تمرین سوم پایتون
شبکه‌های عصبی
بهار ۱۴۰۲

سارا قوام‌پور ۹۸۲۲۷۶۲۱۷۵

هدف از تمرین۳، پیاده‌سازی شبکه RNN به‌منظور حل مسئله sentiment classification می‌باشد. از داده ست Sports and Outdoors کامنت‌های آمازون به این منظور استفاده شده‌است.۵ کلاس وجود دارد که کلاس ۵ به منظور بیشترین رضایت‌مندی است. ابتدا شبکه‌هایی با معماری‌های متفاوت بر روی دیتاست ذکرشده آموزش می‌بینند و سپس تاثیر بازنمایی‌های از قبل آموزش دیده شده و داده‌های ناموجود در دیتاست آموزش بررسی می‌شود. در انتها عملکرد مدل آموزش دیده بر روی دیتاست دیگری به نام دیتاست imdb سنجیده می‌شود.

بخش اول

۱- پیش‌پردازش دادگان

ابتدا دیتاست Sports and Outdoors توسط wget دانلود می‌شود. این دیتاست به صورت فشرده دانلود می‌شود به همین علت ابتدا باید از حالت فشرده خارج شود. این دیتا ست از چند فیلد تشکیل شده‌است که برای تسک این تمرین تنها نیاز به قسمت overview که متن کامنت می‌باشد و overall که عددی بین ۱ تا ۵ است می‌باشد. هر خط از داده پردازش می‌شود و قسمت overview در کلیدی به همین نام در دیتافریم قرار میگیرد. Overall هم به همین صورت.در انتها قسمت overview به عنوان x دیتاست Overall هم به عنوان y دیتاست خروجی داده می‌شود. قبل از خروجی اما پیش پردازش هایی مانند حذف تگ ها، تبدیل جمله به آرایه ای از کلمه‌ها و تبدیل هر کلمه به ریشه آن صورت می‌گیرد.از این دیتاست به تعداد ۲.۴ میلیون داده لود می‌شود تا برای آموزش و تست مورد استفاده قرار بگیرد.

در هر مدل که به داده‌های متنی کار می‌کند باید به نحوی متن را به صورت مجموعه‌ای از اعداد به مدل ورودی داد. به این منظور در ابتدا از Tokenizer موجود در tf استفاده شده است. Tokenizer بر روی قسمت x داده فیت می‌شود و هر کلمه جمله را به صورت آرایه ای از اعداد در می‌آورد که هر عدد درواقع عدد متناظر به کلمه میباشد. به عبارتی Tokenizer با استفادع از داده های داده شده به آن یک دیکشنری تشکیل میدهد که هر کلمه را به یک عدد نسبت میدهد . سپس هر جمله را تبدیل به دنباله ای از اعداد متناضر با کلمه‌های موجود در آن جمله می‌کند.

نکته مورد توجه دیگر این می باشد که ورودی‌های شبکه باید ط.ل برابر داشته باشند و به همین دلیل ابتدا با یک تابع کمکی میانگین طول جمله‌های مجود در دیتاست محاسبه می‌شود و سپس همه جلمه ها به ان سایز تبدیل می‌شوند. همجنین دقت شد که y های مربوط به آموزش باید به فرمت ۱-hot دربیایند.

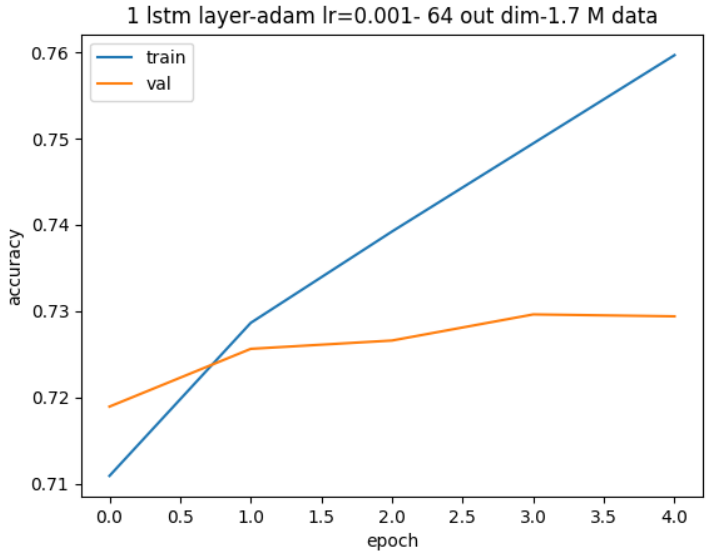
۲- پیاده‌سازی شبکه

برای معماری شبکه ابتدا از یک لایه embedding استفاده می‌شود. از آنجایی که با استفاده از Tokenizer یک دیشکنری ایجاد شد بردار اولیه برای هر کلمه اندازه ای برابر با سایز دیکشنری دارد و به همین علت .ورودی این لایه نیز سایز دیکشنری است. این لایه مسئول یادگیری بازنمایی کلمات می‌باشد. این لایه بازنمایی های ابتدایی را دریافت میکند و پس از آموزش مدل به عنوان خروجی بازنمایی‌های اموزش یافته را خروجی میدهد که سایز این بازنمایی خروجی باید به عنوانورودی به این لایه داده شود. بعد از این لایه

