



سالار نوری ۴۰۲۲۰۳۶۹۱

محسن امیری امجد ۴۰۳۲۰۶۴۰۲

پروژه تشویقی

بخش اول: پیش پردازش داده‌ها

استخراج و ذخیره سازی اطلاعات اولیه

در این مرحله، ابتدا داده‌ها از فایل‌های DICOM استخراج شدند و اطلاعات اولیه‌ی مربوط به هر مجموعه‌ی تصویری در قالب فایل `dicom_info.csv` ذخیره گردید. این فایل شامل چهار ویژگی کلیدی برای هر تصویر بود، از جمله ابعاد تصویر، نوع اصلاحات اعمال شده، اطلاعات تاریخچه‌ای و سایر ویژگی‌های ضروری.

به عنوان گام مقدماتی، بررسی فایل `metadata.csv` انجام شد. این فایل توصیف کلی از مجموعه داده و ساختار فایل‌ها را ارائه می‌داد. هدف از ایجاد این فایل‌های کمکی `metadata.csv` و `dicom_info.csv` تسهیل دسترسی به اطلاعات هر تصویر در مراحل بعدی، بدون نیاز به جستجوی مجدد در فایل‌های حجیم DICOM بود.

تبدیل فرمت و تنظیم مختصات

پس از استخراج اطلاعات، فرآیند تبدیل فرمت از DICOM به NIfTI با استفاده از کتابخانه‌های تخصصی پردازش پزشکی در پایتون انجام شد. این تبدیل، امکان نمایش تصاویر به صورت سه بعدی را فراهم کرد و استانداردسازی داده‌ها را برای اعمال الگوریتم‌های مختلف ثبت (`registration`) و اصلاحات مقدماتی ممکن ساخت.

در مرحله‌ی بعد، تمامی فایل‌های NIfTI بر اساس الگوی LPS/RAI سازمان‌دهی شدند. این اقدام به منظور همخوانی با طیف وسیع‌تری از ابزارهای پردازش تصویر پزشکی انجام شد، زیرا بسیاری از این ابزارها از این الگوی محور مختصات پشتیبانی می‌کنند.

ثبت تصاویر بر روی اطلس استاندارد

یکی از گام‌های اساسی در این فرآیند، ثبت تصاویر (image registration) بر روی اطلس استاندارد SRI۲۴ بود. برای انجام این کار، سه روش مختلف شامل ants, synthmorph و simpleitk مورد آزمایش قرار گرفتند.

اجرای این روش‌ها در محیط‌های رایانش ابری (مانند Google Colab) انجام شد و معیارهایی نظیر سرعت پردازش، میزان همخوانی نهایی با مختصات اطلس و کیفیت ظاهری تصاویر خروجی بررسی شدند. نتایج نشان داد:

- روش ants از نظر دقت ثبت، بهترین عملکرد را ارائه داد.
 - روش synthmorph سرعت بالاتری داشت اما کیفیت خروجی آن اندکی پایین‌تر از ants بود.
- با توجه به محدود بودن تعداد داده‌ها (۴۷۸ تصویر (و قابل قبول بودن زمان پردازش) حدود یک ساعت و نیم)، در نهایت روش ants انتخاب شد تا تصاویر ثبت‌شده نهایی برای مراحل بعدی تحلیل و مدل‌سازی مورد استفاده قرار گیرند.

حذف جمجمه و پردازش نهایی تصاویر

پس از اتمام فرآیند ثبت، مرحله‌ی حذف جمجمه (skull stripping) با استفاده از ابزار BET انجام گرفت. این فرآیند جهت حذف بخش‌های غیرضروری تصویر که خارج از محدوده‌ی اصلی مغز قرار داشتند، ضروری بود.

مقایسه‌ی BET با روش‌های ساده‌تر، مانند برخی متدهای موجود در کتابخانه‌ی simpleitk، نشان داد که BET عملکرد بهتری در تشخیص و تکه‌برداری تصاویر ارائه می‌دهد. پس از حذف بافت‌های بیرونی، اسلایس‌هایی که کاملاً سیاه بودند (فاقد اطلاعات مغزی) حذف شدند تا از ایجاد تداخل بصری یا محاسباتی جلوگیری شود.

از آنجا که تعداد اسلایس‌های باقی‌مانده در تصاویر مختلف متفاوت بود، برای همسان‌سازی داده‌ها از روش‌های میان‌یابی (interpolation) استفاده شد. این اقدام باعث شد که شبکه‌های عصبی یا سایر روش‌های یادگیری ماشین در مراحل بعدی، ورودی‌های یکنواخت و استاندارد دریافت کنند.

