

گزارش پروژه تصحیح املائی و گرامری متون فارسی (کاتب)

دوره کارآموزی عصرگوش پرداز

سپهر نوعی کردکندی

۲۶ سپتامبر ۲۰۲۳

۱ مقدمه

امروزه یکی از کاربردهای محصولات هوش مصنوعی در زمینه پردازش زبان طبیعی، تصحیح املائی و دستور زبانی متون در زبان‌های مختلف می‌باشد. در واقع هدف در این مسائل، پردازش متون ورودی به مدل یادگیری ماشین و اصلاح املائی و دستور زبانی آن با توجه به مفهوم موجود در متن داده شده می‌باشد. محصولات مشابه در زبان‌های دیگر مانند زبان انگلیسی موجود می‌باشد که از دقت خوبی نیز برخوردار می‌باشند، اما در زبان فارسی، مدل‌های آموزش دیده در این زمینه عملکرد مطلوبی ندارند. در این پروژه سعی شده است که با توجه به دادگان موجود برای تصحیح املائی و مدل‌های مطرح جهانی و ایرانی، مدلی برای این کاربرد توسعه داده شود.

۲ فعالیت‌های صورت گرفته در این پروژه

در این بخش، قصد داریم که تمامی فعالیت‌های صورت گرفته در این پروژه را تشریح کنیم. به طور کلی فعالیت‌های صورت گرفته شامل دو بخش جستجوی مقالات و مدل‌ها، و بخش پیاده‌سازی و اشکال‌زدایی می‌باشد.

۱.۲ جستجوی مقالات و مدل‌ها

اولین گام در این پروژه، پیدا کردن مقالات و دادگان موجود می‌باشد. در این مرحله، مقالات دهه اخیر در این زمینه بررسی شدند و بهترین آن‌ها با توجه به نیازهای محصول انتخاب شدند. معیارهای انتخاب مدل‌ها، در ابتدا عملکرد مناسب آن‌ها در دادگانی که به عنوان معیار برای مدل‌های text-to-text وجود دارند، در نظر گرفته شده است. سپس، از بین مدل‌ها، آن‌هایی انتخاب شدند که توانایی پردازش زبان فارسی را نیز دارند و یا حتی به طور خاص برای زبان فارسی توسعه داده شده باشند، و در مرحله آخر، مقالاتی برای استفاده از تکنیک‌های آن‌ها برای افزایش کارایی، سرعت و عملکرد مدل انتخاب شدند.

مدل‌هایی که برای نقطه پایه و شروع آموزش مدل انتخاب شدند عبارتند از: مدل mT5 و مدل ParsBERT. این مدل‌ها ویژگی‌های لازم برای آموزش برای کاربرد مدنظر را دارند که در ادامه این گزارش به تفکیک توضیح داده می‌شوند [۱-۲].

قبل از آن، در اینجا به بررسی تکنیک‌هایی که از مقالات دیگر به دست آورده شده است، شرح داده می‌شوند:

- در یکی از مقالات خوانده شده، یک روش ساده اما کارآمد برای ترکیب مدل‌های مختلف ارائه شده است. روش ترکیب به این صورت است که، هر مدل، چندین ویرایش برای اصلاح متن پیشنهاد می‌دهد، و مدل برای تصمیم‌گیری برای استفاده از این پیشنهاد یا نادیده گرفتن آن، آن را به صورت یک مسئله Binary Classification مدل می‌کند. یعنی برای هر پیشنهاد، در صورتی که طبق مدل ترکیب‌کننده، خروجی تابع Sigmoid بالاتر از آستانه مدنظر باشد، آن ویرایش در ویرایش نهایی مدل خواهد آمد، در غیر این صورت، نادیده گرفته می‌شود.

