

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

گزارش کار تمرین ۴-۳ - یادگیری عمیق

عنوان تمرین :

طراحی و پیاده‌سازی شبکه عصبی تمام متصل (FCN)

برای طبقه‌بندی تصاویر CIFAR-10

دانشجو : محمدداود وهاب رجائی

شماره دانشجویی : ۴۰۴۱۴۱۹۰۴۱

استاد محترم : دکتر کیوان راد

حل تمرین : دکتر گلیزاده

فهرست مطالب

۱	چکیده
۱	مقدمه و بیان مسئله
۲	روش‌شناسی (Methodology)
۲	۱. آماده‌سازی داده‌ها
۳	۲. معماری مدل پیشنهادی
۳	۳. پیکربندی آموزش
۴	نتایج تجربی (Experimental Results)
۴	۱. روند آموزش و اعتبارسنجی
۴	۲. ارزیابی روی داده‌های تست
۵	۳. تحلیل جزئیات کلاس‌ها
۶	بحث و تحلیل (Discussion)
۶	۱. تحلیل ماتریس در هم‌یختگی
۶	۲. محدودیت‌های معماری FCN
۷	نتیجه‌گیری و کارهای آتی

چکیده

در این پروژه، یک شبکه عصبی تمام متصل (Fully Connected Neural Network) برای طبقه‌بندی تصاویر رنگی مجموعه داده CIFAR-10 طراحی و آموزش داده شد. هدف اصلی، بررسی کارایی معماری‌های MLP بر روی داده‌های تصویری و تأثیر تکنیک‌های تنظیم (Regularization) با استفاده از لایه‌های Batch Normalization، Dropout و تابع فعال‌ساز LeakyReLU پیاده‌سازی شد و پس از ۲۵ دور آموزش (Epoch)، به دقت نهایی ۵۵.۵۵٪ روی داده‌های تست دست یافت. نتایج نشان داد که اگرچه شبکه‌های تمام متصل به دلیل نادیده گرفتن اطلاعات مکانی تصاویر دارای محدودیت هستند، اما با تنظیم دقیق هایپرپارامترها می‌توانند عملکردی بسیار بالاتر از حدس تصادفی (۱۰٪) ارائه دهند.

مقدمه و بیان مسئله

طبقه‌بندی تصاویر یکی از مسائل بنیادین در بینایی ماشین است. در حالی که شبکه‌های کانولوشنی (CNN) استاندارد طلایی برای این وظیفه محسوب می‌شوند، درک عملکرد شبکه‌های تمام متصل (FCN) به عنوان یک پایه (Baseline) ضروری است. در این تمرین، چالش اصلی تبدیل داده‌های تصویری دو بعدی به بردارهای یک بعدی و آموزش یک شبکه عمیق برای یادگیری الگوهای بصری بدون استفاده از فیلترهای کانولوشنی است. ما از دیتابست CIFAR-10 شامل ۱۰ کلاس (هوایپیما، خودرو، پرنده، گربه، گوزن، سگ، قورباغه، اسب، کشتی و کامیون) استفاده کردیم که شامل ۶۰,۰۰۰ تصویر رنگی 32×32 پیکسل است.

