ کلاس داریم یک لایه با ۱۰ نورون و با Softmax به عنوان تابع فعال‌ساز می‌گذاریم.

ب) تعداد پارامترهای trainable ۹۵۷۳۱۳۰، است از آنجایی که خروجی مدل VGG19 بعد از فلت کردن یک بردار ۵۱۲ تاست و به یک لایه ۲۵۶ نورونی وصل است پس ۲۵۶*(۵۱۲+۱) = ۱۳۱۳۲۸ و بعد این ۲۵۶ را به یک لایه ۱۰ تایی وصل می‌کنیم و ۱۰*(۲۵۶+۱) = ۲۵۷۰ و باقی مربوط به لایه‌های آخرunfreeze شده مدل VGG19 است.

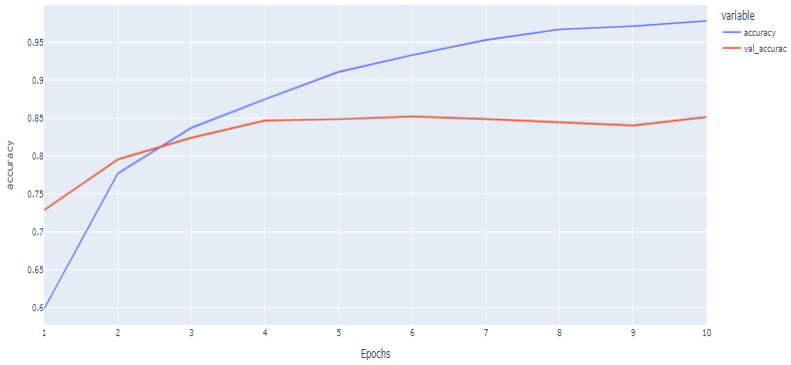
ج) حال مدل را با بهینه‌ساز آدام و لرنینگ ریت ۰.۰۰۰۱ و تابع loss ، categorical_crossentropy و با batchsize ، ۶۴ را به تعداد ۱۰ ایپاک ران می‌کنیم.

د) در پایان ایپاک دهم دقت و مقدار تابع هزینه روی داده‌های آموزشی و ارزیابی به شکل زیر می‌شوند

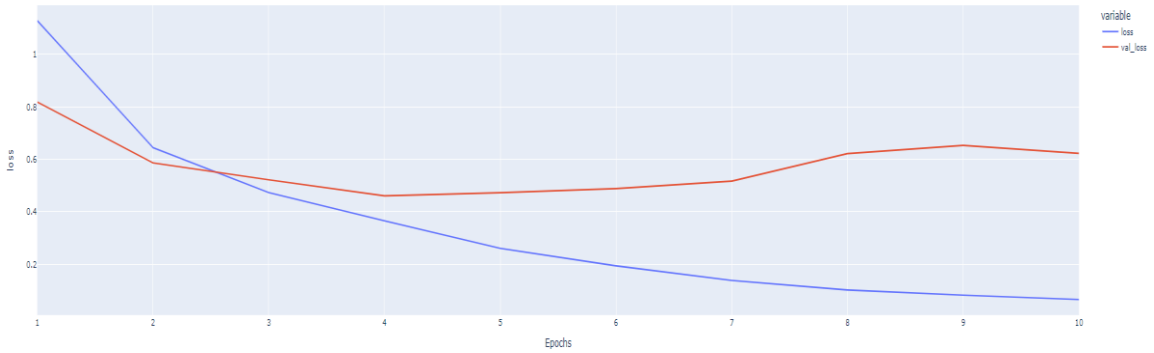
	دقت
داده آموزشی	۹۷.۸
داده ارزیابی	۸۵.۲

	تابع هزینه
داده آموزشی	۰.۰۶
داده ارزیابی	۰.۶۲

همچنین نمودار تغییراتشان را نیز در زیر نشان دادیم



شکل ۵. نمودار تغییرات دقت در داده‌های آموزشی و ارزیابی



شکل ۶. نمودار تغییرات تابع هزینه در داده‌های آموزشی و ارزیابی

ه)