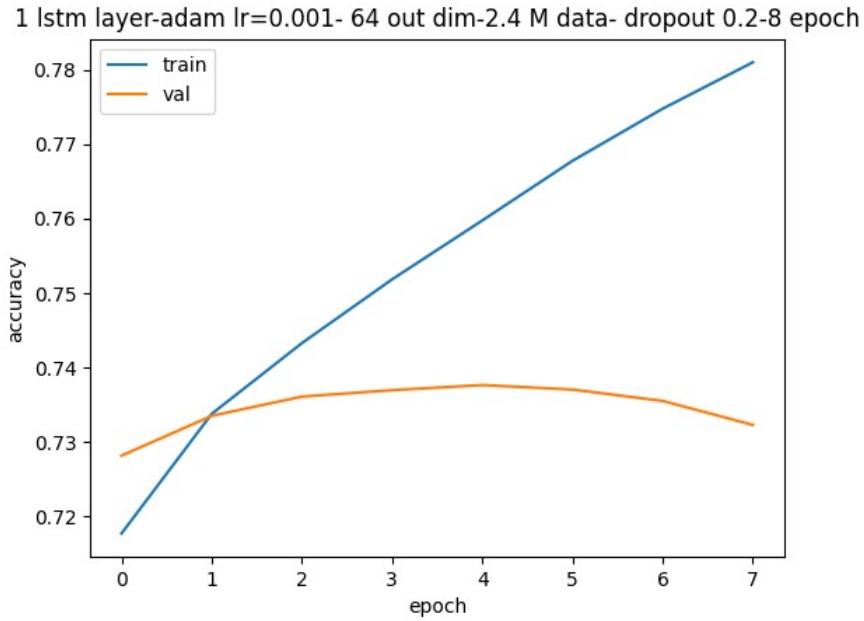
ابتدا با تعداد یک لایه، لایه های مختلف مانند Bidirectional،GRU، LSTM و SimpleRNN تست شده‌اند.در ادامه مشاهده می‌شود که این مدل ها با overfit روبه‌رو می‌شوند و به این منظور از لایه های dropout و BtchNormalization نیز استفاده شده است. یک لایه Dense با ۵ نورون برای ۵ کلاس با تابع فعال سازی سافت مکس در انتها قرار میگیرد. برای بهینه‌ساز با دو بهینه ساز adam و RMSprop تست انجام شده است. متریک انتخاب شده اکیورسی و تابع لاس هم CregoricalCE انتخاب شده است.

۳- انجام آزمایش‌ها

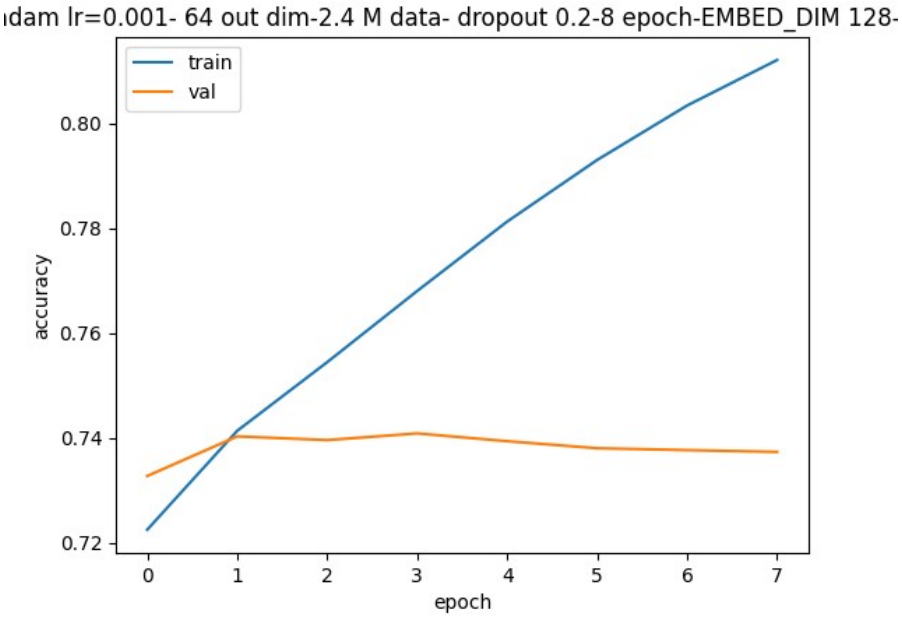
در ایتدا مدل زیر با پارمترهای مشخص شده ران می‌شود و این مدل دچار مشکل اورفیت است. در ایتدا تعداد داده ۱.۷ میلیون استفاده می‌شد.