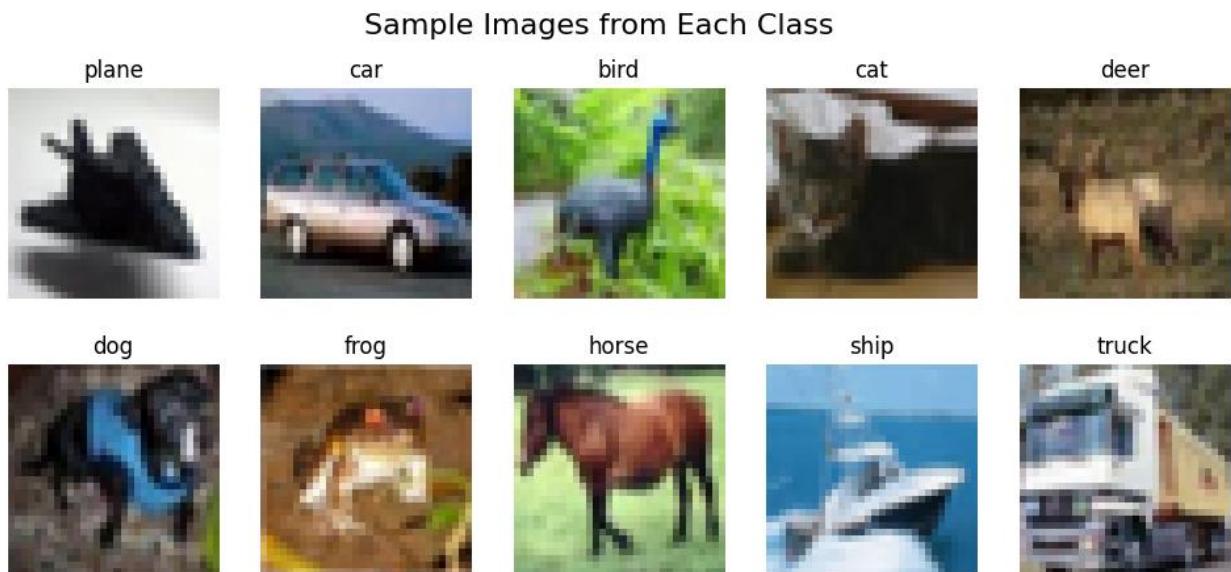
روش‌شناسی (Methodology)

۱. آماده‌سازی داده‌ها

- پیش‌پردازش (Preprocessing) : تمامی تصاویر ابتدا به تانسورهای PyTorch تبدیل شدند. جهت کمک به همگرایی سریع‌تر مدل و پایداری گردیدن، نرمال‌سازی (Normalization) با میانگین ۰.۵ و انحراف معیار ۰.۵ روی کانال‌های رنگی اعمال شد تا مقادیر پیکسل‌ها در بازه $[1, -1]$ قرار گیرند.
- تقسیم‌بندی داده‌ها :

 - داده‌های آموزش (Training Set) : ۴۲,۵۰۰ تصویر (۸۵٪)
 - داده‌های اعتبارسنجی (Validation Set) : ۷,۵۰۰ تصویر (۱۵٪) جهت تنظیم هایپرپارامترها.
 - داده‌های تست (Test Set) : ۱۰,۰۰۰ تصویر جهت ارزیابی نهایی.

- بارگذاری : از DataLoader با اندازه دسته (Batch Size) برابر با ۱۲۸ استفاده شد.



شکل ۱ : نمونه تصاویر مجموعه داده CIFAR-۱۰ پس از نرمال‌سازی

۲. معماری مدل پیشنهادی

مدل طراحی شده یک شبکه چندلایه پرسپترون (MLP) با مشخصات زیر است :

۱. لایه ورودی: تصویر $32 \times 32 \times 3$ باز شده (Flatten) و به یک بردار با ابعاد

۳۰۷۲ تبدیل می‌شود.

۲. لایه مخفی اول: لایه خطی با ۵۱۲ نورون + Batch Normalization + LeakyReLU ($0, 1$) + Dropout ($0, 0.5$).

۳. لایه مخفی دوم: لایه خطی با ۲۵۶ نورون + Batch Normalization + Dropout (0.5) + LeakyReLU ($0, 1$).

۴. لایه خروجی: لایه خطی با ۱۰ نورون (متناظر با ۱۰ کلاس).

- تعداد کل پارامترهای قابل آموزش: $1,708,810$ پارامتر.
- دلیل انتخاب: استفاده از LeakyReLU برای جلوگیری از مشکل "مرگ نورون‌ها" و استفاده از Dropout سنگین (0.5) برای مقابله با بیشبرازش شدید در شبکه‌های تمام متصل.

