**فصل 22 - شبکه‌های عصبی برای داده‌های بدون ساختار**

**22.0 مقدمه**

در فصل قبل، ما بر روی دستور العمل‌های شبکه عصبی برای داده‌های ساخت یافته، یعنی داده‌های جدولی تمرکز کردیم. بسیاری از بزرگترین پیشرفت‌ها در چند سال گذشته در واقع شامل استفاده از شبکه‌های عصبی و یادگیری عمیق برای داده‌های بدون ساختار، مانند متن یا تصاویر بوده است. کار با این مجموعه داده‌های بدون ساختار کمی متفاوت از کار با منابع داده‌های ساخت یافته است.

یادگیری عمیق به‌ویژه در فضای داده‌های بدون ساختار قدرتمند است، جایی که تکنیک‌های یادگیری ماشینی «کلاسیک» (مانند درخت‌های تقویت‌شده) معمولاً نمی‌توانند تمام پیچیدگی‌ها و تفاوت‌های ظریف موجود در داده‌های متنی، صدا، تصاویر، ویدیوها و غیره را ثبت کنند. ما در این فصل استفاده از یادگیری عمیق را به طور خاص برای داده‌های متن و تصویر بررسی خواهیم کرد.

در یک فضای یادگیری تحت نظارت برای متن و تصاویر، وظایف فرعی یا «انواع» یادگیری بسیاری وجود دارد. در زیر چند نمونه آورده شده است (اگرچه این فهرست جامعی نیست):

* طبقه بندی متن یا تصویر (مثال: طبقه بندی اینکه آیا یک تصویر، تصویر یک هات داگ است یا خیر)
* انتقال یادگیری (مثال: استفاده از یک مدل زمینه‌ای از پیش آموزش‌دیده‌شده مانند BERT و تنظیم دقیق آن در یک کار برای پیش بینی هرزنامه بودن یا نبودن ایمیل)
* تشخیص اشیا (به عنوان مثال: شناسایی و طبقه بندی اشیاء خاص در یک تصویر)
* مدل‌های مولد (مثال: مدل‌هایی که متن را بر اساس یک ورودی مشخص مانند مدل‌های GPT تولید می‌کنند)

با افزایش محبوبیت یادگیری عمیق و کالایی شدن روزافزون آن، راه حل‌های منبع باز و سازمانی برای مقابله با این موارد استفاده آسان تر شده اند. در این فصل، ما از چند کتابخانه کلیدی به عنوان نقطه‌ی ورود خود برای انجام این وظایف یادگیری عمیق استفاده خواهیم کرد. به طور خاص، ما از کتابخانه‌های PyTorch، Torchvision و Transformers در Python برای انجام مجموعه‌ای از وظایف و عملکردها در داده‌های متنی و تصویری ML استفاده می‌کنیم.

**22.1 آموزش شبکه عصبی برای طبقه بندی تصاویر**

**مسئله**

شما باید یک شبکه عصبی طبقه بندی تصویر را آموزش دهید.

**راه‌حل**

از یک شبکه عصبی کانولوشنال[[1]](#footnote-1) در PyTorch استفاده کنید:



**بحث**

شبکه‌های عصبی کانولوشنال معمولاً برای عملکردهایی در تشخیص تصویر و بینایی کامپیوتری استفاده می‌شوند. آنها معمولاً از لایه‌های کانولوشن، لایه‌های ترکیبی و یک لایه کاملاً متصل تشکیل شده‌اند.

هدف از لایه‌های کانولوشن، یادگیری ویژگی‌های مهم تصویر است. لایه‌های کانولوشن با اعمال یک فیلتر در ناحیه خاصی از تصویر (به اندازه کانولوشن) کار می‌کنند. وزن‌های این لایه سپس یاد می‌گیرند که ویژگی‌های تصویری خاص را که در کار طبقه‌بندی حیاتی هستند، تشخیص دهند. به عنوان مثال، اگر مدلی را آموزش دهیم که دست فرد را تشخیص دهد، ممکن است فیلتر تشخیص انگشتان را بیاموزد.

هدف لایه ادغام معمولاً کاهش ابعاد ورودی‌ها از لایه قبلی است. این لایه همچنین از فیلتر اعمال شده بر روی بخشی از ورودی استفاده می‌کند، اما هیچ فعال سازی ندارد. در عوض، ابعاد ورودی را با انجام حداکثر ادغام (جایی که پیکسل را در فیلتر با بالاترین مقدار انتخاب می‌کند) یا ادغام متوسط (که در آن به طور متوسط از پیکسل‌های ورودی استفاده می‌شود) کاهش می‌دهد.

در نهایت، لایه‌ی کاملا متصل را می‌توان با چیزی مانند تابع فعال سازی softmax برای ایجاد یک طبقه بندی باینری استفاده کرد.

**همچنین ببینید:**

* [شبکه‌های عصبی کانولوشنال (CNN)](https://www.ibm.com/topics/convolutional-neural-networks)

**22.2 آموزش شبکه عصبی برای طبقه بندی متن**

**مسئله**

برای طبقه بندی داده‌های متنی باید یک شبکه عصبی آموزش دهید.