	میانگین زمان(برحسب ثانیه)
زمان آموزش	۲۹۸.۶۶
زمان ارزیابی	۲۵.۲۱

۴-۲. fine-tuning شبکه ترنسفرمر

الف) از مدل پیش آموخته شده DeiTBaseDistilled استفاده می‌کنیم که بر روی مجموعه Imagenet1k یادگیری شده می‌خواهیم از ویژگی‌های استخراج شده توسط این مدل را استفاده کنیم.حال برای اینکه بعضی از لایه‌های مدل را دوباره آموزش بدیم طبق مقاله لایه‌های دوازدهم به بعد را از قسمت ترنسفرمر شبکه را unfreeze می‌کنیم تا دوباره مدل را آموزش دهیم و fine-tuning کنیم حال برای

تسک خودمان طبق گفته مقاله ابتدا لایه ها فلت می‌کنیم تا سایز داده‌ها یک بعدی شه حال از بردارهای به دست آمده را به یک لایه فولی کانکتت ۲۵۶ با نورون وصل می‌کنیم. حال برای جلوگیری از بیش بردازش Dropout با مقدار احتمال ۵۰ درصد اضافه کنم حال برای خروجی از آنجایی که ۱۰ کلاس داریم یک لایه با ۱۰ نورون و با Softmax به عنوان تابع فعال‌ساز می‌گذاریم.

ب) تعداد پارامترهای trainable ،۹۶۳۰۴۲۶ است از آنجایی که خروجی مدل یک بردار ۱۰۰۰ تاست و به یک لایه ۲۵۶ نورونی وصل است پس ۲۵۶*(۱۰۰۰+۱) = ۲۵۶۲۵۶ و بعد این ۲۵۶ را به یک لایه ۱۰تایی وصل می‌کنیم و ۱۰*(۲۵۶+۱) = ۲۵۷۰ و باقی مربوط به لایه‌های آخرunfreeze شده مدل است.

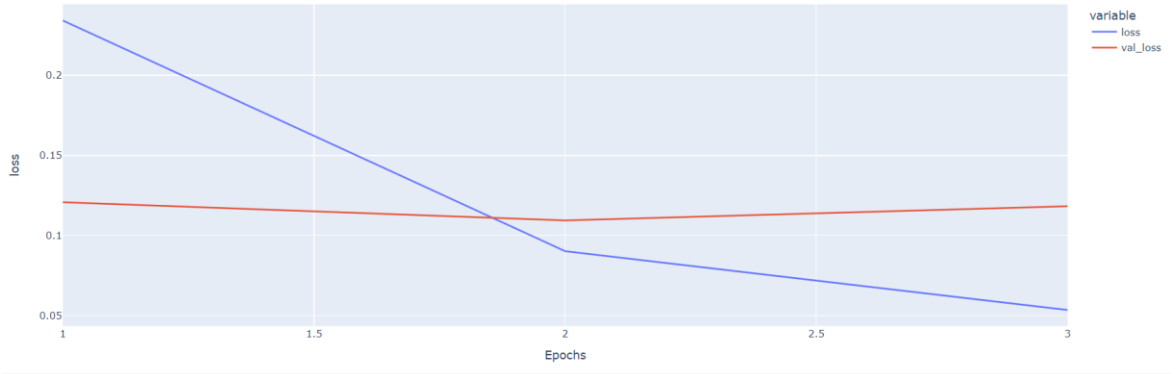
ج) حال مدل را با بهینه‌ساز آدام و لرنینگ ریت ۰.۰۰۰۱ و تابع loss ، categorical_crossentropy و با batchsize ، ۶۴ را به تعداد ۳ ایپاک(به علت طولانی بودن و رسیدن به جواب خوب در همین تعداد نسبت به حالت قبل تعداد ایپاک‌ها را کم کردیم) ران می‌کنیم.

