

به نام خدا



دانشکده مهندسی مکانیک

نام درس: هوش مصنوعی

تمرین ۳ (شبکه عصبی)

استاد درس: دکتر شریعت‌پناهی

دانشجو:

مهدی نوذری

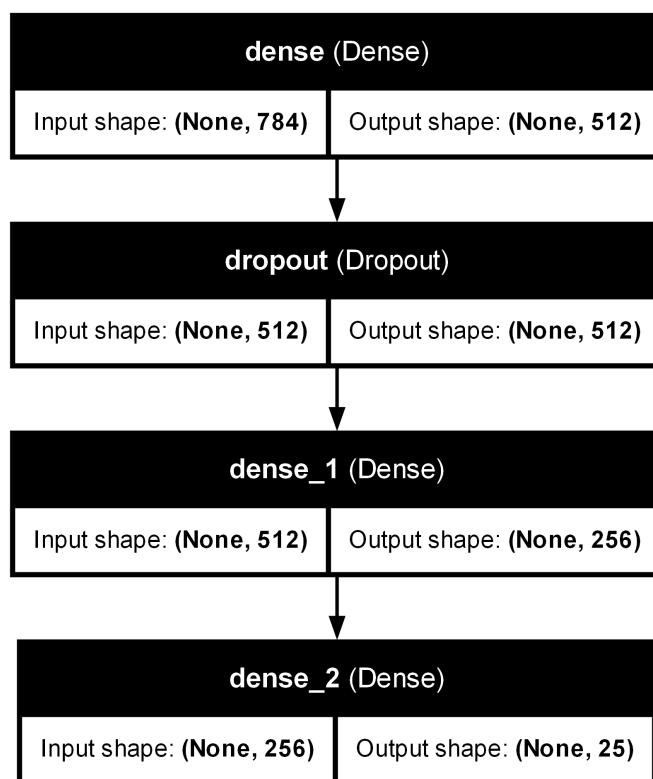
۸۱۰۶۰۱۱۳۹

بهار ۱۴۰۳

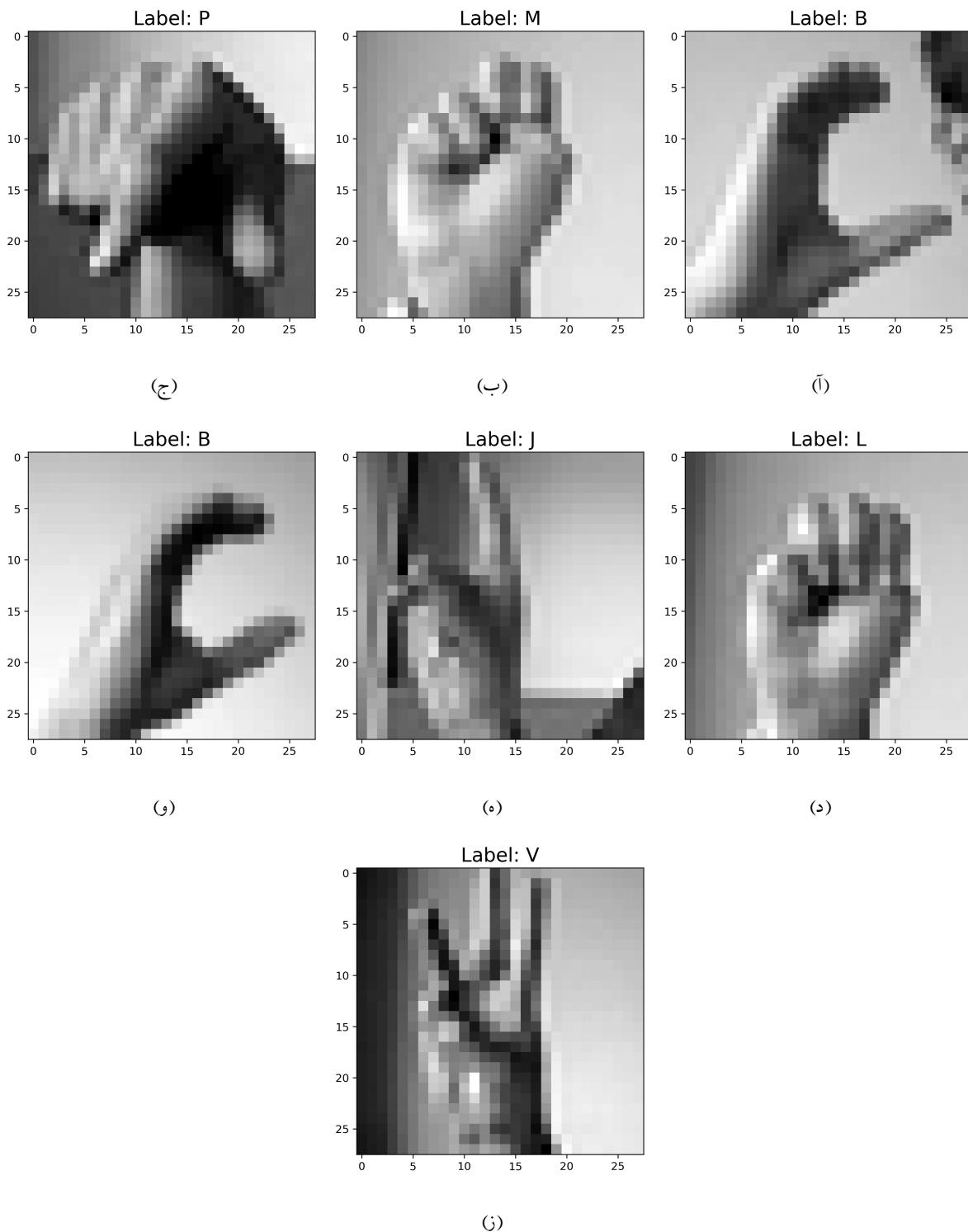
۱ شبکه MLP

داده‌های در اختیار ما برای این مسئله شامل اطلاعات تصاویر ۲۸ در ۲۸ به صورت پیکسلی است که هر پیکسل یک مقدار از ۰ تا ۲۵۵ را دارد و نشان‌دهنده آن پیکسل در طیف خاکستری است. برچسب داده‌ها نیز شامل یک عدد از ۱ تا ۲۴ است که به معنای حروف A تا Y باشد.

ابتدا ۷ عدد از این داده‌ها را به صورت تصادفی جدا کرده و آن‌ها را رسم می‌کنیم. این کار در شکل ۲ نشان داده شده است. ساختار شبکه MLP استفاده شده در شکل ۱ آمده است. این شبکه از دو لایه پنهان Dense و یک لایه Dropout