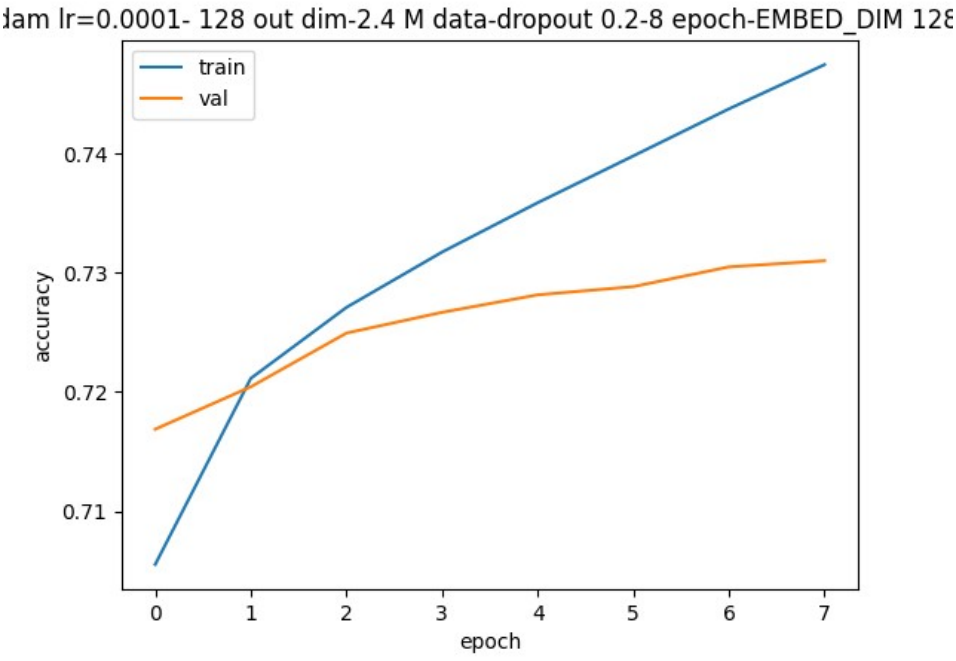
برای آدرس دادن به این مشکل تعداد داده‌ها به ۲.۴ میلیون افزایش داده شد و لایه dropout با نرخ ۰.۲ اضافه شد و مدل زیر ران شد.



مشاهده می‌شود که همچنان مشکل اورفیت وجود داردو حل نشده است و این در حالی است که تعداد داه زیادی به شبکه داد ه شده است و از دراپ اوت هم استفاده شده است. تصمیم بعدی برای حل مشکل اورفیت این است که خروجی لایه بازنمایی از ۶۴ به ۱۲۸ افزایش پیدا کند.