د) در پایان ایپاک دهم دقت و مقدار تابع هزینه روی داده‌های آموزشی و ارزیابی به شکل زیر می‌شوند

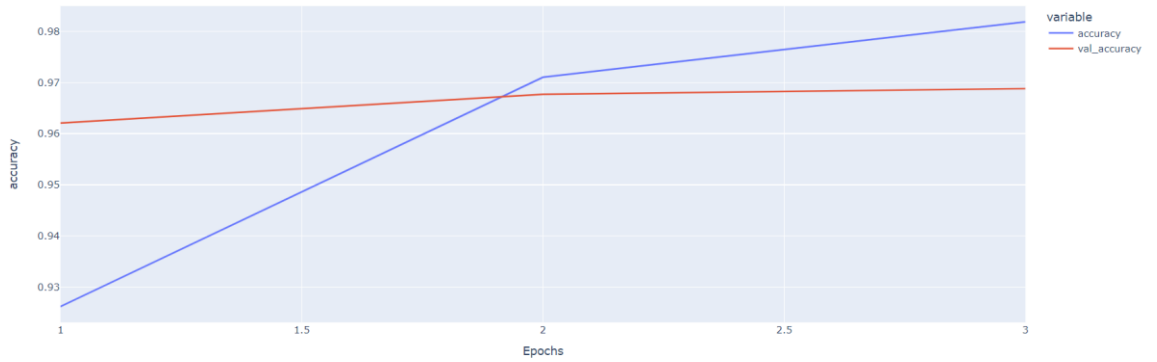
	دقت
داده آموزشی	۹۸.۲
داده ارزیابی	۹۶.۷

	تابع هزینه
داده آموزشی	۰.۰۵۳
داده ارزیابی	۰.۱۱۸

همچنین نمودار تغییراتشان را نیز در زیر نشان دادیم



شکل ۵. نمودار تغییرات تابع هزینه در داده‌های آموزشی و ارزیابی



شکل ۶. نمودار تغییرات دقت در داده‌های آموزشی و ارزیابی

ه)

	میانگین زمان(برحسب ثانیه)
زمان آموزش	۱۳۴۷
زمان ارزیابی	۱۳۷

۵-۲. مقایسه نتایج

خب همینطور که در بخش‌های قبل گفتیم دقت ما روی مجموعه ارزیابی در ViT ، ۹۶.۷ و در CNN ، ۸۵.۲ شده است و همانطور که انتظار داشتیم مدل ترنسفرمری ما علمکرد بهتری نسبت به CNN داشته است که در مقدار تابه هزینه و مقایسه بینشون مشخص هست

و همانطور که از نمودار هایشان مشخص است مدل ترنسفری در همان ایپاک‌های اول به نتایج خوبی می‌رسد و از کانولوشنی بهتر عمل کرده است که می‌توان تاثیر معماری ترنسفرمر و مفهوم attention در بهبود دقت را مشاهده کرد

همچنین دو مدلی انتخابی تعداد پارامترهای قابل یادگیریشون تفاوتی زیادی را دارا نمی‌باشند اما خب از سویی معماری ترنسفورمر باعث می‌شود که زمان یادگیری آن خیلی بیشتر باشد و تقریبا ۴ برابر نسبت به مدل کانولوشنی شده است که این را می‌توان از معایب مدل کانولوشنی دانست .

در مقایسه با مقاله هم دقت ما روی مجموعه اعتبارسنجی در مدل ترسفرمر تقریبا همان شده است اما در مدل کانولشنی حدود ۵ درصد کاهش داشته است که یکی از دلایل این تغییر می‌تواند تعداد ایپاک‌ها باشد اما با بررسی نمودار نظر میرسد که دقت همگرا شده بوده و بعید است تاثیر گذار بوده باشد و باقی شرایط به مانند مقاله بوده اما تفاوتی که ممکن است باشد این که داده افزایی بیشتر و مانند مقاله می‌توانست دقت را بالاتر ببرد و مدل نمونه‌های بیشتری را دارا باشد