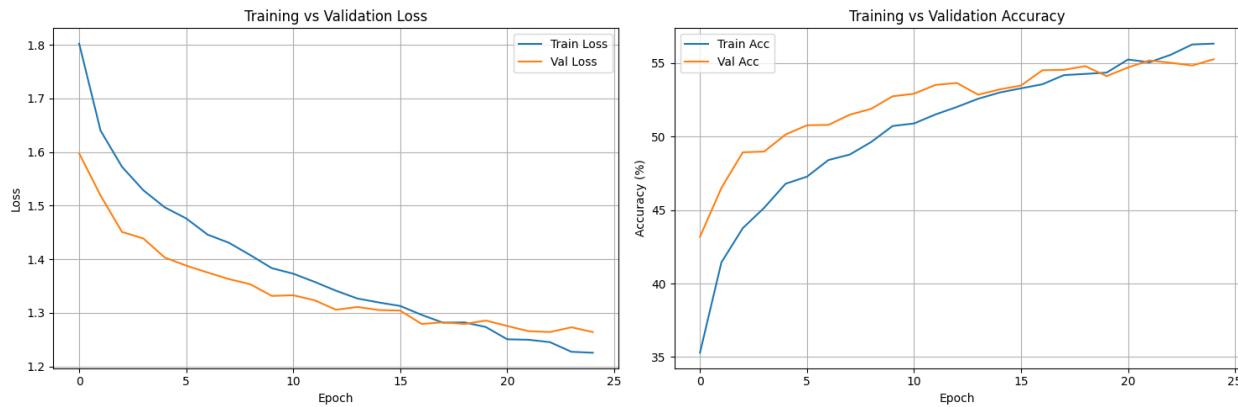
۳. پیکربندی آموزش

- تابع هزینه: CrossEntropyLoss
- بهینه‌ساز Adam: با نرخ یادگیری اولیه 0.001 .
- زمان‌بندی (Scheduler): استفاده از ReduceLROnPlateau برای کاهش نرخ یادگیری در صورت عدم بهبود دقت اعتبارسنجی.
- تعداد اپک: ۲۵ دور کامل.

نتایج تجربی (Experimental Results)

۱. روند آموزش و اعتبارسنجی

فرآیند آموزش مدل نشان دهنده همگرایی مناسب بود. همانطور که در نمودارهای زیر مشاهده می شود، نرخ خطای Loss روی داده های اعتبارسنجی همگام با داده های آموزشی کاهش یافت که نشان دهنده عملکرد مؤثر Dropout در جلوگیری از Overfitting است. مدل در اپک ۲۵ به بهترین دقیقت اعتبارسنجی ۵۵.۲۷٪ دست یافت و مدل در این نقطه ذخیره شد.



شکل ۲ : منحنی های یادگیری؛ روند تغییرات Loss و Accuracy در طول ۲۵ اپک

۲. ارزیابی روی داده های تست

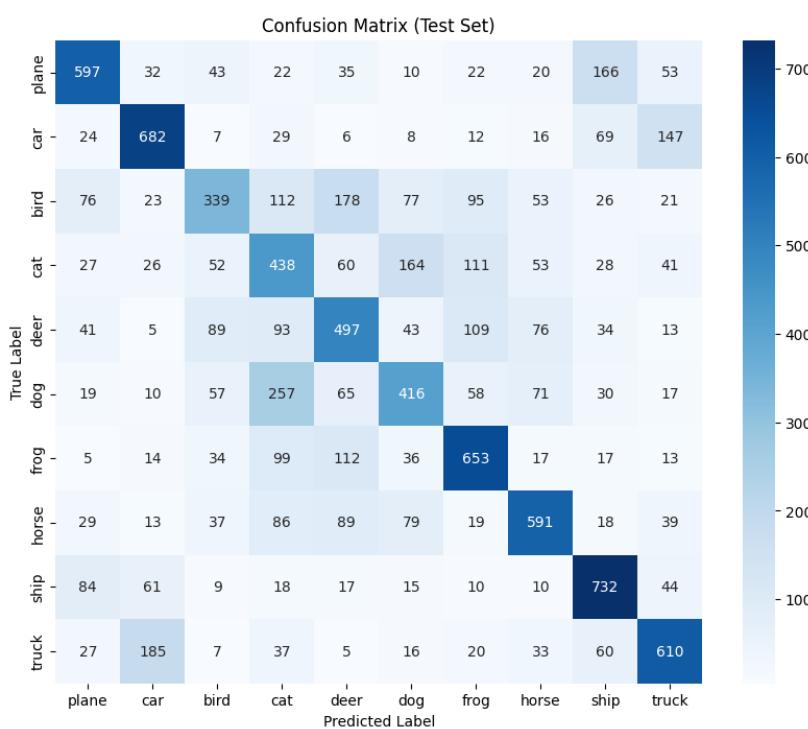
پس از پایان آموزش، مدل نهایی روی ۱۰,۰۰۰ تصویر دیده نشده تست ارزیابی شد :