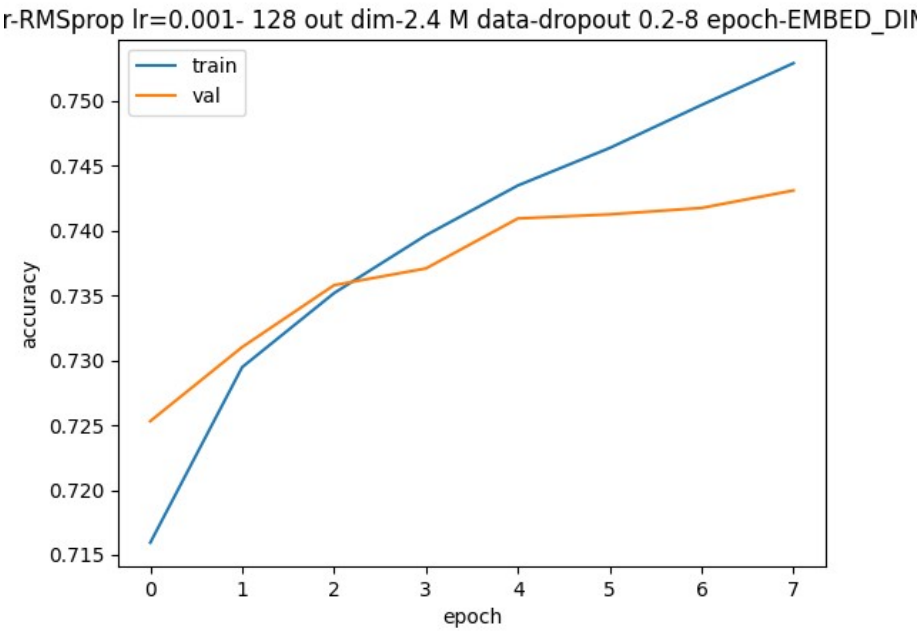


مشاهده میشود که تغییر خاصی رخ نداده است.(مقداری اکیورسی ها در هر دو ست افزایش پیدا کرده اند اما باز هم اورفیت وجود دارد.). همچنین باید توجه داشت که این مدل ها با ۸ ایپاک هم به همگرایی نرسیدند و به علت زمان تخمینی حدود ۱۰ دقیقه برای ران هر مدل امکان استفاده از ایپاک های بیشتر وجود نداشت. آزمایش بعدی به منظور حل اورفیت کاهش Learning rate از ۰.۰۰۱ به ۰.۰۰۰۱ بود.

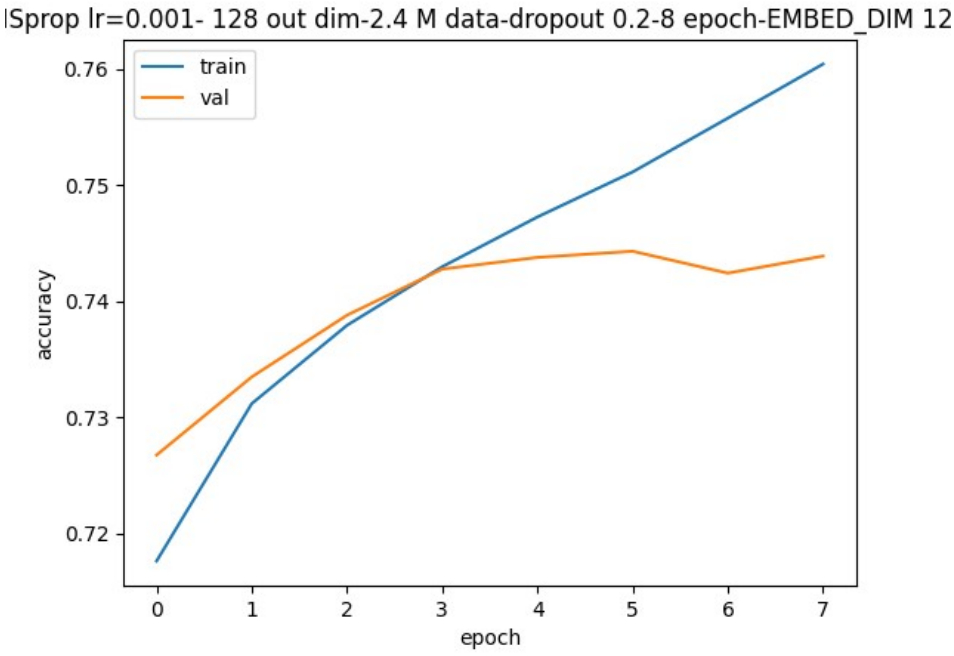


مشاهده می‌شود که اکیورسی ها مقداری کاهش پیدا کرده اند اما اورفیت کمتر شده است. تا اینجا همه مدل ها دارای یک لایـــه lstm بودند. در ادامه برای مقایسه بین انواع لایه‌های RNN مدل های زیر با پارمترهای مشخص شده ران شدند.

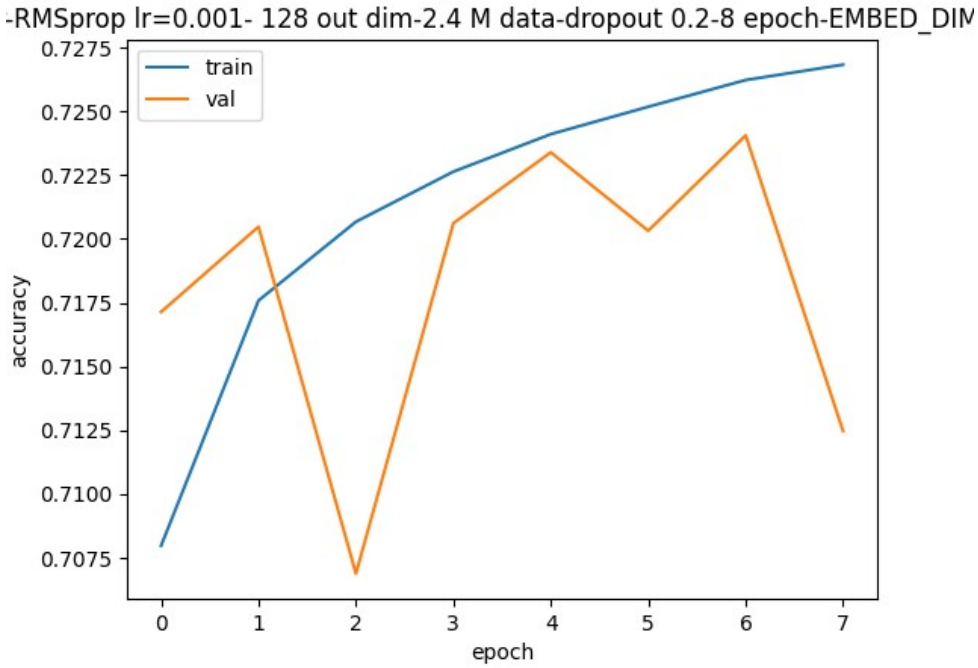
یک لایه Bidirectional:



