

## آرنسفورمها

### سوال ۱

در این تمرین، هدف آشنایی عملی با معماری آرنسفورم و بررسی حساسیت عملکرد آن نسبت به پارامترهای مختلف طراحی و آموزش است. تمرکز این تمرین بر روی استفاده از آرنسفورمها برای وظیفه استخراج موجودیت‌های نام‌دار<sup>۱</sup> و تحلیل اثر اجزای مختلف مدل و تنظیمات آن بر عملکرد نهایی است.

مجموعه داده:

فایل زیپ شامل مجموعه داده مورد استفاده به تعریف تمرین پیوست شده است. این فایل شامل داده‌های آموزشی و آزمون است. جملات مجموعه داده را می‌توانید در فایل‌هایی که با نام tokens مشخص شده‌اند و برچسب هر کلمه را می‌توانید در فایل‌هایی که با نام labels مشخص شده‌اند مشاهده نمایید. از مجموعه داده peyma برای آزمون و از مجموعه داده arman برای آموزش بهره بگیرید.

۱ - یک مدل پایه از نوع BERT برای وظیفه NER بارگذاری کرده و آموزش دهید. از کلاس‌های آماده‌ی کتابخانه transformers (مانند AutoModelForTokenClassification) استفاده نمایید. نتایج اولیه را با معیارهایی مانند F1-score و Accuracy گزارش کنید.

۲ - حساسیت شبکه نسبت به پارامترهای مختلف را با انجام آزمایش‌های زیر بررسی و تحلیل کنید:

– تعداد لایه‌های آرنسفورم (مثلاً ۴، ۶، ۱۲)

– تعداد سرهای توجه<sup>۲</sup>

– اندازه مخفی<sup>۳</sup> در لایه‌های توجه و کاملاً متصل<sup>۴</sup>

– اندازه توکن ورودی<sup>۵</sup>

<sup>۱</sup> Named Entity Recognition - NER

<sup>۲</sup> Attention Heads

<sup>۳</sup> Hidden Size

<sup>۴</sup> Fully Connected

<sup>۵</sup> max sequence length

برای هر مورد، تنها یک پارامتر را تغییر داده (حداقل با ۳ مقدار مختلف) و بقیه پارامترها را ثابت نگه دارید. نتایج را در جدول و نمودار مناسب گزارش و تحلیل کنید.

۳ - یک مدل ترنسفورمر ساده (مثلاً با دو لایه کدگذار<sup>۶</sup>) را از ابتدا با PyTorch پیاده‌سازی کنید. از این مدل برای وظیفه استخراج موجودیت‌های نام‌دار روی زیرمجموعه‌ای از داده‌ها استفاده نمایید. نتایج حاصل را با مدل آماده‌ی BERT مقایسه و تحلیل کنید.

۴ - دو ساختار زیر را بر اساس عملکرد با هم مقایسه کنید.

– استفاده از خروجی آخرین لایه برای پیش‌بینی برچسب هر توکن

– میانگین گرفتن از خروجی چند لایه آخر برای هر توکن

توضیح دهید که کدام روش بهتر عمل می‌کند و چرا؟

۵ - میزان حساسیت مدل به ترتیب توکن‌ها را بررسی کنید:

– ورودی‌هایی را با ترتیب توکن‌های جابه‌جا شده (شبیه word dropout یا shuffle) ایجاد کرده و ارزیابی نمایید.

– تفاوت عملکرد را گزارش کرده و تحلیل نمایید که آیا مدل واقعاً از ساختار ترتیبی استفاده کرده است یا خیر.

## سوال ۲

این سوال، امتیازی است.

در این بخش قصد داریم مدل BERT را برای زبان فارسی تنظیم دقیق<sup>۷</sup> کرده و بررسی کنیم که آیا این آموزش محدود می‌تواند دقت نهایی در استخراج موجودیت‌های نام‌دار را بهبود دهد یا خیر.

۱ - یک زیرمجموعه از داده‌های فارسی (مثلاً ویکی‌پدیای فارسی یا داده‌های مشابه) آماده کرده و از آن برای آموزش بدون نظارت به روش Masked Language Modeling (MLM) استفاده کنید.

۲ - مدل را برای چند تکرار با بکارگیری مجموعه‌داده فارسی، تنظیم دقیق کنید. از Trainer API برای آموزش استفاده نمایید و نمودار خطای آموزش را رسم کنید.

Encoder<sup>۶</sup>  
Fine tune<sup>۷</sup>

۳ - سپس همین مدل را برای استخراج موجودیت‌های نام‌دار آموزش دهید و با مدل بدون تنظیم دقیق مقایسه نمایید. آیا تنظیم دقیق توانسته است تفاوتی ایجاد کند؟

در انجام تمرینات به نکات زیر توجه فرمایید.

۱ - پیاده‌سازی با زبان Python و چارچوب کاری PyTorch انجام شود. استفاده از torchvision برای بارگذاری داده‌ها مجاز است.

۲ - گزارش نهایی باید شامل تحلیل دقیق عملکرد، نمودارهای Loss، نتایج عددی FID/IS، تصاویر تولیدشده و مقایسه با نمونه‌های واقعی باشد.

۳ - در صورت استفاده از مدل‌های از پیش آماده یا معماری‌های مقاله‌ای، منابع را به‌طور کامل ذکر کنید.

۴ - ارسال فایل فشرده شامل کد و گزارش با نام StudentID\_HW07.zip در مؤعد مقرر الزامی است. در غیر این صورت نمره تمرین لحاظ نخواهد شد.

۵ - کپی‌برداری از دیگر دانشجویان یا استفاده مستقیم از کدهای آماده بدون درک کامل، موجب حذف کل نمره تمرین خواهد شد.