ادغام داده‌های پردازش‌شده در یک مجموعه‌ی واحد

در پایان، اطلاعات مربوط به تمامی فایل‌های پردازش‌شده، همراه با لیبل‌های مرتبط، در یک دیتاست واحد و منظم ادغام شد. این ساختار منسجم، امکان استفاده‌ی آسان در مراحل بعدی مدل‌سازی را فراهم کرد.

این قالب استاندارد موجب شد که در مراحل بعدی، مانند طراحی مدل‌های عصبی برای کلاسیفیکیشن یا سایر وظایف تحلیلی، نیازی به انجام مجدد پیش‌پردازش نباشد. به این ترتیب، با صرف زمان و انرژی کمتر، امکان تمرکز بر فرآیند آموزش مدل‌ها فراهم شد.

جمع‌بندی فرآیند پیش‌پردازش داده‌ها

مراحل اصلی پیش‌پردازش شامل موارد زیر بود:

۱. استخراج داده‌ها از فایل‌های DICOM و ذخیره‌ی اطلاعات اولیه در dicom_info.csv
 ۲. تبدیل فرمت از DICOM به NIfTI
 ۳. استانداردسازی مختصات بر اساس LPS/RAI
 ۴. ثبت تصاویر به اطلس استاندارد با روش ants (پس از مقایسه با simpleitk و synthmorph)
 ۵. حذف جمجمه (skull stripping) با BET
 ۶. حذف اسلایس‌های کاملاً سیاه برای جلوگیری از تداخل محاسباتی
 ۷. همسان‌سازی اسلایس‌ها با استفاده از میان‌یابی (interpolation)
 ۸. ادغام داده‌های پردازش‌شده در یک مجموعه‌ی واحد برای مراحل بعدی مدل‌سازی
- با اجرای این فرآیندها، مجموعه داده‌ی نهایی به‌شکلی شفاف و استاندارد آماده شد تا در فازهای بعدی، از جمله مدل‌سازی شبکه‌های عصبی و تحلیل‌های تخصصی تصویربرداری مغزی، مورد استفاده قرار گیرد.

بخش دوم: آماده‌سازی داده و آموزش مدل

شرح مجموعه داده

مجموعه داده شامل ۱۵۹ نمونه است که هر نمونه دارای دو تصویر (T1C و T2W) است که در مجموع ۳۱۸ تصویر را تشکیل می‌دهد. تصاویر اصلی در تعداد برش‌ها متفاوت هستند و برخی از آنها حاوی برش‌های کاملاً سیاهی هستند که باید حذف شوند. آماده‌سازی شامل هم‌اندازه‌سازی، حذف برش‌های سیاه و تغییر نام تصاویر برای آموزش بهتر مدل است.

آماده‌سازی داده‌ها

این بخش جزئیات مراحل آماده‌سازی اعمال شده روی تصاویر MRI قبل از استفاده از آنها برای یادگیری مدل را شرح می‌دهد. این آماده‌سازی‌ها تضمین می‌کند که تصاویر تمیز، یک‌نواخت و یک‌شکل شده‌اند که از گنگ‌شدن پروسه جلوگیری می‌کند.

حذف برش‌های سیاه

تصاویر MRI اغلب حاوی برش‌هایی هستند که کاملاً سیاه هستند و اطلاعات مفیدی برای طبقه‌بندی ارائه نمی‌دهند. این برش‌ها برای کاهش نویز و اطمینان از اینکه مدل فقط روی داده‌های معنادار آموزش می‌بیند، حذف شدند.

هم‌اندازه‌سازی برش‌های تصاویر

تعداد برش‌ها در هر تصویر بین موضوعات مختلف متفاوت بود. تصاویری با فقط یک برش، و تصاویری بیش از ۱۲۰ برش در داده‌ها وجود داشتند. برای اطمینان از یکنواختی، همه تصاویر به ۵۶ برش درون‌یابی شدند که حدوداً میانگین و میانه مجموعه داده بود. همچنین، برای جلوگیری از تولید مصنوعات (artifact)، اگر تصویری کمتر از ۵۶ برش داشت، برش‌های آن، قبل از هم‌اندازه‌سازی با استفاده از درون‌یابی اسپلایت (Spline Interpolation)، تکثیر شدند. در این عملیات، اگر تعداد برش‌های تصویر کمتر از حد نصاب بود، تعداد برش‌ها (با تکرار پیاپی آن‌ها) در ضریب متناسب ضرب می‌شد. در تنظیمات درون‌یابی استفاده شده، از درون‌یابی مکعبی (یا در نمای ۳) استفاده شد، زیرا کمترین آسیب را به ساختار تصاویر MRI وارد می‌کرد.

