معادل سازی هوشمند جملات فارسی توسط شبکه عصبی و یادگیری ماشین

آرمان نیک خواه[1]

[1] دانشجوی کارشناسی علوم کامپیوتر
Arman.nikkhah.79@gmail.com

چکیده

در این مقاله، سعی خواهد شد به نحو پیاده سازی فرآیند معادل سازی جملات فارسی به وسیله شبکه های عصبی و یادگیری ماشین اشاره شود. معادل سازی جملات همواره یکی از نیازمندی های لازم نویسندگان و دانشجویان بوده است چرا که به وسیله آن میتوان ساختار ها را تغییر داد و جملات متفاوتی ایجاد کرد.همچنین این ابزار برای سایر کاربران نیز کاربرد بسیاری دارد که میتوان به تغییر متن ایمیل یا پیامک به وسیله آن اشاره کرد. بر خلاف اینکه نیاز به این ابزار به شدت در محیط آکادمیک کشور احساس میشود اما تا کنون گامی برای تحقق آن برداشته نشده است. به همین سو ما تلاش کردیم که این مشکل را حل و به سمت تولید این ابزار و در دسترس قرار دادن آن به صورت رایگان حرکت کنیم.

کلمات کلیدی

هوش مصنوعی, یادگیری ماشین, پردازش زبان طبیعی, معادل سازی

1- مقدمه

تولید جملات معادل توسط هوش مصنوعی یکی از شاخه های مهم پردازش زبان طبیعی است. برای دستیابی به این مهم نیاز است که چالش های متعددی پشت سر گذاشته شود که در ادامه به آن ها اشاره خواهیم کرد. اولین بودن در این حوزه فرآیند را برای ما جذاب تر کرد چرا که همواره اولین ها در تاریخ به یادگار میمانند. برای دستیابی به این مهم سعی کردیم که کارهای سایر محققان در این زمینه که سعی در پیاده سازی این مدل در زبان انگلیسی را مطالعه و از آن ها الهام بگیریم. مدل های متفاوتی در این زمینه وجود داشتند که ما به بررسی آن پرداخته و بهترین آن را متناسب با شرایط انتخاب کردیم.

2- جمع آوری داده و مشکلات آن

در انجام پروژه های پردازش زبان طبیعی روی داده های فارسی همیشه با مشکل کمبود داده روبرو می شویم .درنتیجه کمیت و کیفیت داده ها در بحث معادل سازی جمالت در زبان فارسی نیز کم است. به همین دلیل ما داده های خود را با استفاده از ترجمه ی دیتاست های زبان انگلیسی مرتبط با این مبحث به دست آوردیم. برای انجام این کار از دیتاست Paranmt 50m استفاده کردیم .این دیتاست شامل 50 میلیون جفت جمله معادل به زبان انگلیسی است و حجمی معادل 11 گیگابایت دارد.این دیتاست مشکلات عمده ای داشت که فرآیند استفاده ار آن را دشوار میکرد.برای مثال تعدادی از جملات آن دقیقا یکسان بودند و در طرف مقابل تعدادی از جملات هیچ ارتباطی با هم نداشتند. بنابراین لازم بود پیش پردازش هایی در این زمینه روی آن ها انجام شود. با استفاده از در نظر گرفتن تعداد کلمات مشابه در جفت جمالت درصد شباهت جمالت یک جفت نسبت به یکدیگر را محاسبه کردیم. در نتیجه جمالتی که دارای شباهت 100 ٪هستند یعنی هیچ تفاوتی با هم ندارند و دقیقا با هم یکسان هستند و نمی توانند چیزی به مدل آموزش دهند پس آن ها را حذف کردیم. همینطور جفت جمالتی که دارای شباهت زیر 15 ٪هستند نیز تاثیری مثبتی در یادگیری مدل ندارند به این علت که می شود گفت این جفت جمالت تا حد زیادی معادل نیستند .جمالت با طول زیاد میتوانند باعث کند شدن روند یادگیری مدل شوند در نتیجه ما جمالتی با طول بالاتر 50 کلمه را از دیتاست حذف کردیم همچنین حجم بالای این دیتاست بارگیری آن روی حافظه کامپیوتر را بسیار سخت میکرد. به همین دلیل باید سعی میکردیم آن را به صورت قسمت های کوچک تر وارد حافظه و سپس پردازش های لازم را روی آن انجام میدادیم .ما توانستیم با پیاده سازی Google Translate با کمک زبان پایتون بخشی از داده ها را ترجمه کنیم. اما به دلیل حجم زیاد دیتاست و محدود بودن مقدار رم در گوگل کولب نمیتوانستیم کل دیتاست را به صورت یکجا ترجمه کنیم. در نتیجه با بخش کردن دیتاست 8 قسمت مساوی توانستیم کل دیتاست را به زبان فارسی