یک لایه GRU:

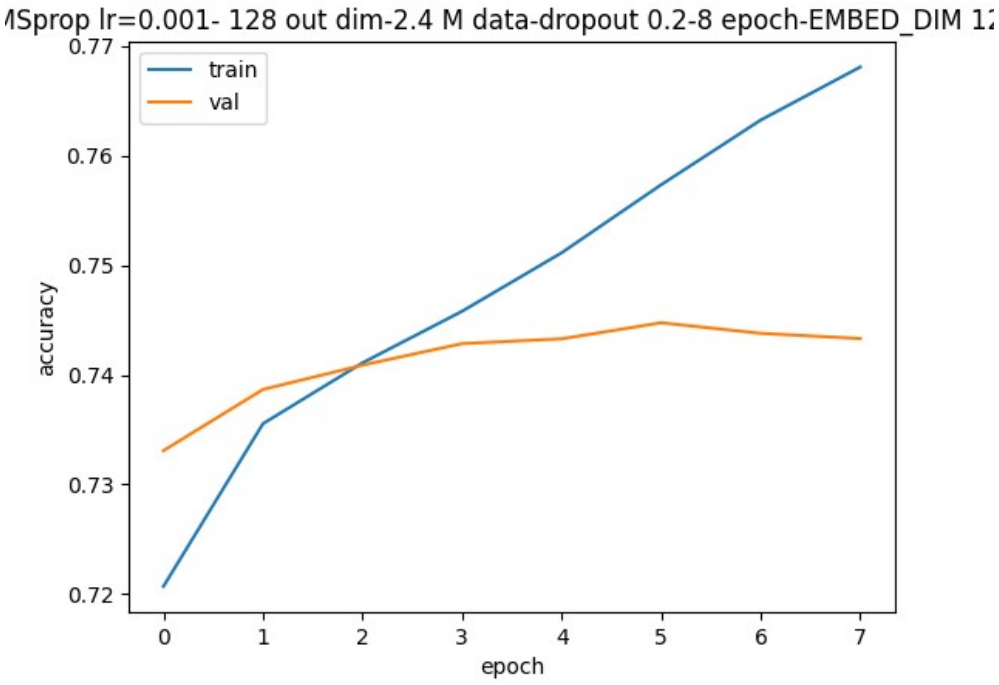


یک لایه SimpleRNN:



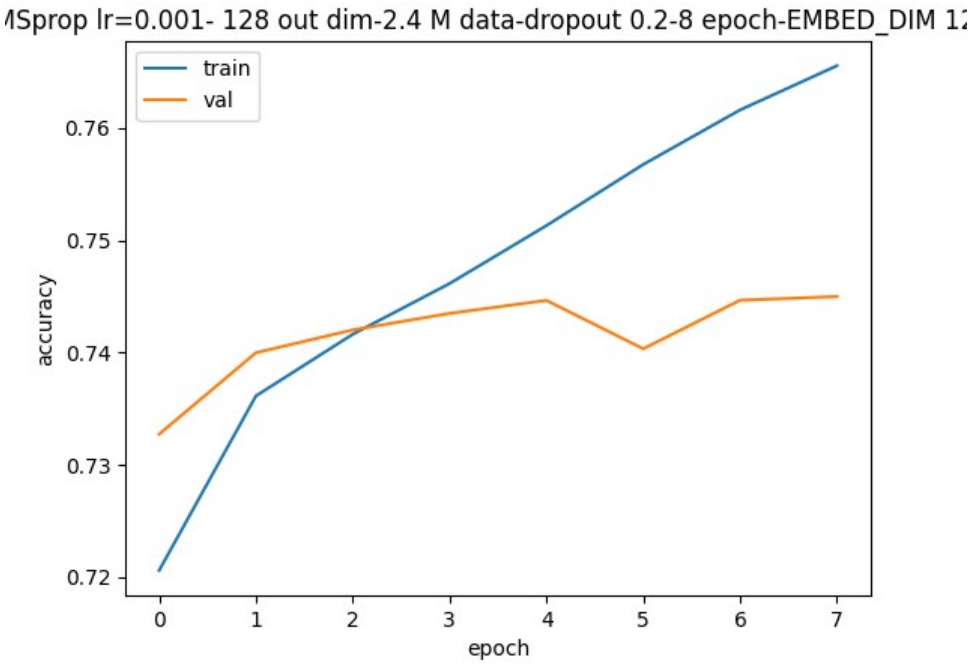
مشاهده می‌شود که بیشترین اکیورسی متعلق به مدل یک لایه GRU است و کمترین متعلق به SimpleRNN. یک لایه Bidirectional مقدار کمی از یک لایه lstm بهتر عمل کرده است اما هر دو کمتر از یک لایه GRU هستند.(مقایسه ها بر اساس جدول ۱ انجام شده است.)