محاسبه نرخ‌های نرمال‌سازی

میانگین و انحراف معیار داده‌به‌داده (per sample) مقادیر واکسل‌های تصاویر محاسبه شد تا در مرحله آماده‌سازی دیتاست، از آن‌ها برای نرمال‌سازی داده‌ها استفاده شود. نرمال‌سازی به تثبیت آموزش و اطمینان از همگرایی سریع‌تر مدل کمک می‌کند.

سازماندهی داده‌ها

پس از پیش‌پردازش، مجموعه داده بر اساس حالت تصویربرداری به دو گروه تقسیم شد: تصاویر T1C و تصاویر T2W. در این مرحله، هر تصویر را تغییر نام داده و در قالب ساختاریافته‌ای در یک پوشه جدید برای تسهیل بارگذاری و آموزش آسان‌تر ذخیره کردیم.

آموزش مدل

این بخش نحوه آماده‌سازی دیتاست آموزش، معماری مدل، استراتژی‌های افزایش داده، انتخاب تابع زیان و فرآیند آموزش را شرح می‌دهد.

ایجاد دیکشنری داده

لیستی از همه ۱۵۹ نمونه ایجاد شد. به ازای هر نمونه، تصاویر T1 و T2 آن، نام نمونه، و برجسب مربوط به نمونه، در یک دیکشنری زیر ذخیره شد. این دیکشنری بارگذاری، پردازش و استفاده از مجموعه داده را در طول آموزش آسان‌تر کرد.

تعریف دیتاست پایتورچ

یک کلاس فرزند کلاس دیتاست پایتورچ، به جهت انجام عملیات زیر پیاده‌سازی شد:

۱. دریافت محل ذخیره‌سازی تصاویر و بارگذاری آن‌ها از دیکشنری دیتا.
۲. الحاق تصاویر T1c و T2w در امتداد بعد کانال و تولید تصویری ۳ بعدی که دارای ۱۱۲ کانال می‌باشد.
۳. جابجایی ابعاد به گونه‌ای که کانال‌ها بعد اول شود (فرمت $C \times H \times W \times D$). این تنظیمات تضمین می‌کند که شکل ورودی با شبکه‌های عصبی کانولوشنال سازگار است.

تقسیم داده‌ها

مجموعه داده به دو دسته آموزش و اعتبارسنجی با ضرائب ۸۰ و ۲۰ درصد تقسیم شد.

از آنجایی که مجموعه داده دارای عدم تعادل بود و یک کلاس تقریباً نصف اندازه کلاس دیگر بود، برای متعادل کردن کلاس‌ها، کلاس اقلیت برای مطابقت با کلاس اکثریت تکثیر شد. این امر از بایاس مدل به سمت کلاس اکثریت جلوگیری می‌کند و تعمیم را بهبود می‌بخشد.

افزایش داده‌ها

به دلیل کم بودن تعداد تصاویر دیتاست، تکنیک‌های افزایش زیر استفاده شد:

- چرخش تصادفی افقی و عمودی (RandomHorizontalFlip و RandomVerticalFlip)
- چرخش تصادفی (RandomRotation)
- تغییرات خطی تصادفی (RandomAffine)
- تغییر زاویه دید، برای تقلید از تغییرات طبیعی در MRI (RandomPerspective)

نرمال‌سازی داده‌ها

تصاویر با استفاده از میانگین و انحراف معیار محاسبه‌شده در مرحله آماده‌سازی، نرمال شدند. این امر تضمین می‌کند که توزیع داده‌ها در طول آموزش و اعتبارسنجی ثابت می‌ماند.

ایجاد DataLoader آموزش و اعتبارسنجی

اندازه هر مینی‌بچ ۸ انتخاب شد، تا تعادل بین پایداری نزول گرادیان (دسته‌های کوچکتر می‌توانند باعث گرادیان‌های نویزی و ناپایدار شوند) و تعمیم‌یابی خوب (دسته‌های بزرگتر می‌توانند منجر به همگرایی در نقاط maxima تیز و غیرتعمیم‌پذیر شوند) برقرار شود.