در واقع، در این روش، مدل تصحیح‌کننده به صورت یک Black Box در نظر گرفته می‌شوند و ما ترکیب مدل‌ها را فقط بر اساس ویرایش‌های پیشنهادشده توسط هر مدل انجام می‌دهیم. روند ترکیب به این صورت است که از پیش‌بینی هر مدل، ویرایش‌ها به فرمت end index, (start correction index, string استخراج می‌شوند. همچنین، هر ویرایش، یک نوع type دارد که توسط یک ابزار تشخیص نوع خطا به صورت خودکار صورت می‌گیرد. سپس نوع خطا و ویرایش‌ها به فرمت ذکر شده، به عنوان feature در مدل استفاده می‌شوند. دقت شود که هر ویرایش، می‌تواند در یکی از سه دسته درج (insertion)، حذف (deletion)، و یا جایگزینی (substitution) باشد. سپس، هر ویرایش پیشنهاد شده توسط مدل‌ها، به صورت مستقل و بدون اطلاع از ویرایش‌های دیگر و کلمات دیگر متن، بررسی می‌شود. در بررسی این ویرایش‌ها از یک مدل خطی تعمیم یافته برای تصمیم‌گیری برای نگهداری یا نادیده گرفتن ویرایش استفاده می‌شود.

به عنوان مثال، بردار ویژگی‌هایی که به ازای k پیش‌بینی از k مدل، به مدل ترکیب‌کننده داده می‌شود، در فضای برداری $k \times |T|$ می‌باشد که در آن $|T|$ اندازه انواع خطاهای ممکن می‌باشد و بردار T نمایش one-hot نوع خطا در آن ویرایش می‌باشد. بدین ترتیب، مدل ترکیب‌کننده یاد می‌گیرد که فقط بر اساس نوع خطا و مدل پیشنهاد دهنده آن، برای نگهداری یک ویرایش در اصلاح نهایی تصمیم بگیرد. همچنین، در این روش، مکانیزم‌هایی برای جلوگیری از وجود همزمان ویرایش‌های تکراری یا ناسازگار به کار گرفته می‌شود. به این صورت که اگر اصلاح‌های انتخاب شده، دارای مکان مشترک در جمله بودند (که با استفاده از اندیس شروع و پایان بررسی می‌شود)، ویرایشی که احتمال بالاتری برای استفاده شدن در ویرایش نهایی داشت، استفاده می‌شود.

در این پژوهش نشان داده شده است که با وجود سادگی روش ترکیب در این رویکرد، این روش عملکرد بهتری نسبت به روش‌های ترکیب پیچیده‌تر داشته است [۳].

- در پژوهش دیگری، به روشی غیر وابسته به زبان برای ساخت دادگان مصنوعی و همچنین دستورالعملی برای آموزش مدل‌هایی برای اصلاح املائی دستور زبانی اشاره شده است. با توجه به نیازمندی‌های ما در این پروژه، می‌توانیم از روش تولید دادگان مصنوعی آن استفاده کنیم. فرایندی که در این پژوهش برای ساخت دادگان مصنوعی استفاده شده است بسیار ساده می‌باشد، به این صورت که دادگان دارای خطا، از ایجاد خطا در دادگان سالم به دست می‌آیند. عملیات استفاده شده، شامل (۱) حذف بعضی از توکن‌ها (۲) جابجایی توکن‌ها (۳) حذف بعضی از کاراکترها (۴) درج کاراکترهای اضافی (۵ و ۶) تبدیل حروف کلمه به حروف کوچک و یا بزرگ می‌باشند. البته روش‌های ایجاد خطای پیچیده‌تری نیز می‌توان برای هر زبان به صورت جداگانه توسعه داد [۴].

- در یکی از پژوهش‌ها، روش دیگری برای آموزش مدل ارائه شده است که می‌تواند باعث افزایش سرعت پیش‌بینی مدل می‌شود. در این روش، به جای آموزش مدلی برای بازتولید کامل متن، مدل برای شناسایی خطاها و برچسب زدن به آن آموزش داده می‌شود. این برچسب شامل نوع خطا و ویرایش پیشنهادی آن می‌باشد. در انتها، این برچسب‌ها، توسط یک تابع تبدیل به ویرایش روی متن تبدیل می‌شوند. در این پژوهش، یک تابع تبدیل مخصوص توسعه داده شده است که شامل دسته‌های زیر می‌باشد:

- تبدیل‌های پایه: شامل <KEEP> برای نگهداری بدون تغییر توکن‌ها، <DELETE> برای حذف توکن فعلی، <APPEND-t1> برای اضافه کردن توکن t1 بعد از توکن فعلی، و <REPLACE> <t2> جابجایی توکن ۲ با توکن فعلی - تبدیل‌های گرامری: مانند <MERGE> برای ترکیب توکن فعلی و توکن بعدی به یک توکن، برعکس آن <SPLIT> برای جداسازی توکن‌ها، <NOUN> <NUMBER> و <VERB-FORM> برای تبدیل کلمات به حالت جمع (یا مفرد) و تبدیل زمان فعل و مواردی مانند آن.