قدم بعدی ازمایش ها مربوط به بررسی تعداد لایه است.ایتدا مدل دو لایه lstm تست می‌شود.



مشاهده میشود که مقدار کمی در حد هزارم اکیورسی از یک لایه lstm بهبود داشته است.(بر اساس جدول ۱)

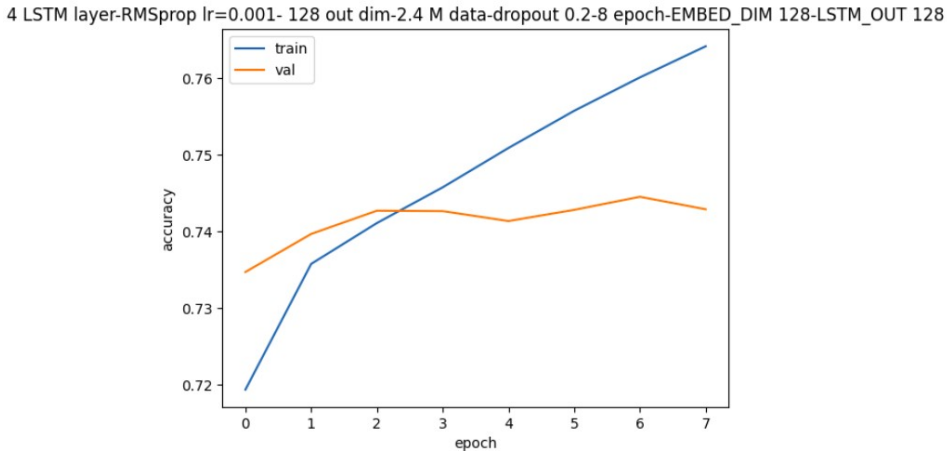
سه لایه lstm:



دقت شود که مقایسه این مدل ها با هم از روی این نمودار ها انجام نشده‌است و از روی اکیورسی که روی داده تست با تابع predict صورت گرفته است، انجام شده ایت و جدول حاوی اکیورسی عملکرد همه مدل ها بر روی داده تست در ادامه داخل جدول ۱ قرار گرفته است.

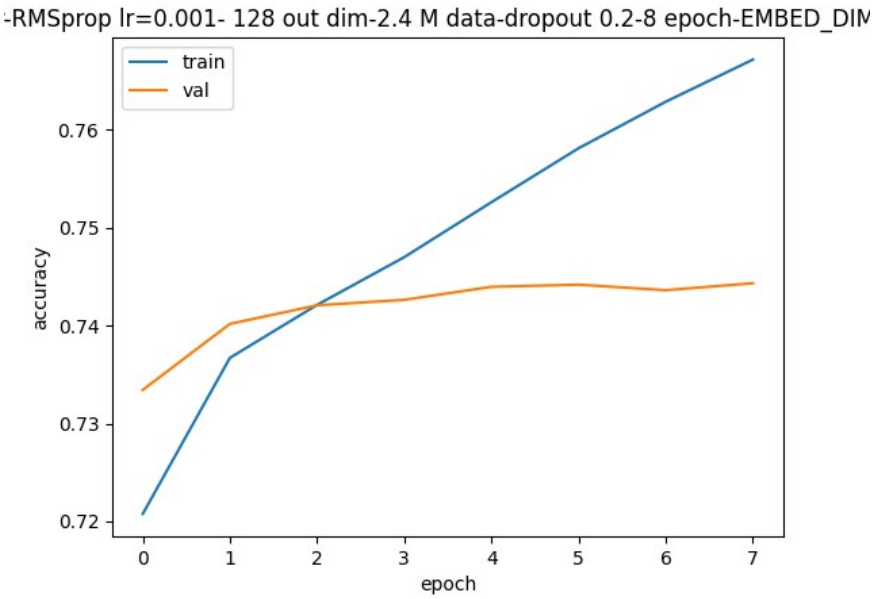
مدل ۳ لایه lstm بر اساس جدول ۱ از ۲ لایه بهتر عمل کرده است بر روی داده تست عمل کرده است و به طور کلی هم به عنوان بهترین مدل(مدل پایهبرا ازمایش های بخش های بعد) برای تست w2v و آزمایش های بعدی بر روی آن انتخاب می‌شود.