ترجمه کنیم . پس از ترجمه نیز مشکلات زیادی پیش پای ما قرار گرفت که میتوان به عدم ترجمه شدن بخشی از کلمات به علت وجود نداشتن معادل برای آن ها اشاره کرد. همچنین بخشی از جملات با وجود متفاوت بودن در زبان مبدا پس از ترجمه دقیقا مشابه یکدیگر میشدند. برای حل این مسائل از معیار فاصله ی بین جملات به اسم فاصله لون‌اشتاین استفاده کردیم. فاصله لون‌اشتاین یا فاصله ویرایش در نظریه اطلاعات و علوم کامپیوتر مقیاسی برای محاسبه‌ی میزان تفاوت میان دو رشته است. فاصله لون‌اشتاین بین دو رشته به وسیله‌ی کمترین تعداد عملیات مورد نیاز برای تبدیل یک رشته به رشته دیگر معین می‌شود، که یک عملیات می‌تواند یک ضمیمه، یا جایگزینی یک کارکتر باشد. تعمیم فاصله لوناشتاین (فاصله دامرا-لون‌اشتاین) اجازه ترانهش دو کاراکتر را به عنوان یک عملیات می‌دهد. این معیار به افتخار ولادمیر لون‌اشتاین، که این فاصله را در سال ۱۹۵۶ مطرح کرد، نام گذاری شده‌است. همچنین از این موضوع در برنامه‌هایی که نیاز به یافتن مقدار شباهت، یا تفاوت دو رشته را دارند، مانند مقابله گر املائی، استفاده می‌شود. همچنین برای حل مکشل وجود کلمات انگلیسی اقدام به حذف کلی داده هایی که دارای کارکتر های انگلیسی هستند کردیم.

2-1- دیتاست نهایی

در زیر میتوانید نمونه ای از داده های ترجمه شده را مشاهده کنید. ما این دیتاست را به صورت دسترسی آزاد برای همگان روی کگل قرار داده ایم. همانطور که مشاهده میشود داده ها در سه ستون دسته بندی شده اند. ستون اول نوع وظیفه ی تعریف شده را مشخص میکند که در این مورد مطابق فرمت رسمی مدل های T5 به صورت paraphrase تعریف شده است. دو ستون بعدی مربوط به جملات معادل به هم هستند که ستون اول جمله اصلی و ستون آخر جمله ی معادل با آن است. همانطور که مشاهده میشود برخی ار داده ها دقیق نیستند اما به دلیل وجود مقدار زیاد داده این نویز ها در مرحله آموزش مدل نادیده گرفته میشوند.