پس از تعیین کلیات روش مورد استفاده در این مقاله، به بررسی دقیق‌تر گام‌های این روش می‌پردازیم. دقت شود که برای آموزش به این روش، نیاز داریم که دادگان ما در قسمت ورودی و هدف، با هم نسبت داده شده باشند، و برچسب تبدیل بین آن‌ها نیز، تعیین شود. این آماده‌سازی دادگان برای آموزش در ۳ مرحله صورت می‌گیرد.

(۱) در ابتدا، هر توکن در جمله ورودی، به دنباله‌ای از توکن‌های مربوط آن در جمله هدف نسبت داده شود. برای این کار از کمینه کردن نوعی از فاصله Levenshtein استفاده می‌شود تا بهترین توکن‌هایی که می‌توان به یکدیگر نسبت داد، پیدا شوند.

(۲) برای هر جفت توکن نسبت داده شده به یکدیگر، تبدیل در سطح توکن بین آن‌ها پیدا شود.

(۳) برای هر توکن، فقط یک برچسب تبدیل نسبت داده شود، و این عمل به صورت تکرار شونده تکرار می‌شود و این عمل در ادامه به صورت تکرار شونده، ادامه می‌یابد و پس از چند مرحله، جمله نهایی مدنظر به دست می‌آید. در این پژوهش نشان داده شده است که این روش می‌تواند جایگزین مناسبی برای بازتولید متن باشد، به صورتی که در عین حال که دقت بالایی دارد، ولی تا ۱۰ برابر سرعت پیش‌بینی بیشتری نسبت به مدل‌های Transformer-based دارد [۵].

۲.۲ مدل mT5

یکی از مدل‌هایی که در این پروژه به عنوان مدل پایه در نظر گرفته شد، مدل mT5 می‌باشد. این مدل که توسط شرکت گوگل ارائه شده است، یک مدل چندزبانه می‌باشد که بر روی ۱۰۱ زبان مختلف آموزش دیده است. ساختار این مدل بر پایه مدل transformer می‌باشد و از لایه‌های attention بهره می‌گیرد. این مدل در سبک‌های متفاوت با تعداد لایه‌ها و پارامترهای متفاوت وجود دارد. این مدل برای یادگیری زبان‌ها با روش Fill-Mask پیش‌آموزش دیده است [۱]. در شکل ۱، یک نمونه از این روش مشاهده می‌شود. مدل mT5 یک مدل با ساختار Encoder - Decoder می‌باشد و از یک واژه‌نامه مشترک برای تمام زبان‌ها

store gallon
↑ ↑
the man went to the [MASK] to buy a [MASK] of milk

شکل ۱: روش Fill-Mask در زبان انگلیسی

استفاده می‌کند که این عمل، محاسبات و تبدیل به توکن کردن متون را بین زبان‌های مختلف آسان‌تر می‌کند. مدل mT5، مانند مدل T5، تمام کاربردهای پردازش زبان طبیعی را به صورت مسائل text-to-text مدل‌سازی می‌کند که باعث تسهیل روند آموزش بین همه کاربردها می‌شود. در روند پیش‌آموزش این مدل، یکی از چالش‌های اصلی، نحوه نمونه‌گیری دادگان از زبان‌های مختلف می‌باشد و دلیل اهمیت این موضوع این است که در صورتی که از زبان‌های با منابع (دادگان) کم، نمونه‌گیری به صورت زیاد انجام شود، ممکن است که مدل overfit کند، و در