انتخاب مدل

معماری مدل انتخاب شده EfficientNet-b0 بود. این مدل با استفاده از پیاده‌سازی EfficientNet از MONAI است که روی ImageNet آموزش داده شده بود. استفاده از یک مدل از پیش آموزش دیده به یادگیری انتقالی کمک می‌کند و به مدل اجازه می‌دهد از ویژگی‌های آموخته شده از تصاویر عمومی استفاده کند.

مدل‌های سه‌بعدی دیگری مانند DenseNet121، EfficientNet-b1، EfficientNet-b2، ResNet18 و ResNet10 نیز بررسی شدند. همچنین در مراحل ارزیابی، چهاربعدی کردن داده‌ها (با ۲ کانال که یکی T1 و دیگری T2 باشد) نیز روی مدل‌های متعددی بررسی شد. در نهایت مدل EfficientNet-b0 به دلیل سرعت بالا (مشکل پردازشی زیاد بود)، حجم پایین (رم گرافیک کم بود) و عملکرد بهتر از دیگر روش‌ها انتخاب شد. همه‌ی مدل‌های استفاده شده به دلیل سهولت، با استفاده از MONAI پیاده‌سازی شده بودند.

تابع زیان و بهینه‌ساز

برای پیاده‌سازی این عملیات از یک تابع زیان ترکیبی استفاده شد. تابع زیان نهایی یک ترکیب وزنی ۱:۱ از زیان BCE و دایس بود.

- زیان آنتروپی متقابل دودویی (BCE): تابع زیان پایه، برای عملیات کلاس‌بندی دودویی.
- زیان دایس: این تابع زیان در آموزش‌هایی که دیتا نسبت به سختی عملیات محدود باشد، به تعمیم‌پذیری مدل خروجی کمک می‌کند.

از بهینه‌ساز AdamW برای بهینه‌سازی استفاده شد. همچنین با استفاده از زمان‌بند نرخ یادگیری CosineAnnealing، کاهش تدریجی نرخ یادگیری در طول آموزش پیاده‌سازی شد. نرخ یادگیری اولیه، و نرخ یادگیری نهایی طی چند مرحله آموزش آزمایشی انتخاب شد.

آموزش مدل

حلقه آموزش با استفاده از PyTorch با اجزای کلیدی زیر پیاده‌سازی شد:

- **گذر رو به جلو:** تصاویر به مدل EfficientNet داده شدند.
- **محاسبه زیان:** با استفاده از تابع زیان ترکیبی BCE + دایس.
- **گذر رو به عقب:** گرادیان‌ها محاسبه شدند و برای به‌روزرسانی وزن‌های مدل استفاده شدند.
- **مرحله اعتبارسنجی:** پس از هر دوره، زیان اعتبارسنجی برای نظارت بر بیش‌برازش محاسبه شده و مقادیری مانند ماتریس confusion و دقت در خروجی چاپ شدند.
- **ذخیره مدل:** پس از هر اپاک، در صورت بهتر شدن عملکرد مدل، وزن‌های مدل در سیستم ذخیره شدند تا در آینده از آن‌ها استفاده شود.

گزارش آموزش مدل‌ها

آموزش مدل‌ها بسته به اندازه k -fold اضافه‌سازی دیتا که با مقادیر ۱ تا ۳ مورد بررسی قرار گرفت، و همچنین با توجه نوع ساختار تصویر مورد استفاده (سه یا چهاربعدی)، و نیز مدل مورد استفاده، حدوداً بین ۱۰ تا ۷۵۰ ثانیه به ازای هر اپاک زمان نیاز داشت. مدل نهایی انتخاب شده، $k=3$ اضافه‌سازی داده، با تصاویر سه‌بعدی، و مدل EfficientNet-b0 داشت که در هر اپاک به ۹۰ ثانیه زمان نیاز داشت. کل پروسه آموزش که ۱۰۰ اپاک بود حدود ۲.۵ ساعت زمان برد. در نهایت مدل ارائه شده، روی دیتای اعتبارسنجی (Validation Set) به دقت ۷۵٪ رسید و داده‌های تست ارائه شده از دامنه‌ی هدف را با دقت ۱۰۰٪ برچسب‌گذاری کرد. آموزش مدل‌ها همگی روی سیستم محلی (Local)، روی کارت گرافیک 4060 انجام شد.