prefix	input_text	target_text
1 unique value	614104 unique values	583263 unique values
paraphrase	بنابراین من به شما نیاز دارم که از زندگی شخصی من دور بمانید.	من به شما نیاز دارم که از زندگی شخصی من دور بمانید.
paraphrase	احتمالا شما مادر بسیار سلطه پذیری داشتید.	احتمالاً به این دلیل که شما یک مادر بسیار مسلط داشتید.
paraphrase	وحشتناک به نظر می رسد.	وحشتناک به نظر می رسد
paraphrase	نه، نه، این نیست.	نه، اون نه.
paraphrase	می دانید چه کسی پوست بدی داشت و چه کسی موهای بدی داشت.	می دانید، پوست بد و موی بد.
paraphrase	مطمئنی که نمیتونم از این موضوع با تو حرف بزنم	مطمئنی که من تو را از آن حرف نمی زنم
paraphrase	پایلوت به این نتیجه رسید که مقامات ذیصلاح ملی که در آزمایشی شرکت کردند، در موقعیتی هستند که نرخ ساعت...	پروژه آزمایشی به این نتیجه رسید که مقامات ملی مربوطه که در پروژه آزمایشی شرکت کردند، می توانند نرخ س...
paraphrase	چی	وای چی
paraphrase	به همین دلیل آن اعتراضات بسیار مهم بود.	به همین دلیل این اعتراضات بسیار مهم بود.
paraphrase	مثل عکس زیباست	او مثل یک عکس زیباست
paraphrase	شب قبل سعی کرده بود او را بکشد.	شب قبل سعی کرد او را بکشد.
paraphrase	آرام بخش، درست است	اطمینان بخش، درست است
paraphrase	زن خانواده اش را در خانه ای	زنی خانواده اش را در خانه ای

3- مدل

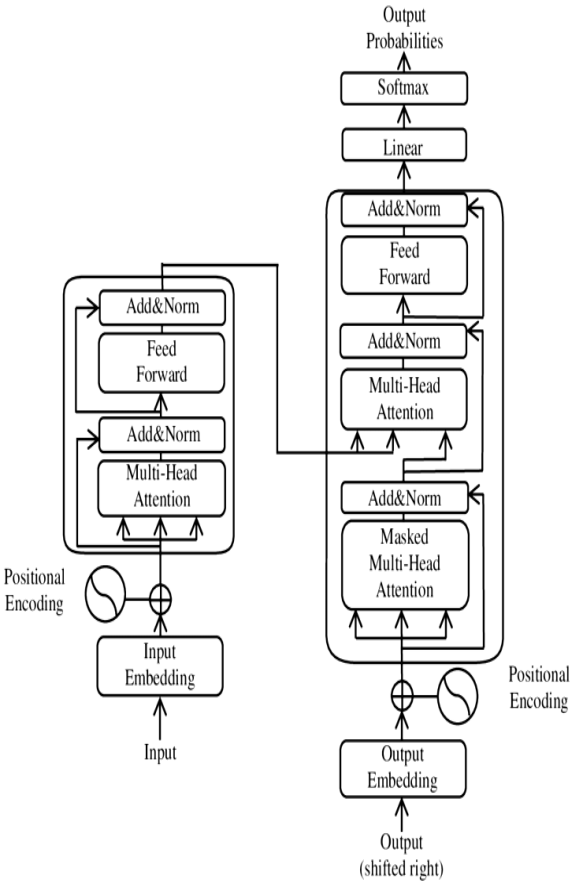
در حال حاضر بهترین مدل ها در زمینه پردازش زبان طبیعی مدل هایی با ساختار Transformers هستند. مدل Transformer(Transfer Text-to-Text (T5 نیز از همین ساختار استفاده کرده . T5یک مدل زبانی است که توسط Google توسعه داده شده و هدف آن بهبود عملکرد یادگیری انتقالی است . یادگیری انتقالی به این معناست که در ابتدا یک مدل را ابتدا برای یک عمل که داده های غنی و جامعی دارد پیش آموزش میدهیم سپس مدل را برای عمل مورد نظر خود با استفاده از داده های متناسب با عمل بهینه کنیم. در پردازش زبان طبیعی اغلب از این روش برای استفاده میشود به دلیل اینکه با پیش آموزش مدل روی حجم زیادی از داده ها مدل درک بهتری از زبان و ساختار آن پیدا خواهد کرد و میتواند سریع تر عمل مورد نظر را یاد بگیر. ما از مدل mT5 در این پروژه استفاده کردیم که ساختاری مشابه با مدل T5 دارد. تنها تفاوت این دو مدل در دیتاست هایی است که روی آن پیش آموزش دیده اند. مدل mT5 روی دیتاست mC4 پیش آموزش داده شده. این دیتاست شامل داده های متنی به 108 زبان مختلف است که زبان فارسی یکی از آن هاست. حجم داده فارسی موجود در این دیتاست برابر با 220 گیگابایت است

در نتیجه این مدل درک مناسبی از زبان فارسی دارد در نتیجه انتخاب مناسبی برای این پروژه است.