صورتی که نمونه‌گیری از زبان‌های با حجم دادگان بالا (مانند انگلیسی) و آموزش روی آن، به اندازه کافی انجام نشود، مدل **underfit** خواهد کرد. به همین دلیل، روشی که برای نمونه‌گیری برای این مدل استفاده شده است، بر مبنای نسبت دادگان زبان‌ها می‌باشد، به صورتی که هرچه یک زبان دادگان بیشتری داشته باشد، احتمال نمونه‌گیری از آن بیشتر خواهد بود. فرمول استفاده شده در این فرایند عبارت است از:

$$p(L) \propto |L|^\alpha$$

که در آن، $p(L)$ احتمال نمونه‌گیری از یک زبان، $|L|$ تعداد رکوردهای موجود از آن زبان، و پارامتر α نیز به ما توانایی کنترل میزان نمونه‌گیری را می‌دهد. در آموزش این مدل، از مقدار $\alpha = 0.3$ استفاده شده است. در این مدل، به دلیل چندزبانه بودن، از واژه‌نامه بزرگتری به اندازه ۲۵۰۰۰۰ استفاده شده است، و **tokenizer** استفاده شده در این مدل، **SentencePiece** می‌باشد.

طبق پژوهش صورت گرفته برای ایجاد مدل **mT5** برای سنجش عملکرد آن، این مدل بر روی ۶ کاربرد از **XTREME-multilingual-benchmark** آموزش داده شد است که شامل تشخیص موجودیت‌های دارای نام، برخی مدل‌های پرسش و پاسخ و ... می‌باشد. دقت شود که تمام این تسک‌ها به صورت یک مسئله **text-to-text** داده می‌شوند و عملکرد آن در سه مرحله **zero-shot** که در آن مدل فقط روی دادگان انگلیسی آموزش داده می‌شود، **translate-train** که جواب‌ها از زبان انگلیسی به زبان هدف ترجمه می‌شوند، و **in-language-multitask** ارزیابی می‌شود. در همه این بررسی‌ها مشاهده شده است که سایز مدل، تأثیر مستقیم در بهبود عملکرد مدل دارد، ولی در عین حال، هزینه محاسباتی بالاتری نیز دارد.

۳.۲ مدل ParsBERT

در طی آموزش مدل **mT5**، به مشکلاتی برخوردیم که در طی آن تصمیم گرفته شد که از مدل **ParsBERT**، به جای آن استفاده شود. مدل **ParsBERT** یک مدل بر مبنای **BERT** می‌باشد و مانند آن از مکانیزم‌های **self-attention** برای کشف روابط مفهومی بین کلمات در جمله ورودی استفاده می‌کند [۲]. مدل **ParsBERT** برخلاف مدل **mT5**، به طور اختصاصی برای زبان فارسی ساخته شده و فقط روی دادگان فارسی پیش‌آموزش دیده است، و به همین علت، می‌تواند توانایی بالاتری در شناخت دستور زبان فارسی داشته باشد. همچنین، به علت تک‌زبانه بودن، فقط واژه‌نامه زبان فارسی را دارد و مدل حجم کمتری دارد و آموزش آن راحت‌تر خواهد بود. دقت شود که یکی از تفاوت‌های این مدل با مدل **mT5** این است که مدل **ParsBERT**، یک مدل **Only-Encoder** است و گر نیاز باشد که از این مدل برای آموزش یک مدل **text-to-text** استفاده شود، باید از دو مدل جداگانه برای **encoder** و **decoder** استفاده شود، که در این صورت به مدل حاصل، یک مدل **BERT-2-BERT** گفته می‌شود. در شکل ۲، ساختار یک مدل **BERT-2-BERT** مشاهده می‌شود.

۴.۲ جستجوی دادگان موجود

پس از بررسی مدل‌های موجود، باید دادگان موجود برای کاربرد اصلاح املائی و دستور زبانی پیدا می‌شد. از بین دادگان موجود، دادگان **PerSpellData** انتخاب شد. این دیتاست، مجموعه‌ای از جفت جملات غلط و صحیح می‌باشد که مجموعاً حدود ۴.۶ میلیون رکورد می‌باشد که حدود ۸.۳ میلیون رکورد آن، دارای خطای کلمات بی‌مفهوم می‌باشند [۶].