مدل ۴ لایه lstm:



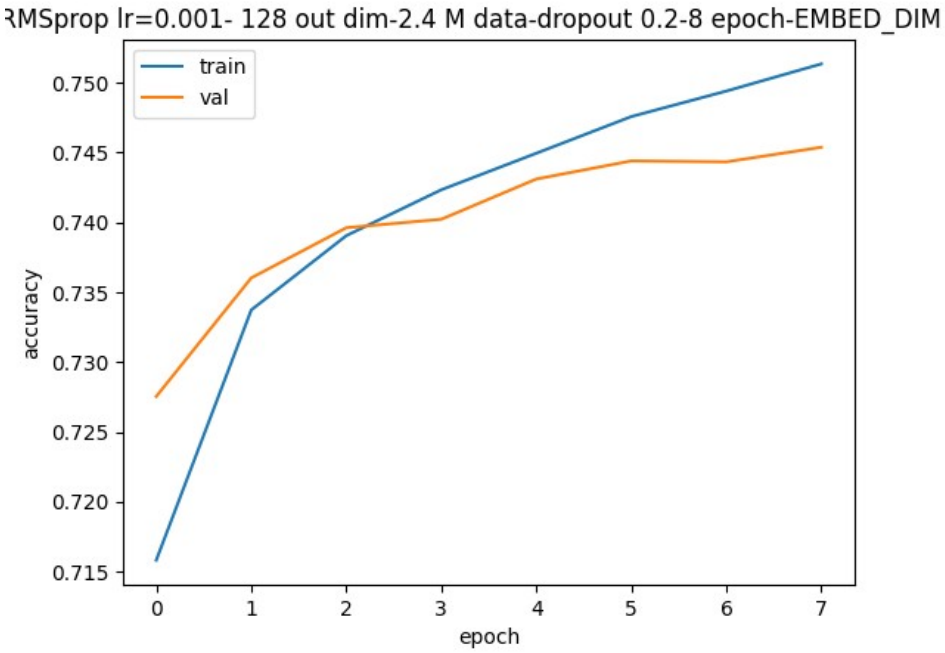
مدل ۴ لایهlstm نسبت به ۳ لایه بهبود نداشته است.

در انتهای این مرحله نیز مدل ۳ لایه که ۲ لایه اول آن lstm و لایه سوم آنgru استآزمایش شده است.



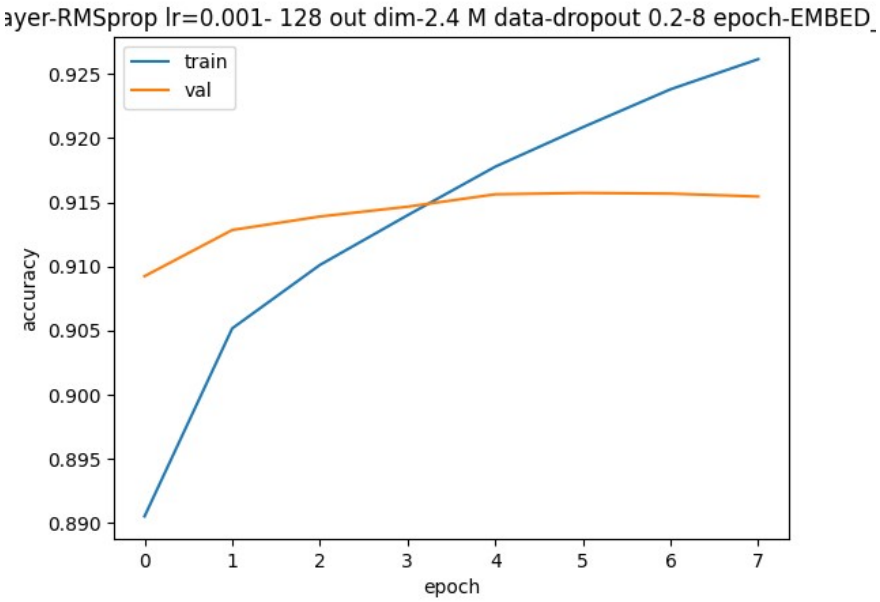
طبق جدول ۱ این مدل نسبت به مدل ۳ لایه و ۴ لایه lstm کاهش اکیورسی داشته‌است.