3-1- ساختار مدل

همانطور که در شکل زیر مشاهده می کنید ساختار مدل mT5 یک ساختار استاندارد Transformer هست. این مدل دو قسمت کدگذ (Encoder) و کدگشا (Decoder)است بعلاوه در آخر یک الیه شبکه عصبی خطی(Linear) دارد که روی آن تابع SoftMax اعمال می شود و خروجی نهایی تولید میشود .قسمت Encoder داری دو لایه Self -Attention و Feed Forward است- Self Attentionبا استفاده از مکانیزم Attention ارتباط میان دو کلمه در جمله را پیدا میکند. این به ما کمک میکند که درک درستی از جمله پیدا کنیم. به عنوان مثال جمله زیر را در نظر بگیرید :من آب درون بطری را داخل لیوان ریختم تا آن پر شود .در این جمله واضح است کلمه آن به کلمه لیوان اشاره دارد حال جمله بعدی را در نظر بگیرید : من آب درون بطری را داخل لیوان ریختم تا آن خالی شود .همانطور که می بینید با تغییر دادن فقط یک کلمه در جمله معنای کلمه آن تغییر کرد. اینجا کلمه آن به کلمه بطری اشاره دارد. با استفاده از مکانیزم Attention میتوان ارتباط کلمه را با سایر کلمات در جمله با استفاده از مفهوم کلی جمله محاسبه کرد.

لایه Feed Forward در قسمت Encoder یک شبکه شامل وزن هایی است که در طی فرایند یادگیری مقدار مناسب را بدست می آورند .قسمت Decoder عالوه بر دو لایه ای که در قسمت Encoder توضیح داده شد یک لایه دیگر به نام -Attention Decoder Encoder دارد. این لایه همانند Self- Attention عمل می کند با این تفاوت که ارتباط بین کلمات در جمله ورودی و جمله خروجی را محاسبه میکند. لازم به ذکر است که قسمت Decoderجمله خروجی را به عنوان ورودی دریافت می کند و جمله ورودی اصلی را از قسمت Encoderمی گیرد.

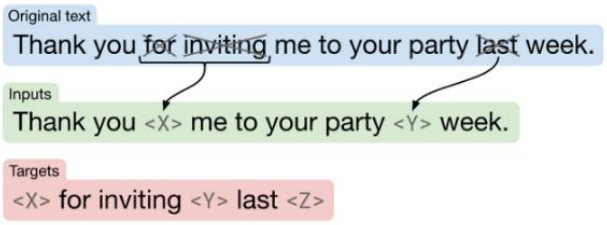


3-2- مراحل آموزش مدل

آموزش این مدل در دومرحله صورت گرفته. 1 :آموزش بدون نظارت روی داده های. 2 mC4 آموزش نظارت شده روی داده های جمع آوری شده برای معادل سازی جمالت آموزش بدون نظارت در این مرحله با استفاده از روش Spans Corrupting یا همان Objective Denoisingروی داده های دیتاست mC4 اعمال می شود. دیتاست mC4شامل متن هایی از بستر وب هستند با استفاده از چند خزنده جمع آوری شده اند. این دیتاست حاوی داده های تمیز است به این معنی که عالیم نگارشی هشتگ ها و ...از متن ها پاک شده است. روش Objective Denoising به این صورت عمل میکند که متنی را از دیتاست انتخاب میکند، سپس قسمتی از جمله که شامل تعداد دلخواهی از کلمات است را می پوشاند، در ادامه سعی می کند که با استفاده از مدل کلمات پوشیده شده را حدس بزند. به این ترتیب عمل یادگیری فقط با استفاده از یک متن برچسب گذاری نشده انجام می شود. این قسمت از آموزش توسط افرادی که مدل را معرفی و پیاده سازی کردند ارایه شده است. به زبانی دیگر ما یک مدل از پیش آموزش دیده شده داریم.