این دیتاست، از چند منبع به دست آمده است، به این صورت که بخشی از آن از نوشته‌های ثبت شده به صورت

کلمه دارای خطا	کلمه اصلاح شده	نوع خطا
کره	که	کلمه واقعی
یال	یا	کلمه واقعی
وبا	وب	کلمه واقعی
روسن	روشن	کلمه بی مفهوم
اپلیکشین	اپلیکیشن	کلمه بی مفهوم
فناروی	فناوری	کلمه بی مفهوم

جدول ۱: خطای کلمه واقعی و خطای کلمات بی مفهوم

- **عبارات دارای گزار یا گذار:** یکی از اشتباهات رایج دیگر، استفاده "گزار" و "گذار" به جای یکدیگر می باشد. مانند "سپاس گزار" و "گشت و گذار".
- **همزه:** استفاده نادرست از همزه و یا حذف اشتباه آن. مانند "رییس" به جای "رئیس"، "متأسفانه" به جای "متأسفانه".
- **تنوین:** استفاده از تنوین برای کلمات فارسی اشتباه است. مانند "ناچاراً" که شکل صحیح آن، "به ناچار" می باشد.
- و انواع خطاهای دیگر ...

۵.۲ آموزش مدل

پس از انتخاب مدل مورد نظر و دادگان مناسب، فرایند آموزش مدل شروع شد. در ابتدا مدل mT5 را به عنوان مدل پایه انتخاب کردیم. برای آموزش مدل از فریمورک های PyTorch و transformers استفاده شده است. برای آموزش مدل mT5، پیکربندی روند آموزش با توجه به مقاله مدل و کاربرد مورد نیاز پروژه انتخاب شده است که در جدول ۲ ذکر شده است.

دقت شود که برای آموزش این مدل، از روش تطبیق LoRA استفاده شده است. ر این روش، به جای آپدیت کل پارامترها، با استفاده از تجزیه ماتریس پارامترها به ماتریس rank آن و تکنیک های کاهش بعد، فقط بخشی از پارامترهای مدل را آپدیت می کند و بقیه پارامترها ثابت می ماند [۷]. مزیت این روش، سرعت بالاتر آموزش مدل در عین حفظ دقت و عملکرد خوب مدل می باشد. همچنین، حجم دادگانی که برای آموزش به این روش نیاز است به نسبت fine-tuning بسیار کمتر است.

با پیکربندی های ذکر شده، آموزش مدل شروع شد و به آهستگی به هدف خود نزدیک تر می شدیم ولی آموزش مدل با چالش هایی روبه رو شد که یادگیری مدل را متوقف کرد، در ادامه به بررسی دقیق تر این چالش ها می پردازیم.

۶.۲ چالش‌ها و راهکارها

■ فاز اول

در طی آموزش مدل، چالش‌هایی روند آموزش مدل را دشوار کرد و باعث تعویق نتیجه‌دهی مدل شد. مهم‌ترین چالشی که در این پروژه با آن روبه‌رو شدیم، این بود که در طی آموزش هر دو مدل، پیش‌بینی‌های مدل، به رشته‌های خالی می‌انجامید. در این پروژه، در ابتدا، سعی شد که از یک مدل پیش‌آموزش دیده^۲ برای اصلاح متون در زبان اوکراینی برای آموزش بیشتر در زبان فارسی استفاده شود. در روند آموزش مدل تا گام (step) حدود ۵۰۰۰ نیز در حال یادگیری بود، و می‌توانست تا حدی جملات فارسی ورودی را بازتولید و اصلاح کند و خطای آموزش کمتر از ۸.۰ شد. ولی با ادامه آموزش، مدل به سمت پیش‌بینی رشته‌های خالی به عنوان جملات اصلاح شده پیش می‌رود. در جدول ۳، نمونه‌هایی از پیش‌بینی‌های مدل مشاهده می‌شود. در نمونه‌های این جدول، جمله ورودی (input) و جمله هدف (target) به ترتیب عبارت اند از: "ای معلم گل اگر در فصل گل بوئیدنی است" و "ای معلم گل اگر در فصل گل بوئیدنی است".

