



بسم الله الرحمن الرحيم

کارآموزی AIMedic

تاپستان ۱۴۰۰

تمرین سری چهارم

مجتبی نافذ

شبکه های عصبی

در ابتدا نوت بوک حدود ۱۰ مگ بود و در ایمیل قابل ارسال نبود و کد در لینک [google colab](https://colab.research.google.com/drive/1nUVojdQnY9xBWgYn5FvK2erfD-mCHELt?usp=sharing) موجود است

لینک برای تغییر کد: <https://colab.research.google.com/drive/1nUVojdQnY9xBWgYn5FvK2erfD-mCHELt?usp=sharing>

لینک برای مشاهده: <https://colab.research.google.com/drive/1nUVojdQnY9xBWgYn5FvK2erfD-mCHELt?usp=sharing>

لینک پوشه ی لاگی ها و مدل های ذخیره شده:

<https://drive.google.com/drive/folders/1-5RvxsysXrGwNiBkqgPAvFS6uq5AJedN?usp=sharing>

برای باز شدن tensorboard در نوت بوک باید کمی منتظر بمانید تا لود شود. (تست کردم قابل رویت از سمت شما میباشد)

قسمت اول: توضیحات دیتاست

به لحاظ پزشکی:

در صنعت پزشکی رنگ آمیزی با H&E یکی از متداول ترین تکنیک های رنگ آمیزی در بافت شناسی است. در بافتی که با هماتوکسیلین Haematoxylin و ائوزین Eosin آغشته شده باشد، سیتوپلاسم به رنگ صورتی نارنجی و هسته به صورت تیره رنگ، یا به رنگ آبی یا بنفش نشان داده می شود. همچنین با رنگ آمیزی ائوزین، سلول های قرمز خون به رنگ شدیداً قرمز نشان داده می شوند. ائوزین یک رنگ اسیدی است و بخش اصلی یا اساسی هر سلول، یعنی سیتوپلاسم آن را نشان می دهد. گرچه هماتوکسیلین یک رنگ اولیه یا پایه ای است و در قسمت های اسیدی سلول مانند هسته، جاهایی را که اسیدهای نوکلئیک (DNA و RNA) در آنها متمرکز شده است را نشان می دهد.

به لحاظ دیتا ساینس:

دیتاست به شدت کوچک و داده ها خیلی کم است و train کردن شبکه تقریباً ممکن نیست.

یا اگر بخواهیم واقعاً دقت بگیریم باید مدل بیش از ۶، ۷ ساعت train شود که برای بنده مقدور نبود

قسمت دوم: توضیحات Augmentation

Augmentation: می توانیم از rotation , vertical flip , rescale , zoom , horizontal flip استفاده کرد با توجه به نوع مساله نباید آگمنتیشن طوری باشد که آسیبی به رنگ های کنار هم برسد چون به طور مثال پیکسل های مربوط به یک سلول باید کنار هم باشند و اگر از width_shift و hight_shift استفاده کنیم امکان دارد مقداری از سلول بریده شود

و شبکه چیز اشتباهی را آموزش ببیند

قسمت سوم : روند کلی انجام پروژه

همه ی تحلیل ها روی همه ی متریک ها را از روی **tensorboard** میتوانید مشاهده کنید.

۱. بنده در ابتدای کار یک **Unet** ساده آموزش دادم که تغییر دقت آن نیاز به زمان خیلی زیاد داشت و دقت خوبی نگرفتم

۲. سپس از مدل **VGG19** که روی **resnet** آموزش دیده بود به عنوان بخش **encoder** در معماری **unet** استفاده کردم که تقریباً مثل قبلی عمل کردم و تغییر دقت آن خیلی محسوس نبود ولی بهتر از آن بود و سعی کردم آن را هم **fine Tune** کنم اما نتیجه بهتر نشد

۳. سپس از یک لایبری که به روی دیتاست بزرگ تری خود **unet** را آموزش داده بود استفاده کردم که آن هم در مینیمم محلی گیر میکرد که زمان زیادی نیاز داشت تا بهبود حاصل شود اما قطعاً بهتر میشد اگر زمان میدادیم

۴. سپس همان لایبری را **fine tune** کردم که مینیمم محلی قبلی که در خطای حد ۰.۶۹ بود را رد کرد و بهبود یافت و بهترین نتیجه را داشت اما به زمان نیاز داشت که بهبود بیشتری حاصل شود

قسمت چهارم : بررسی نتیجه ی آموزش شبکه ها

۱- دیتاست به شدت کوچک است.

۲- بهترین دقت در آخرین اجرا بود زمانی که از **pretrain library** استفاده کردیم این لایبری خود **Unet** را روی دیتاست بزرگی ران کرده و ما حال فقط آخرین لایه های آن را **Fine Tune** کردیم که در این حالت معیار دایس در داده های آموزشی به ۷۶ و در داده های تست به ۷۰ درصد رسید که نسبتاً بد نبود اما برای دقت بالا و کاربردی بیش از این ها نیاز به آموزش مدل داریم.

۳- برای افزایش دقت نیاز بود که چندین ساعت **train** با **GPU** شود به همین خیلی آموزش را طولانی نکردم.