به منظور کم کردن احتمال اورفیت شدن بھر می‌برد. تابع فعال‌سازی لایه‌های پنهان Relu می‌باشد و تابع لایه خروجی Softmax است. علت انتخاب تابع Softmax برای خروجی این است که در لایه خروجی به تعداد کلاس‌ها نورون وجود دارد که هر کدام یک مقدار را خروجی می‌دهند. این تابع بیشترین مقدار خروجی‌ها را به ۱ و سایر را به ۰ تبدیل می‌کند که نهایتاً به معنی انتخاب یک کلاس و انجام دسته‌بندی است.



شکل ۱:

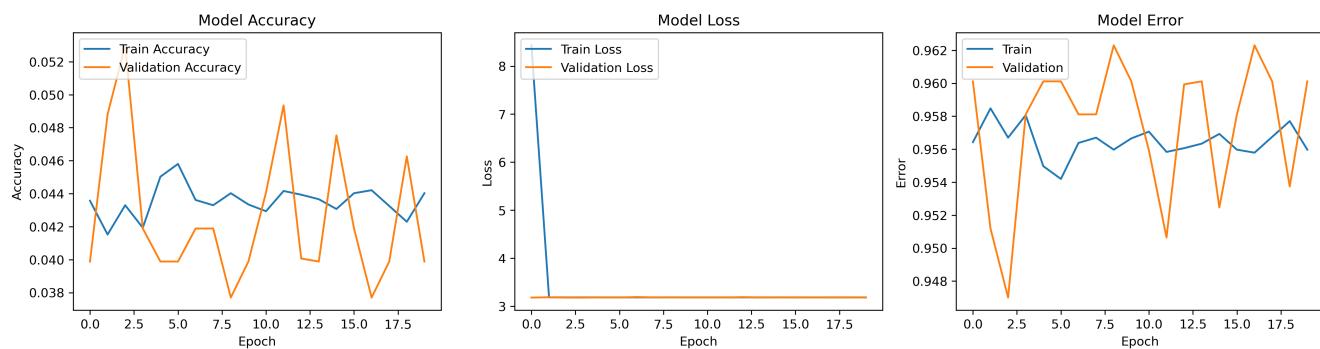


شکل ۲ : تصویر ۷ عدد از داده‌ها به صورت تصادفی

شبکه‌ی مورد نظر با استفاده از دو بهینه‌ساز و دو نرخ یادگیری تمرین داده خواهد شد که نتایج در ادامه خواهد آمد.

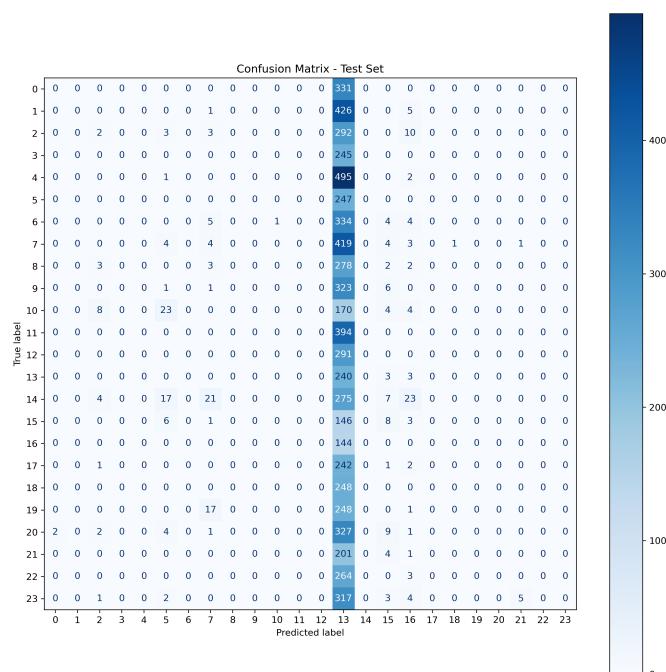
SGD ($\eta = 0.9$) ●

نمودار خطأ و دقت در شکل ۳ نشان داده شده است.