همان‌طور که مشاهده می‌شود، مدل تا گام حدود ۵۰۰۰، در حال یادگیری و پیشرفت است، ولی از این گام به بعد، رشته‌های به طول صفر به عنوان جمله اصلاح‌شده پیش‌بینی می‌کند. همچنین، در این مرحله مشاهده می‌شود که خطای آموزش^۳ و ارزیابی^۴، همگرا نمی‌شود و تا حد زیادی نوسان دارد. این مشکل در حالی پیش آمده بود که هنوز یک دوره کامل (epoch)، تمام نشده بود.

به دلیل مشکل به وجود آمده و با توجه به جستجوهای که برای رفع مشکل انجام شد، حدس ما بر این بود که مشکل می‌تواند ناشی از ایرادی در مدل باشد، به همین دلیل تصمیم گرفته شد که مدل دیگری برای آموزش انتخاب شود.

■ فاز دوم

مدل دیگری که برای آموزش انتخاب شد، مدل ParsBERT بود. برای آموزش این مدل، به دلیل این که مدل پیش‌آموزش دیده برای تصحیح املائی و دستور زبانی وجود نداشت، از مدل پایه آن استفاده شد. پیکربندی استفاده شده برای روند آموزش برای این مدل، با مدل قبلی یکسان می‌باشد، با این تفاوت که از روش fine-tuning عادی استفاده شد و همچنین، به دلیل اینکه این مدل، یک مدل Encoder - Only می‌باشد، نیاز بود که برای ساخت یک مدل Seq-2-Seq، از یک ساختار Encoder - Decoder استفاده شود. همچنین در آموزش این مدل، از مدل پیش‌آموزش دیده پایه^۵ آن استفاده شد. جدول ۴ تفاوت پیکربندی در روند آموزش این مدل را نشان می‌دهد. پس از تنظیم پیکربندی، آموزش مدل شروع شد و تا گام حدود ۱۰۰۰۰، یادگیری به خوبی انجام می‌شد اما دوباره از این مرحله به بعد، رشته‌های خالی توسط مدل ساخته شد. به همین دلیل، حدس ما بر این بود که مشکل ممکن است از دادگان ورودی به مدل باشد، پس باید فرایندی برای اصلاح و پاکسازی دادگان انجام می‌شد.

■ فاز سوم

این بار سعی شد دادگان ورودی به مدل پاکسازی شوند. برای این کار، با استفاده از متریک میزان خطای کلمه^۶، تمامی جفت جمله‌های دادگان پردازش شد، و هر رکوردی که این خطا برای آن، بزرگتر یا مساوی ۳.۰ بود یا اینکه رشته ورودی یا هدف طول صفر داشت، از دادگان حذف شد. دلیل انتخاب این روش پاکسازی این بود که در بررسی رکوردهای دادگان به صورت دستی، مشاهده شد که رکوردهایی وجود دارند که جمله ورودی یا هدف، رشته خالی هستند یا اینکه، تعداد کلمات جمله ورودی با هدف تفاوت زیادی دارد (که علت آن می‌تواند