آموزش با نظارت در این مرحله ما با استفاده از مدل mT5 و دیتاستی که جمع آوری کرده ایم مدل را برای عمل معادل سازی جملات آموزش می دهیم. برای پیاده سازی این پروژه ما از زبان پایتون

استفاده کردیم که کتابخانه های غنی و متعددی در زمینه پردازش عمیق و پردازش زبان های طبیعی دارد. در دیتاست ما هر مثال آموزشی شامل دو جمله میشود. جمله اول به عنوان جمله اصلی و ورودی مدل و جمله دوم به عنوان جمله معادل سازی شده با استفاده از جمله اصلی و خروجی یا همان هدف مدل است. درنتیجه روش آموزش استفاده شده روش یادگیری با نظارت است.



3-3- توکن سازی(tokenization) و embedding

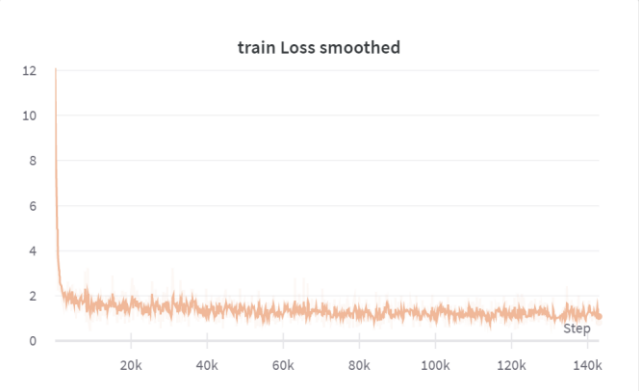
Tokenization اولین مرحله در هر تسک NLP است و تأثیر مهمی بر روند آموزش مدل دارد. توکنایزر داده های بدون ساختار و متن زبان طبیعی را به تکه هایی از اطلاعات که می توانند به عنوان عناصر مجزا در نظر گرفته شوند، تجزیه می کند.

توکنایز کردن دارای انواع مختلفی است که ما در این پروژه مطابق با استاندارد مدل های T5 از تکنیک sub-word برای توکن سازی استفاده کردیم. کتابخانه ای که برای اینکار انتخاب کردیم کتابخانه sentencePiece میباشد که توسط گوگل طراحی شده است. SentencePiece یک توکنایزر و دی-توکنایزر (تبدیل توکن به متن) بدون نظارت است که عمدتاً برای سیستم‌های تولید متن مبتنی بر شبکه عصبی است که در آن اندازه واژگان قبل از آموزش مدل عصبی تعیین شده است.

همچنین میدانیم که کامپیوتر ها تنها قادر به درک اعداد هستند پس برای فهم جملات و کلمات به زبان انسانی نیاز به تبدیل تک تک کلمات به بردار هایی معادل هستند. به این عملیات word embedding گفته میشود. embedding یک نمایش آموخته شده برای متن است که در آن کلماتی که معنی یکسانی دارند بازنمایی مشابهی دارند. این رویکرد در نمایش کلمات و اسناد را میتوان یکی از پیشرفت های کلیدی یادگیری عمیق در مشکلات چالش برانگیز پردازش زبان طبیعی در نظر گرفت. word embedding در واقع دسته‌ای از تکنیک‌ها هستند که در آن کلمات جداگانه به‌عنوان بردارهایی با ارزش حقیقی در یک فضای برداری از پیش تعریف‌شده نمایش داده می‌شوند. هر کلمه به یک بردار نگاشت می شود و مقادیر بردار توسط یک شبکه عصبی آموخته می شوند، و از این رو این تکنیک اغلب در حوزه یادگیری عمیق قرار می گیرد.