شکل ۳: نمودار خطأ، هزینه و دقت - بهینه‌ساز SGD با $\eta = 0.9$

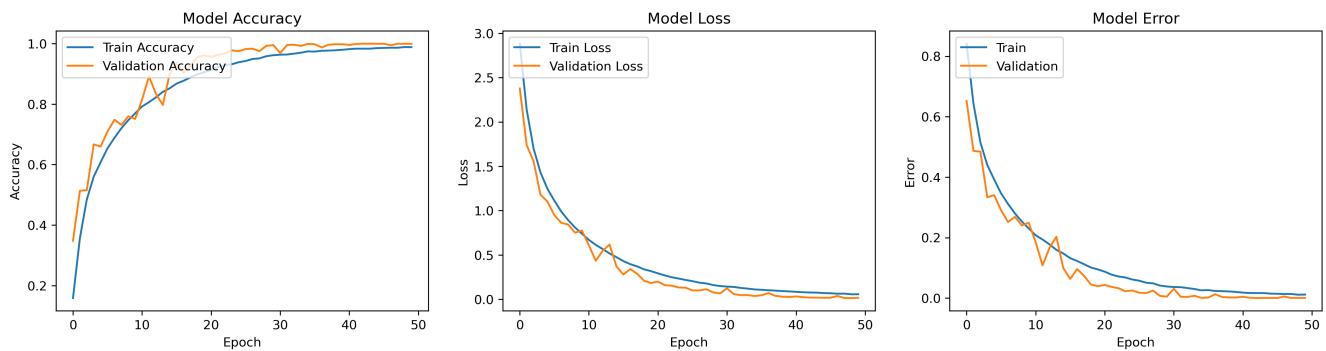
از جایی که نرخ یادگیری بالا است، دیده می‌شود که مدل به دقت خوبی نمیرسد. چنانچه این نرخ پایین باید می‌توان مدل را بهتر آموخت داد. ماتریس آشتفتگی در شکل ۴ آمده است.



شکل ۴: ماتریس آشتفتگی - بهینه‌ساز SGD با $\eta = 0.9$

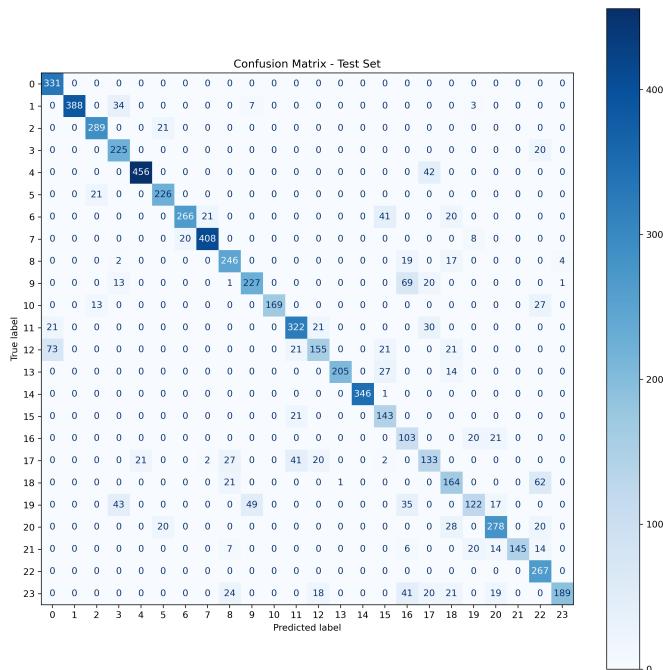
SGD ($\eta = 0.01$) ●

نمودار خطأ و دقت در شکل ۵ نشان داده شده است.



شکل ۵: نمودار خطأ، هزینه و دقت - بهینهساز SGD با $\eta = 0.01$

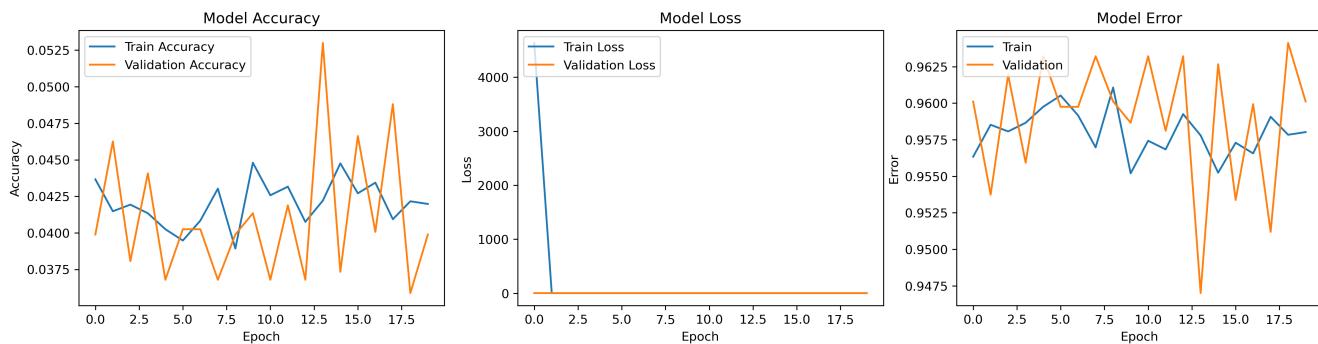
با کم کردن نرخ یادگیری، دیده می شود که حالا مدل در آموزش به همگرایی مناسبی می رسد. ماتریس آشفتگی در شکل ۶ آمده است.