^۲<https://huggingface.co/smartist/mt5-small-finetuned-gec-0.2>

^۳Training Loss

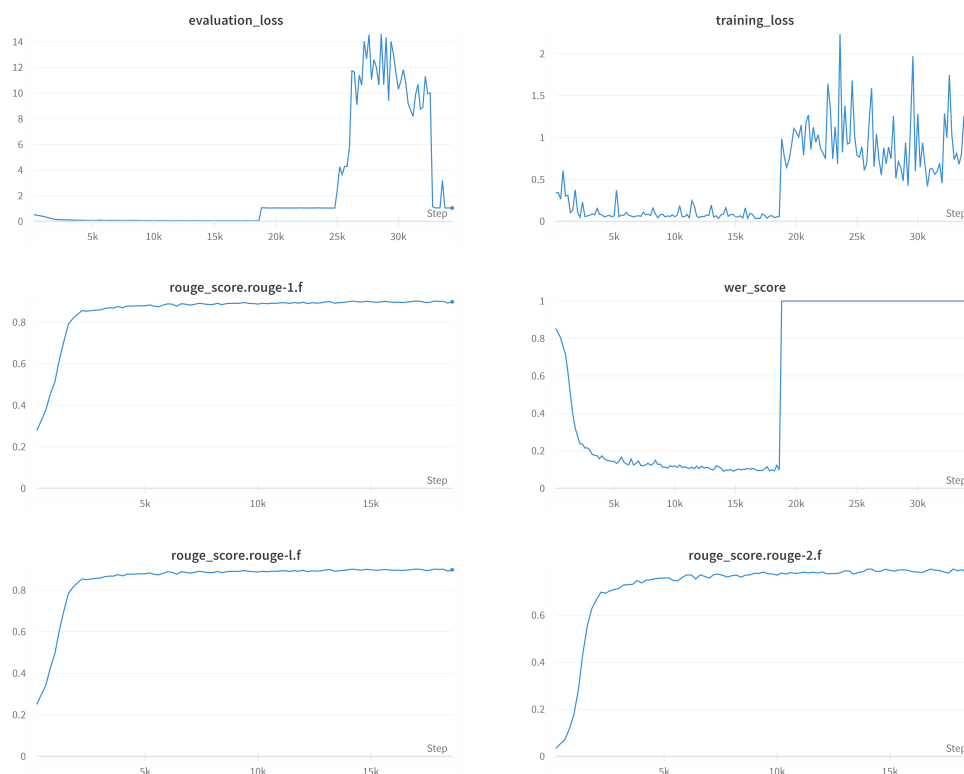
^۴Evaluation Loss

^۵<https://huggingface.co/HooshvareLab/bert-fa-zwnj-base>

^۶Word Error Rate

حذف شدن احتمالی بخشی از عبارت در حین ساخت دادگان باشد). به این ترتیب، دادگان پاکسازی شده به وجود آمد که تنها حدود ۲ درصد حجم کمتری داشت.

پس از ایجاد دادگان پاکسازی شده، دوباره فرایند آموزش مدل ParsBERT را شروع کردیم، و تا حدی امیدوارکننده بود، اما باز هم مشاهده شد که این بار در گام حدود ۱۸۰۰۰، مدل شروع به تولید رشته‌های خالی کرد و خطای آموزش و ارزیابی شروع به نوسان کرد. شکل ۳ نمودار خطاها و متریک‌های اندازه‌گیری شده را نشان می‌دهد.



شکل ۳: از بالا چپ به راست به ترتیب، خطای آموزش، خطای ارزیابی، متریک میزان خطای کلمه، متریک rouge-1 f measure، متریک rouge-2 f measure، متریک rouge-L f measure. دقت شود که همان‌طور که گفته شد، از گام حدود ۱۸۰۰۰ به بعد، رشته‌های خالی تولید می‌شود و به همین دلیل، متریک‌های rouge نمی‌توانند برای آن خروجی‌ها اندازه‌گیری شوند و فقط تا گام ۱۸۰۰۰ اندازه‌گیری شده‌اند.

۷.۲ نتایج آموزش مدل

همان‌طور که گفته شد، بهترین مدلی که توانستم تاکنون در ساعات کارآموزی‌ام، آموزش دهم، مدلی بود که بر مبنای ParsBERT ساخته و آموزش داده شد ولی عملکرد چندان مناسبی ندارد. در جدول ۵، ورودی و خروجی‌هایی از این نسخه مدل، مشاهده می‌شود.

در جدول بالا که چند نمونه اصلاح مدل آورده شده است، مشاهده می‌شود که در جداسازی "به" از عبارت بعدی آن، خوب عمل کرده است ولی در شناسایی جمع‌های مکسر و املائی بعضی کلمات کم کاربرد (مثل پروتکل)، به درستی عمل نکرده است و انتظار داریم که در صورتی که مشکل تولید رشته‌های خالی رفع شود، مدل بتواند دقت خوبی از خود نشان دهد.

۳ نتیجه‌گیری

همان‌طور که گفته شد، در این پروژه سعی شده است که با توجه به مدل‌های موجود، مدلی برای اصلاح املائی و دستور زبانی در زبان فارسی توسعه داده شود. اما تاکنون به دلیل رخداد چالش‌های ذکر شده، مدل توسعه داده شده به دقت خوبی نرسیده است. حدس من بر این است که با پاکسازی دقیق‌تر دادگان موجود، یا ساخت دادگان جدید، بتوان به نتیجه خوبی رسید.