4- نتایج

برای آموزش چنین مدل های سنگینی همواره نیاز به وجود GPU (کارت گرافیک) قدرتمند است. این کارت های گرافیک امکان موازی سازی عملیات های لایه ها را محیا میسازند و تاثیر چشم گیری در سرعت پردازش خواهند داشت. برای آموزش این مدل ما از کارت گرافیک NVIDIA TESLA K80 استفاده کردیم. این کارت گرافیک دارای 24 گیگ حافظه GDDR5 میباشد. در زیر نمودار خطای مدل پس از 12 ساعت آموزش را مشاهده میکنید.



همانطور که مشاهده میشود خطای مدل در ابتدا بسیار بالاست و علت آن را میتوان به عدم آشنایی مدل به تسک معرفی شده ریشه یابی کرد. اما این خطا پس از چند step با شیب زیادی کاهش میابد که گواه این است که مدل به درک درستی از مسئله نزدیک شده است و اکنون میتواند جملات را به گونه ای تغییر دهد که خطای پایینی حاصل شود.

5- توسعه های آتی

این مدل با وجود تلاش های ما دارای نقصان هایی است که قابل بهبود میباشند. مهم ترین گام برای بهبود این مدل اقدام به تهیه ی دیتاستی دقیق از جملات معادل فارسی است که این مهم بـدون صـرف وقـت و هزینه زیاد میسر نخواهد بود. در گام بعدی باید سیستم های گرافیکـی قوی تری اختیار کرد که پروسه آموزش مـدل را سـریع تـر و بـا دقـت بیشتری پیش برد. پس از انجام گام های یادشده میتوان این مدل را به مرحله استفاده تجاری و درآمد زایی رساند.

6- قدردانی

با تشکر از استاد عزیزم جناب آقای دکتر میرزایی بابت همراهی و همیاریشان در این پروژه.

مراجع

[1] Bird, Jordan J., Anikó Ekárt, and Diego R. Faria. "Chatbot Interaction with Artificial Intelligence: human data augmentation with T5 and language transformer ensemble for text classification." *Journal of Ambient Intelligence and Humanized Computing* (2021): 1-16.

[2] Hudson, G. Thomas, and Noura Al Moubayed. "Ask me in your own words: paraphrasing for multitask question answering." *PeerJ Computer Science* 7 (2021): e759.

[3] Chada, Rakesh. "Simultaneous paraphrasing and translation by fine-tuning transformer models." *arXiv preprint arXiv:2005.05570* (2020).

[4] Witteveen, Sam, and Martin Andrews. "Paraphrasing with large language models." *arXiv preprint arXiv:1911.09661* (2019).

[5] Egonmwan, Elozino, and Yllias Chali. "Transformer and seq2seq model for paraphrase generation." *Proceedings of the 3rd Workshop on Neural Generation and Translation*. 2019.

[6] Xue, Linting, et al. "mT5: A massively multilingual pre-trained text-to-text transformer." *arXiv preprint arXiv:2010.11934* (2020).

[7] Chi, Zewen, et al. "mT6: Multilingual pretrained text-to-text transformer with translation pairs." *arXiv preprint arXiv:2104.08692* (2021).

[8] Wieting, John, and Kevin Gimpel. "ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations." *arXiv preprint arXiv:1711.05732* (2017).

[9] Wieting, John, Jonathan Mallinson, and Kevin Gimpel. "Learning paraphrastic sentence embeddings from back-translated bitext." *arXiv preprint arXiv:1706.01847* (2017).

[10] Wieting, John, et al. "Towards universal paraphrastic sentence embeddings." *arXiv preprint arXiv:1511.08198* (2015).

[11] Hu, J. Edward, et al. "Parabank: Monolingual bitext generation and sentential paraphrasing via lexically-constrained neural machine translation." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 2019.

[12] Grover, Khushnuma, et al. "Deep learning based question generation using t5 transformer." *International Advanced Computing Conference*. Springer, Singapore, 2020