شکل ۶: ماتریس آشفتگی - بهینهساز SGD با $\eta = 0.01$

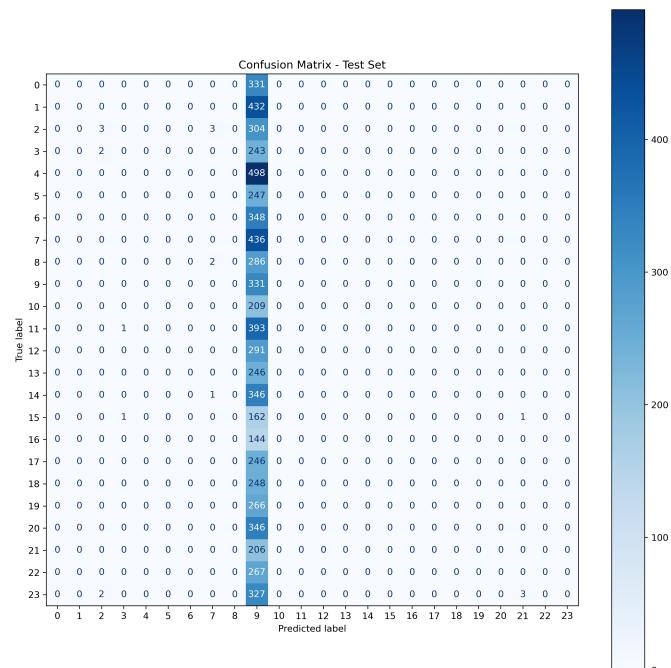
Adam ($\eta = 0.9$) ●

نمودار خطأ و دقت در شکل ۷ نشان داده شده است.



شکل ۷: نمودار خطأ، هزینه و دقت - بهینه‌ساز Adam با $\eta = 0.9$

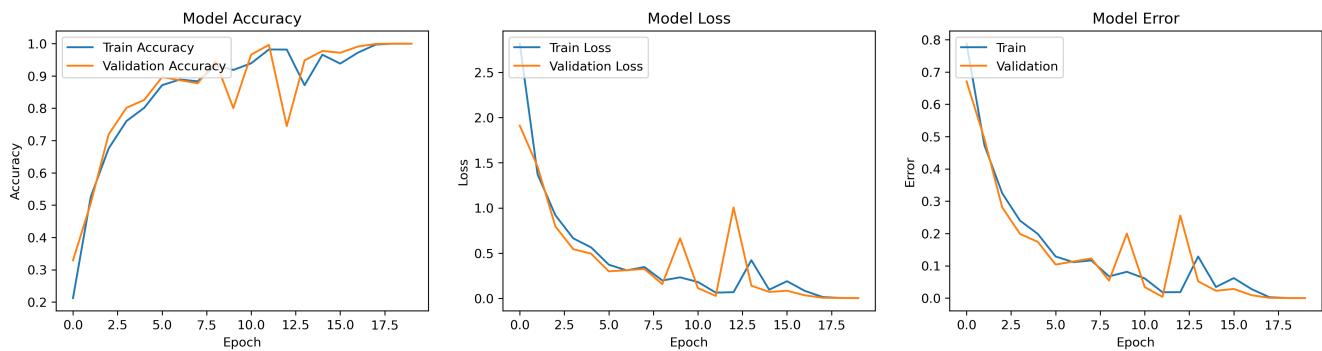
همانند حالت اول در بهینه‌ساز قبلی، این مدل نیز با نرخ یادگیری بالا همگرا نمی‌شود. ماتریس آشфтگی در شکل ۸ آمده است.