برای مرحله دوم ازمایش ها از بارنمایی های از پیش موزش داده شده (w2v) Word2Vec استفاده می‌شود.از کتابخانه gensim این مدل از پیش اموزش داده بر روی دیتاست ویکی‌پدیا را با نام glove-wiki-gigaword-100 لود می‌شود این مدل برای هر کلمه بازنمایی به طول ۱۰۰ ایجاد می‌کند. برای استفاده از این بازنمایی‌ها کافیست تا در ابتدا برای همه لغات موجود در دیکشنری ساخته شده توسط Tokenizer بازنمایی w2v آن هارا از مدل لود شده دریافت کرده و در ماتریسی به نام ناتریس امبدینگ قرار دهیم. سپس در لایه Embedding این ماتریس را به عنوان وزن ورودی دهیم. در اینجا اضافه کردن باز نمایی های w2v به مدل ۳ لایه lstm انجام شده است.



با توجه به جدول ۱ عملکرد این مدل نسبت به مدل ۳ لایه lstm کاهش جزئی داشته است.

برای مرحله سوم آزمایش‌ها که اموزش دادن مدل روی داده‌های ۳ کلاس ۱و۳و۵ و تست آن روی داده هـای کلاس هـای ۲و ۴ اسـت، یک تابع نوشته شده است که داده های سه کلاس ۱,۳,۵ را بعنوان داده اموزش و دادهه ای کلاس های ۲ و ۴ را به عنـوان داده تسـت خروجی میدهد. سپس بر روی این داده اموزش جدید نیز پیش‌پردازش های قبلی به وسـیله Tokenizer انجـام میشـود و ورودی بـه مدل (بدون بازنمایی w2v) داده می‌شود. پس از اموزش این مدل پلات ان به صورت زیر است:



مشاهده میشود که اکیورسی آموزش و validation افزایش پیدا کرده است و به بـالای ۹۰ رسـیده اسـت امـا بـا توجـه بـه جـدول ۱ اکیورسی این مدل برای داده های تست برابر با ۰ است. داده های تست از کلاس هایی بودند که تا حـالا مـدل ان هـارا ندیـده اسـت و مدل سعی میکند تا نزدیک ترین کلاسی را که میشناسد به ان ها نسبت دهد. به عنوان مثال به کلاس ۲ کلاس های ۱و ۳ و به کلاس ۴ کلاس های ۳ و ۵ را نسبت دهد. اما باز هم چجون این کلاس ها، کلاس های درست نیستند اکیورسی ۰ میشود.

در بخش سوم آزمایشه اعملکرد مدل اموزش دیده قبلی(بر روی داده های ۳ کلاس ۱و۳و۵) ازمایش می‌شود. به‌این منظور ابتـدا دیتـا ست imdb با wget دانلود میشود و توسط یک تابع مشابه تابع قبلی یک دیتا فریم یاخته میشود و پیش پردازش هـای ماننـد حـذف تگ ها و تیدیل به حروف کوچک و تبدیل به ریشه مردن صورت میگیرد سپس این داده های به عنوان داده تست به مدل داده میشوند با استفاده از predict و اکیورسی برای این پاسخ ها شنجیده میشود که برابر با مقداری نزدیک به ۰ و در اینجا ۰.۰۲ است.

دلیل این موضوع این میباشد که داده های imdb دارای دو کلاس مثبت و منفی هستند که چون توشط مدل در حین امـوزش دیـده نشده اند، مدل داده های مثبت را به کلاس ۵ و داده های منفی را به کلاس ۱ نسبت میدهد و چون این کلاس ها، کلاس های درسـت نیستند اکیورسی مقدار پایینی به دست می‌آید.

نام مدل	اکیورسی روی داده تست
۱ لایه LSTM	۰.۷۴۴۷۰۶۵۶۸۸۵۶۲۸۵۴
۱ لایه Bidirectional	۰.۷۴۷۹۵۷۶۵۲۵۵۰۸۵۰۳
۱ لایه GRU	۰.۷۴۹۴۵۸۱۵۲۷۱۷۵۷۲۵
۱ لایه SimpleRNN	۰.۷۱۱۴۰۳۸۰۱۲۶۷۰۸۹
۲ لایه LSTM	۰.۷۴۸۹۱۶۳۰۵۴۳۵۱۴۵۱
۳ لایه LSTM	۰.۷۵۱۲۰۸۷۳۶۲۴۵۴۱۵۲
۳ لایه LSTM	۰.۷۵۱۰۰۰۳۳۳۴۴۴۴۸۱۵
۲ لایه LSTM و ۱ لایه GRU	۰.۷۴۷۶۶۵۸۸۸۶۲۹۵۴۳۲
۳ لایه LSTM با بازنمایی w2v	۰.۷۴۹۶۲۴۸۷۴۹۵۸۳۱۹۴

جدول ۱