

فراموشی فاجعه‌بار

علی احمدی اسفیدی

۸ خرداد ۱۴۰۴

۱ مقدمه

در این گزارش به بررسی دو نسخه از پیاده‌سازی یادگیری متوالی در مسئله طبقه‌بندی ارقام مجموعه MNIST می‌پردازیم. نسخه اول بدون هیچ مکانیسم محافظتی آموزش می‌بیند و نسخه دوم از روش Orthogonal Gradient Descent برای جلوگیری از فراموشی فاجعه‌بار بهره می‌برد.

۲ نسخه اول: آموزش ساده

۱.۲ داده‌ها و آماده‌سازی

- مجموعه MNIST به دو زیرمجموعه تقسیم شد:

- Task A: ارقام 0 تا 4

- Task B: ارقام 5 تا 9

- تبدیل‌ها: تبدیل به تنسور و نرمال‌سازی با میانگین 0.1307 و انحراف معیار 0.3081.

۲.۲ معماری مدل

```
model = nn.Sequential(  
    nn.Flatten(),  
    nn.Linear(28*28, 100), nn.ReLU(),  
    nn.Linear(100, 100), nn.ReLU(),  
    nn.Linear(100, 10)  
)
```

۳.۲ تنظیمات آموزش

- بهینه‌ساز: SGD با نرخ یادگیری $1e^{-3}$

- تابع هزینه: CrossEntropyLoss

- تعداد epoch در هر فاز: 3

۴.۲ نتایج

- قبل از Task B (فاز Task A): دقت تست Task A معادل 97.55%
- بعد از Task B: دقت تست Task A سقوط می کند و به 0% می رسد.

۵.۲ تحلیل

این پدیده که مدل پس از یادگیری یک وظیفه جدید تمام دانش وظیفه قبلی را از دست می دهد، فراموشی فاجعه بار نامیده می شود. در این پیاده سازی هیچ گونه مکانیسمی برای حفظ دانش قبلی وجود ندارد.

۳ نسخه دوم: بوسیله OGD

۱.۳ Orthogonal Gradient Descent

۱. محاسبه گرادیان ∇L روی نمونه های Task A
۲. اعمال فرآیند گرام-شمیدت برای استخراج بردارهای عمود نرمال شده
۳. در هنگام آموزش Task B، پروجکت گرادیان کلی مدل بر زیرفضای عمود بر این بردارها

۲.۳ نتایج

- پس از Task A: دقت تست Task A برابر 93.75%
- تعداد جهات ذخیره شده: 32
- پس از Task B: دقت تست Task A کاهش جزئی داشته و به 93.44% می رسد.

۴ مقایسه و تحلیل نهایی

معیار	نسخه اول	نسخه دوم
دقت Task A قبل از Task B	97.55%	92.90%
دقت Task A پس از Task B	0.00%	92.16%
شدت فراموشی فاجعه بار	شدید	تقریباً رفع شده

نسخه دوم بوسیله OGD توانسته است حافظه عملی از Task A را در طول یادگیری Task B حفظ کند.

۵ منابع

- *PyTorch: An Imperative Style, High-Performance Deep Learning Library*
- *Orthogonal Gradient Descent for Continual Learning (Farajtabar, et al.)*