شکل ۸: ماتریس آشفتگی - بهینه‌ساز Adam با $\eta = 0.9$

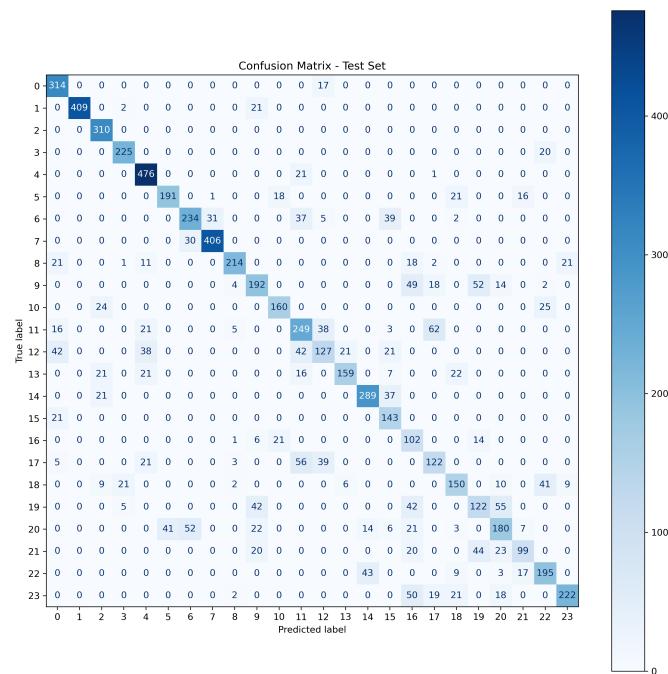
Adam ($\eta = 0.01$) ●

نمودار خط و دقت در شکل ۹ نشان داده شده است.



شکل ۹: نمودار خط، هزینه و دقت - بهینه‌ساز Adam با $\eta = 0.01$

از جایی که نرخ یادگیری بالا است، دیده می‌شود که مدل به دقت خوبی نمیرسد. چنانچه این نرخ پایین باید می‌توان مدل را بهتر آموخت داد. ماتریس آشфтگی در؟ آمده است.



شکل ۱۰: ماتریس آشфтگی - بهینه‌ساز Adam با $\eta = 0.9$

در نهایت، دقت هر حالت در ؟ به صورت تجمعی آمده است.

جدول ۱ : جمع‌بندی نتایج مدل‌ها برای پیش‌بینی دیابت

Optimizer	Learning Rate	Test accuracy
SGD	0.9	0.0365
SGD	0.9	0.8044
Adam	0.01	0.0465
Adam	0.01	0.7279

لازم به ذکر است که هنگام تمرین داده‌ها مشاهده شد که بهینه‌ساز Adam با نرخ Dropout کمتر نسبت به SGD بهتر همگرا می‌شود.

CNN شبکه ۲

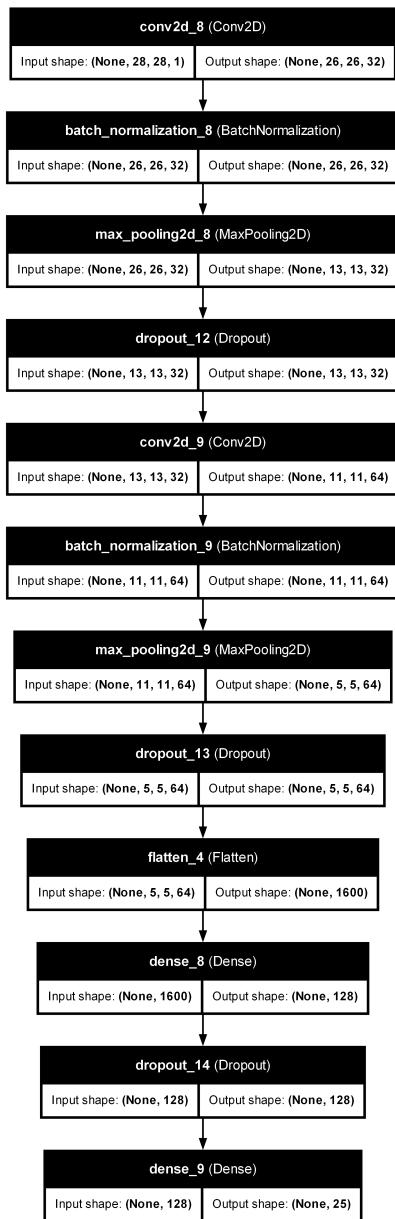
به منظور دسته‌بندی تصاویر، استفاده از مدل‌ها CNN در این قسمت بررسی می‌شود. ابتدا یک مدل با ساختار ؟ تشکیل داده می‌شود. این مدل از دو لایه پیچشی، لایه‌های انباشی، و یک لایه پنهان Dense Normalization تشکیل شده است. در خروجی از تابع Softmax استفاده شده است که در نهایت مقدار یکی از کلاس‌ها را با عدد ۱ و سایر را با صفر برمی‌گرداند.

برای بهینه‌سازی مدل از الگوریتم Adam استفاده می‌شود. دقت نهایی مدل برابر با ۰.۸۵۹۹ می‌باشد. نمودارهای خطأ و دقت در ؟ و ماتریس آشفتگی در صفحه ؟ آمده‌اند.

در این مدل برای جلوگیری از اورفیت شدن از لایه‌های Dropout نیز استفاده شده است. یک نوع از این لایه‌ها که معمولاً در شبکه‌های پیچشی موثرتر است لایه Block Dropout می‌باشد. در حالت قبلی مقدار تعدادی از نورون‌ها به صورت تصادفی نادیده گرفته می‌شود که برای یک تصویر دو بعدی ممکن است هرجای شکل باشد. اما در لایه Block Dropout، یک قسمت از تصویر پیکسل‌های آن مشابه هستند به صورت کامل، مثلاً ۳ در ۳، نادیده گرفته می‌شوند. به این ترتیب به جای حذف شدن مقدار پیکسل‌ها به تنها یک، درواقع یک ویژگی از تصویر حذف می‌شود و در شبکه‌های پیچشی که ویژگی‌های مختلف از تصویر را استخراج می‌کنند مفید است.