- دقت نهایی (Test Accuracy) : ۵۵.۵۵٪
- عملکرد نسبت به پایه : با توجه به وجود ۱۰ کلاس، حدس تصادفی دقیقی برابر ۱۰٪ دارد. مدل ما ۴۵٪ بهبود نسبت به حالت تصادفی داشته است.

۳. تحلیل جزئیات کلاس‌ها

با بررسی گزارش طبقه‌بندی (Classification Report)، نتایج زیر حاصل شد:

- بهترین عملکرد: کلاس‌های "کشتی" (Ship) با F-Score ۰.۶۷ و "خودرو" (Car) با ۰.۶۷.
- ضعیفترین عملکرد: کلاس‌های "گربه" (Cat) با F-Score ۰.۴۰ و "پرنده" (Bird) با ۰.۴۱.



شکل ۳ : ماتریس درهم‌ریختگی نتایج تست

بحث و تحلیل (Discussion)

۱. تحلیل ماتریس درهم ریختگی

با دقت در ماتریس درهم ریختگی (شکل ۳) مشاهده می شود که بیشترین اشتباهات مدل بین کلاس های زیر رخ داده است:

۱. گربه و سگ: مدل اغلب این دو حیوان را با هم اشتباه می گیرد. دلیل این امر شباهت ساختاری (چهارپا بودن، داشتن گوش و دم) و عدم توانایی شبکه FCN در استخراج ویژگی های ظریف محلی است.

۲. پرنده و هوایپیما: اشتباه بین این دو کلاس اغلب به دلیل رنگ پس زمینه (آسمان آبی) رخ می دهد. از آنجا که شبکه تمام متصل به شدت به مقادیر پیکسل ها و رنگ وابسته است، بافت کلی تصویر را ملاک قرار می دهد.

۲. محدودیت های معماری FCN

دستیابی به دقت ۵۵٪ برای یک شبکه تمام متصل روی دیتاست پیچیده ای مثل CIFAR-10 نتیجه ای قابل قبول است.

دلیل عدم دستیابی به دقت های بالاتر (مانند ۹۰٪ در CNN ها)، از بین رفتن اطلاعات مکانی (Spatial Information) هنگام مسطح کردن (Flatten) تصویر است. شبکه FCN رابطه همسایگی بین پیکسل ها را درک نمی کند و نمی تواند ویژگی های مستقل از مکان (Translation Invariance) را یاد بگیرد.

نتیجه‌گیری و کارهای آتی

در این تمرین، یک خط لوله کامل یادگیری عمیق با استفاده از شبکه تمام متصل پیاده‌سازی شد. استفاده از تکنیک‌های Dropout و Batch Normalization پایداری مدل را تضمین کرد و دقت ۵۵.۵۵٪ حاصل شد. نتایج نشان داد که مدل در تشخیص وسایل نقلیه موفق‌تر از حیوانات عمل می‌کند و محدودیت اصلی، نوع معماری شبکه است.

پیشنهادات برای کارهای آتی :

استفاده از شبکه‌های کانولوشنی (CNN) برای حفظ اطلاعات مکانی و استخراج ویژگی‌های بصری عمیق‌تر.
استفاده از Data Augmentation (مانند چرخش، برش و تغییر رنگ تصادفی) برای افزایش تنوع داده‌های آموزشی و بهبود تعیین‌پذیری مدل.

با تشکر از نگاه پر مهر شما