مراجع

- [۱] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, C. Raffel. "mT5: A massively multilingual pre-trained text-to-text transformer", Mar. 2021.
- [۲] M. Farahani, M. Gharachorloo, Ma. Farahani, M. Manthouri. "ParsBERT: Transformer-based Model for Persian Language Understanding", May 2020.
- [۳] M. R. Qorib, S.H. Na, H. T. Ng. "Frustratingly Easy System Combination for Grammatical Error Correction", Jul. 2022.
- [۴] S. Rothe, J. Mallinson, E. Malmi, S. Krause, A. Severyn. "A Simple Recipe for Multilingual Grammatical Error Correction", Aug 2022.
- [۵] K. Omelianchuk, V. Atrasevych, A. Chernodub, O. Skurzhanskyi. "GECToR – Grammatical Error Correction: Tag, Not Rewrite", May 2020.
- [۶] R. Oji, N. Taghizadeh, H. Faili. "PerSpellData: An Exhaustive Parallel Spell Dataset For Persian", 2021.
- [۷] E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen. "LoRA: Low-Rank Adaptation of Large Language Models", Oct. 2021.

نوع پیکربندی	پیکربندی انتخاب شده در این پروژه
Programming Language	Python
Used Frameworks	PyTorch, Huggingface Libraries, Pandas, Numpy
Model	mT5
Checkpoint	smartik/mt5-small-finetuned-gec-0.2
Huggingface Model Class	mT5
Model Maximum Seq. Len.	256
Tokenizer	SentencePiece
Tokenizer Maximum Seq. Len.	256
Tuning Method	LoRA (Using peft library)
Loss Function	CrossEntropyLoss
Data Collator	DataCollatorForSeq2Seq
Initial Learning Rate	1e-3
Epochs	4
Batch Size	6 - 10
Learning Rate Scheduler	Linear
Lora Attention Dimension (r)	8
Lora Scaling parameter (alpha)	16
Lora Target Modules	"q" and "v"
Lora Dropout Probability	0.01
Device Specs	Kaggle Virt. Machine – 12 GB RAM – GPU Tesla P100 16 GB RAM

جدول ۲: پیکربندی آموزش مدل mT5

گام	عبارت اصلاح شده توسط مدل
< ۱۰۰۰	"است گل معلم ای"
۳۰۰۰	"است گل فصل در اگر گل معلم ای"
۴۰۰۰	"است دیدنی گل فصل در اگر گل معلم ای"
> ۵۰۰۰	""

جدول ۳: جملات اصلاح شده توسط مدل در گام‌های متفاوت

نوع پیکربندی	پیکربندی انتخاب شده برای ParsBERT
Model	ParsBERT
Checkpoint	HooshvareLab/bert-fa-zwnj-base
Huggingface Model Class	EncoderDecoderModel
Tokenizer	WordPiece
Tuning Method	Traditional Fine-Tuning (Normal)

جدول ۴: پیکربندی تغییر یافته برای مدل ParsBERT

عبارت ورودی	عبارت هدف	عبارت اصلاح شده توسط مدل
به گونه‌ای که اغلب آنها با چهره‌ای حق بجانب می‌گویند	به گونه‌ای که اغلب آنها با چهره‌ای حق به جانب می‌گویند	به گونه‌ای که اغلب آنها با چهره‌ای حق بهب می‌گویند
بعنوان‌های اعتباری مشهور شد	بعنوان‌های اعتباری مشهور شد	به عنوان‌های اعتباری مشهور شد
سعید شاه‌رخ‌ی با اشاره به حوادث‌ها اخیر در کشور بیان داشت	سعید شاه‌رخ‌ی با اشاره به حوادث اخیر در کشور بیان داشت	سعید شاه‌رخ‌ی با اشاره به حوادث‌های اخیر در کشور بیان داشت
دریافت سفر هفته آینده سلطان قابوس به تهران یک سفر پرتکلی و معمولی نیست	دریافت سفر هفته آینده سلطان قابوس به تهران یک سفر پروتکلی و معمولی نیست	دریافت سفر هفته آینده سلطان قابوس به تهران یک سفر عمیق و معمولی نیست

جدول ۵: ورودی، خروجی و عبارت هدف در بهترین مدل به‌دست آمده تاکنون