پیاده‌سازی این لایه به جای حالت قبلی، دقت را به ۰.۹۵۴۱ افزایش می‌دهد. نمودار خطأ و دقت در ؟ و ماتریس آشفتگی در صفحه ؟ آمده‌اند.

تجزیه فیلترها یا Kernel Factorization روشهایی است که در آن فیلترهای لایه‌های پیچشی به فیلترهای کوچک‌تر تجزیه می‌شوند. به عنوان مثال یک فیلتر ۳ در ۳ به دو فیلتر ۱ در ۳ و ۳ در ۱ تجزیه می‌شود و در دو مرحله اجرا می‌شود. در همین مثال، در حالت اول ۹ پارامتر وجود دارند که باید تنظیم شوند (k^2) اما در حالت دوم ۶ پارامتر ($2k$) وجود دارد. بنابراین هنگام آموزش مدل هزینه محاسباتی پایین می‌آید. همچنین با کم شدن پارامترها، احتمال اورفیت شدن مدل نیز کاهش پیدا می‌کند. نتایج پیاده‌سازی این روش در ؟ و صفحه ؟ آمده‌اند. دقت نهایی این مدل روی داده‌های تست ۰.۹۶۳۰ می‌باشد.

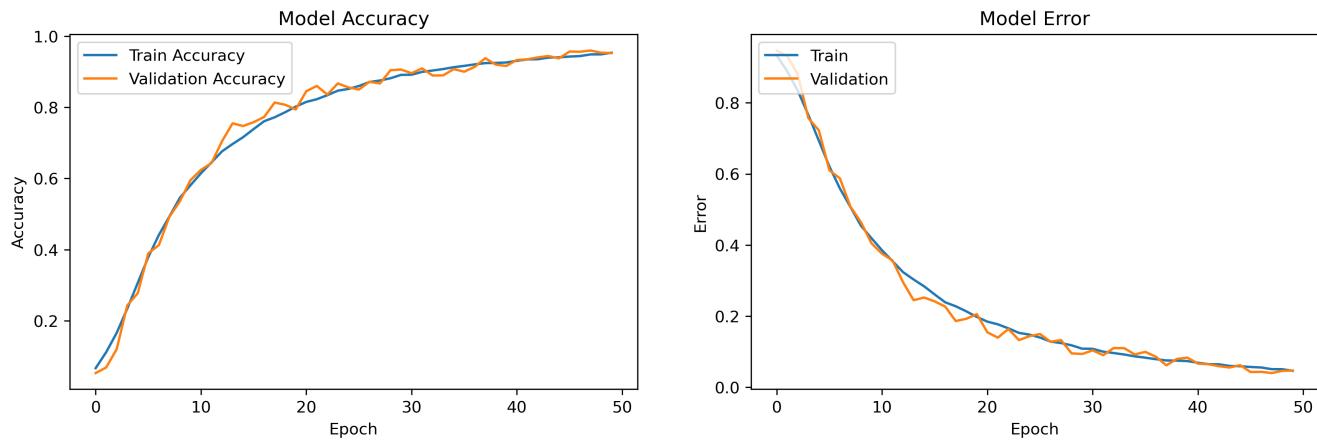


شکل ۱۱ :

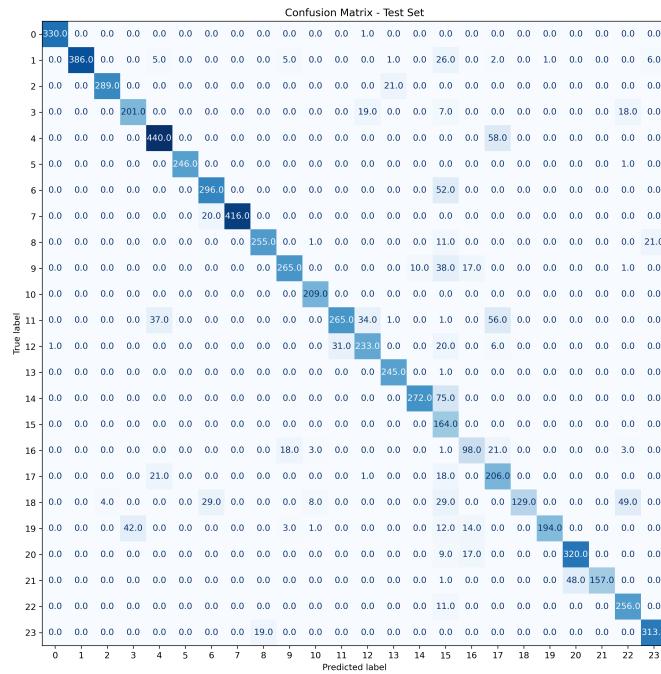
در نهایت، جمع‌بندی دقت حالت‌های مختلف در ؟؟ نوشته شده است.