**راه‌حل**

از یک شبکه عصبی PyTorch استفاده کنید که اولین لایه آن به اندازه دایره لغات شما باشد:

**بحث**

برخلاف تصاویر، داده‌های متنی ذاتا غیر عددی هستند. قبل از آموزش یک مدل، باید متن را به یک نمایش عددی تبدیل کنیم که مدل بتواند از آن استفاده کند تا بفهمد کدام کلمات و ترکیب‌های کلمه برای طبقه‌بندی مهم هستند. در این مثال، ازCountVetorizer در scikit-learn برای رمزگذاری واژگان به عنوان بردار به اندازه‌ی کل واژگان استفاده می‌کنیم، جایی که هر کلمه به یک شاخص خاص در بردار اختصاص داده می‌شود، و مقدار در آن مکان، تعداد دفعات تکرار آن کلمه است که در یک پاراگراف مشخص ظاهر می‌شود. در این مورد، با نگاه کردن به مجموعه آموزشی خود می‌توانیم اندازه واژگان را ببینیم:

ما از همین مقدار در اولین لایه شبکه عصبی خود برای تعیین اندازه لایه ورودی استفاده می‌کنیم: self.fc1 = nn.Linear(X\_train.shape[1], 128). این عملکرد به شبکه ما اجازه می‌دهد تا آنچه را که به آن تعبیه‌های کلمه می‌گویند بیاموزد، یعنی بازنمایی برداری از کلماتی که از یک یادگیری تحت نظارت مانند کاری که در این بخش انجام دادیم آمده است. این کار به ما امکان می‌دهد جاسازی‌های کلمات با اندازه ۱۲۸ را یاد بگیریم، اگرچه این جاسازی‌ها در درجه اول برای این کار خاص و واژگان مفید خواهند بود.

**22.3 تنظیم دقیق یک مدل از پیش آموزش‌دیده برای طبقه بندی تصویر**

**مسئله**

شما می‌خواهید یک مدل طبقه بندی تصویر را با استفاده از آموخته‌های یک مدل از پیش آموزش‌دیده آموزش دهید.

**راه‌حل**

از کتابخانه transformers با torchvision برای تنظیم دقیق یک مدل از پیش آموزش‌دیده روی داده‌های خود استفاده کنید:



**بحث**

در حوزه داده‌های بدون ساختار مانند متن و تصاویر، به‌جای شروع از ابتدا، به‌ویژه در مواردی که به داده‌های برچسب‌گذاری‌شده زیادی دسترسی نداریم، بسیار رایج است که از مدل‌های از پیش آموزش‌دیده‌شده روی مجموعه داده‌های بزرگ شروع کنیم. با استفاده از جاسازی‌ها و اطلاعات دیگر از مدل بزرگ‌تر، می‌توانیم مدل خود را برای یک کار جدید بدون نیاز به اطلاعات برچسب‌گذاری شده زیادی تنظیم کنیم. علاوه بر این، مدل از پیش آموزش‌دیده ممکن است اطلاعاتی داشته باشد که اصلاً در مجموعه داده آموزشی ما ثبت نشده باشد، که منجر به بهبود عملکرد کلی می‌شود. این فرآیند به یادگیری انتقالی معروف است.

در این مثال، وزن‌ها را از مدل ViT (Vision Transformer) گوگل بارگیری می‌کنیم. سپس، از کتابخانه transformers برای تنظیم دقیق آن برای یک کار طبقه‌بندی در مجموعه داده‌های مد MNIST، مجموعه داده‌ای ساده از اقلام لباس، استفاده می‌کنیم. این رویکرد را می‌توان برای افزایش عملکرد در هر مجموعه داده بینایی کامپیوتری اعمال کرد، و کتابخانه transformers یک رابط سطح بالایی را ارائه می‌دهد که می‌توانیم از آن برای تنظیم دقیق مدل خود از مدل‌های بزرگتر و از پیش آموزش‌دیده‌شده بدون نوشتن مقدار زیادی کد استفاده کنیم.

**همچنین ببینید:**

* [وب‌سایت و مستندات Hugging Face](https://huggingface.co/)

**22.4 تنظیم دقیق یک مدل از پیش آموزش‌دیده‌شده برای طبقه بندی متن**

**مسئله**

شما می‌خواهید یک مدل طبقه بندی متن را با استفاده از آموخته‌های یک مدل از پیش آموزش‌دیده‌شده آموزش دهید.

**راه‌حل**

از کتابخانه transformers استفاده کنید:



**بحث**

درست مانند استفاده از مدل‌های تصویری از پیش آموزش‌دیده شده، مدل‌های زبانی از قبل آموزش‌دیده شده، زمینه‌ی زیادی در مورد زبان دارند، زیرا معمولاً در طیف گسترده‌ای از منابع اینترنتیِ باز آموزش داده می‌شوند. هنگامی که از یک پایه مدل از پیش آموزش‌دیده شده شروع می‌کنیم، کاری که معمولا انجام می‌دهیم این است که لایه طبقه بندی شبکه موجود را با یکی از لایه‌های خود عوض می‌کنیم. این مورد به ما امکان می‌دهد تا وزن‌های شبکه را که قبلاً آموخته‌ایم، متناسب با وظایف خاص خود تغییر دهیم.

در این مثال، ما در حال تنظیم دقیق مدل DistilBERT هستیم تا تشخیص دهیم که آیا نقدهای فیلم IMDB مثبت (1) یا منفی (0) بوده است. مدل DistilBERT از قبل آموزش داده شده است که مجموعه بزرگی از کلمات و زمینه را در هر یک، علاوه بر وزن شبکه عصبی که از آموزشی قبلی آموخته شده است، ارائه می‌دهد. یادگیری انتقالی به ما این امکان را می‌دهد که از تمام کارهای اولیه انجام شده در آموزش مدل DistilBERT استفاده کنیم و آن را برای مورد استفاده خود، که در این مثال طبقه بندی نقدهای فیلم است، تغییر دهیم.

**همچنین ببینید:**

* [طبقه بندی متن در کتابخانه transformers](https://huggingface.co/docs/transformers/tasks/sequence_classification)

1. - Convolutional Neural Network (CNN) [↑](#footnote-ref-1)