

به نام ...

جمال الدین دمیرچی

983212093

پروژه هوش محاسباتی:

ما در این پروژه تلاش کرده‌ایم تا با استفاده از کتابخانه **torch**، دو شبکه عصبی برای دسته‌بندی تصاویر دیتاست ایجاد کنیم.

در ابتدا پارامترهای مورد نظر تعریف میشود:

- **(Neuron)**: هر لایه از شبکه عصبی شامل تعدادی نورون است که ورودی‌ها را از تعداد نورون‌های لایه‌ی قبل دریافت می‌کند.
- **(Hidden Layer)**: این لایه‌ها در میان لایه ورودی و لایه خروجی قرار دارند و محاسباتی جهت ساده‌سازی و کاهش اندازه مقادیر صورت می‌گیرد. پس از انجام محاسبات، وزن‌ها از لایه ورودی به لایه‌ی بعدی، که ممکن است یک لایه پنهان دیگر یا لایه خروجی باشد، منتقل می‌شوند.
- **(Learning Rate)**: این پارامتر یک ضریب ثابت است که استفاده از آن امکان تغییر وزن‌ها در لایه‌ها را فراهم می‌کند. این مقدار بر اساس تست انواع مقادیر به طور عمومی انتخاب می‌شود.
- **(Optimization Function)**: این تابع وزن‌ها را پس از محاسبه دریافت کرده و برای بهترین نگاشت به یک کلاس، از یک مقدار کمتر استفاده می‌کند.
- **(Loss Function)**: به طور پیش‌فرض، تابع **CrossEntropy** بر اساس نیاز مسئله برای محاسبه خطا استفاده شده است.

در شبکه اول، با استفاده از یک لایه پنهان با ۳۲ نورون، ساختاری بنیادی تدارک دیده شد. برای بهینه‌سازی، بهینه‌ساز Adam استفاده شد و تابع خسارت CrossEntropy به کار رفت.

نتایج به دست آمده به ازای مقادیر مختلف به شکل زیر است:

Batch size	Learning Rate	Epoch	Accuracy
16	0.01	100	66%
32	0.01	200	69%
64	0.01	100	68%
128	0.01	100	71%
64	0.001	50	70%
128	0.001	50	69%
128	0.001	100	69%

در شبکه دوم، در قسمت نخست، یک لایه کانولوشن با بعدهای دوگانه تعبیه شده است که توجه به نوع داده‌ها (یعنی تصاویر)، موجب بهبود عملکرد شبکه می‌گردد. برای بهره‌برداری از داده‌ها در این لایه، لازم است آن‌ها از حالت خطی به نمایشی ساختارمند با اندازه‌های معین و تعداد کانال‌های مشخص شکل‌گیری کنند که این فرایند در تابع forward رخ می‌دهد. اضافه شدن یک لایه pool بلافاصله پس از لایه کانولوشن بهبود قابل توجهی در نتیجه ایجاد کرده است.

چشمگیر حضور و عدم حضور این لایه را نمایان ساخته‌اند.

پس از گذر از لایه pool، داده‌ها دوباره به فرم خطی درآمده تا برای پردازش‌های بعدی به شبکه‌ای که در مرحله اول ساخته شده بود، انتقال یابند. ابعاد داده‌های خروجی به اندازه‌ها و ابعاد کرنل بستگی دارند.

زمان آموزش شبکه به شکل قابل ملاحظه‌ای افزایش یافته است که این امر ناشی از پیاده‌سازی لایه‌های متنوع است. در مقابل، دقت شبکه بسیار بهبود یافته و زمان تست داده‌ها تغییر محسوسی نکرده که در کلیت به معنی افزایش کارایی شبکه است.

با افزایش اندازه خروجی لایه کانولوشن و ابعاد kernel، نتایج بهبود پیدا کرده‌اند. با افزایش بیش از اندازه خروجی، برنامه به دلیل محدودیت‌های حافظه، قابل اجرا نبود. ابعاد بزرگتر برای کرنا نیز نتایج بهتری را به دنبال نداشته‌اند. تغییر learning rate به ۰.۱ باعث پیشرفت چشمگیری در نتایج شد.

تغییرات سایر پارامترها تأثیری مشابه با بخش اول داشته‌اند.

باتوجه به تجزیه و تحلیل این نتیجه رسیدیم که نگاهی کلی به پارامترهای موثر بر کیفیت نهایی شبکه عصبی، نقش مهمی در بهبود طراحی و استقرار مدل‌های یادگیری عمیق دارد.

نتایج بدست آمده را مشاهده میکنید:

Learning Rate	Kernel Size	Parameters	Accuracy
0.01	3	841784	0.71
0.01	3	81784	0.77
0.01	5	82936	0.79
0.1	5	82936	0.83