جدول ۲ : جمع‌بندی نتایج مدل‌ها برای پیش‌بینی دیابت

Optimizer	Test accuracy
CNN	0.8599
CNN + Block Dropout	0.9541
CNN + Kernal Factorization	0.9630



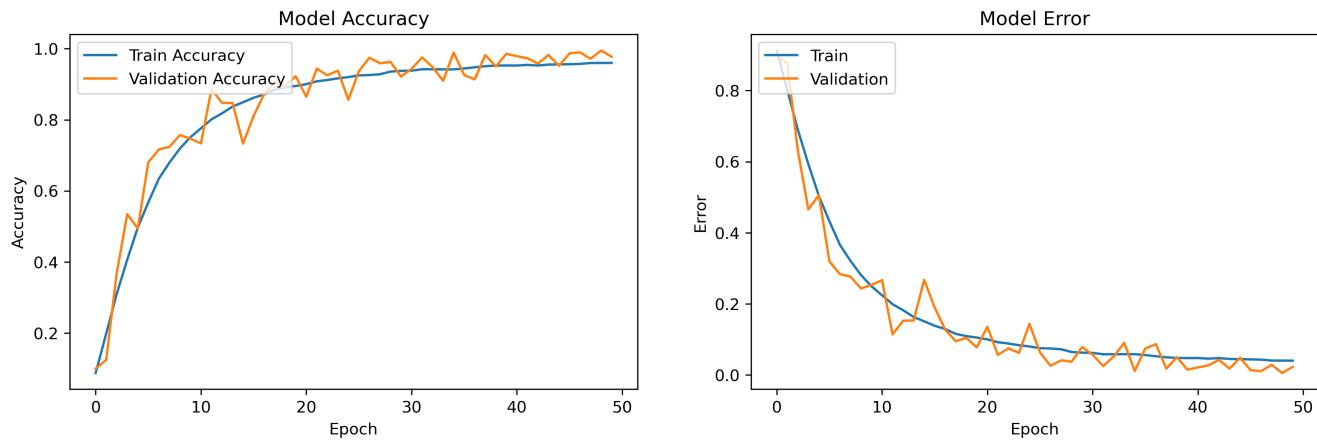
شکل ۱۲: نمودار خطای دقت - مدل CNN



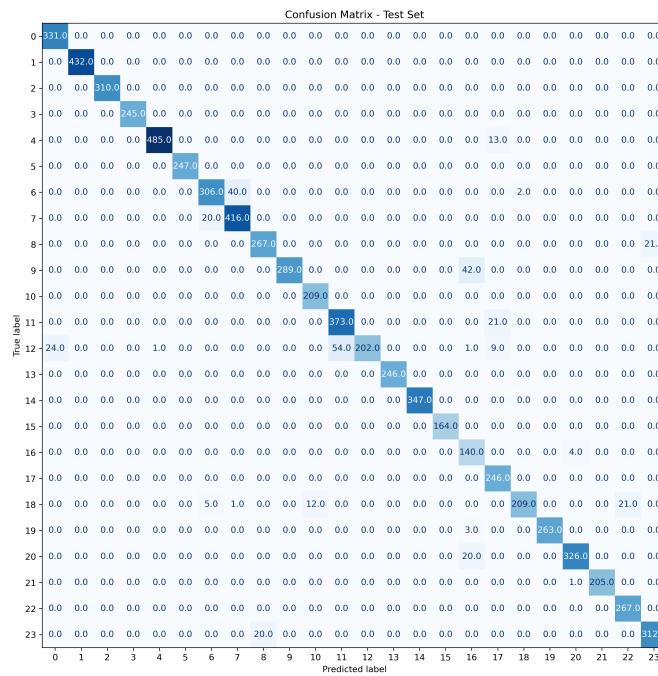
شکل ۱۳: ماتریس آشنازگی مدل CNN

۳ آموزش انتقالی

در این قسمت باید ابتدا شبکه‌ای با ساختار ResNet50 با استفاده از داده‌های دیتاست EMNIST پیش‌تریت شوند و سپس با استفاده از همین مدل که ضرایب آن پیش‌تریت شده‌اند، تریت را با استفاده از داده‌های مسئله انجام داد. این کار نه تنها موقوعی که داده به اندازه کافی وجود ندارد به کار می‌آید، بلکه از جایی که از ضرایب پیش‌تریت شده استفاده می‌شوند، روند تریت دارای هزینه محاسباتی کمتری خواهد بود و سریع‌تر به دقت مورد نظر خواهد رسید. از همین رو تعدادی از مدل‌های پیش‌تریت شده مانند همین معماري شبکه در



شكل ١٤: نمودار خطأ و دقت - مدل CNN با Block Dropout

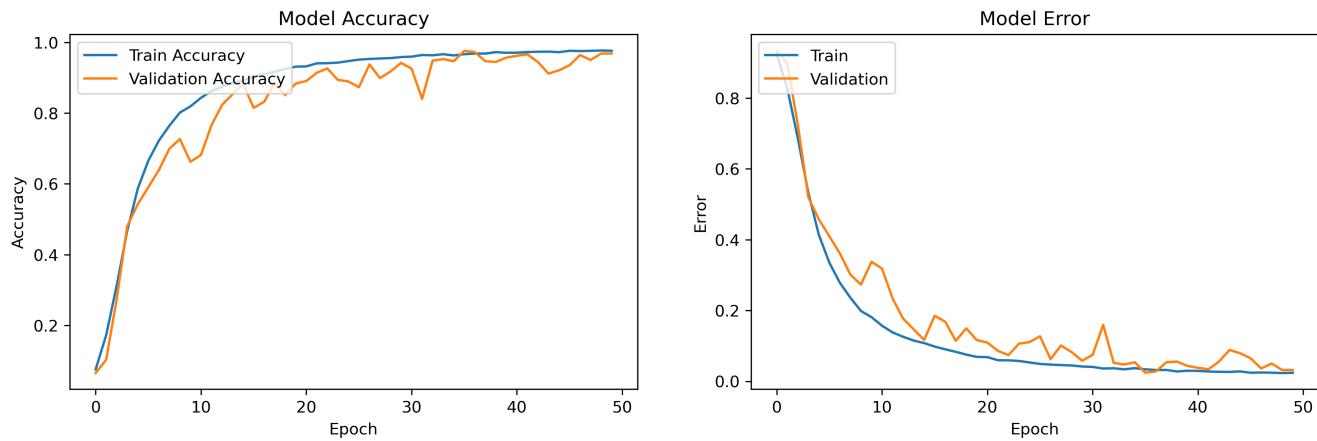


شکل ۱۵: ماتریس آشفتگی - مدل CNN با Block Dropout

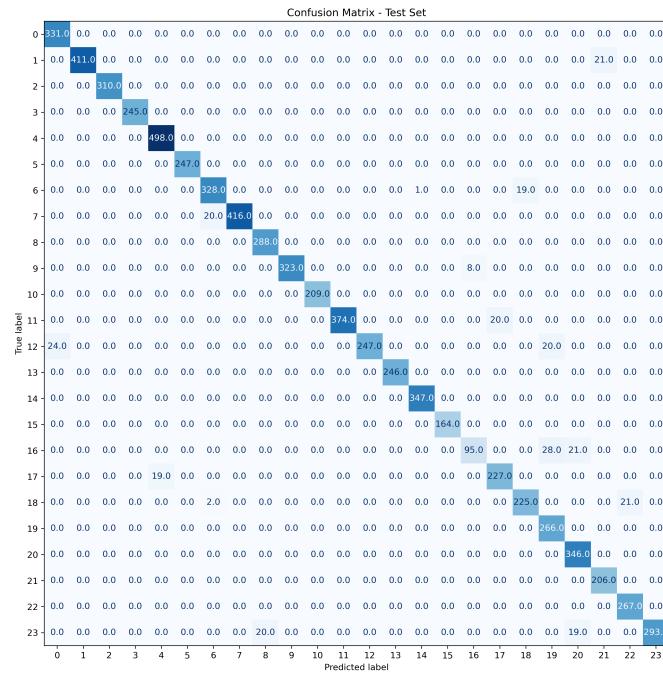
دسترس عموم بوده و می‌توان در مسائل استفاده کرد.

داده‌های دیتاست EMNIST مانند داده‌های مسئله عکس‌های تک کاناله ۲۸ در ۲۸ هستند که حاوی کاراکترهای مختلف دست‌نویسنده‌اند. این دیتاست انواع مختلف دارد. در این مسئله از دیتاست Balanced استفاده شده که در آن کل کلاس‌ها حاوی سه‌پل‌های یکسانی هستند. این دیتاست به صورت فایل CSV در کد آپلود می‌شود.

وروودی شبکه ResNet50 باید به اندازه $(n, n, 3)$ باشد. به این معنی که تصویر مورد نظر باید دارای سه کانال باشد. همچنین حداقل اندازه تصاویر باید 32×32 در 32 باشند. به این ترتیب قبل از اینکه تصاویر دیتاست EMNIST را بتوان به شبکه داد، باید اندازه آن را تغییر داد. این کار به کمک تابعی به نام `resize` که در کد نوشته شده است انجام می‌شود. از جایی که هم شبکه پیچیده بوده و داده‌ها نیز زیاد هستند، این



شکل ۱۶: نمودار خطوط دقت - مدل CNN با تجزیه فیلترها



شکل ۱۷: ماتریس آشفتگی - مدل CNN با تجزیه فیلترها

تابع عکس‌های تک کاناله ۲۸ در ۲۸ را گرفته و داده را به یک داده سه کاناله ۳۲ در ۳۲ تبدیل می‌کند تا کمترین محاسبات ممکن را داشته باشیم. لازم به ذکر است که سه کانال مورد نظر در تصویر نهایی دارای مقدار یکسان بوده و همان تک کانال تصویر اولیه هستند. سپس این تصاویر در بسته‌های دلخواه (در این کد ۳۲ تایی) با استفاده از کتابخانه `ImageDataGenerator` قرار داده می‌شوند تا به مدل داده شوند.