دانشکدگان علوم
دانشکده ریاضی، آمار و علوم کامپیوتر

شناسایی نویسنده متون فارسی با بکارگیری مدل BERT

نگارنده

سپهر عبّاسپور

استاد راهنما: دکتر هدیه ساجدی

پایان‌نامه برای دریافت درجه کارشناسی
در رشته علوم کامپیوتر

مرداد ۱۴۰۳

چکیده

یکی از کاربردهای اساسی در حوزه هوش مصنوعی و یادگیری ماشین، دسته بندی مجموعه ای از اعضا در دسته های از پیش تعریف شده است و مسئله تشخیص و الصاق یک نویسنده به یک یا چند متن نیز نمونه ای از این کاربرد است. هر کلمه می تواند شامل چندین معنی مخصوصاً به هنگام قرار گرفتن در کنار کلمات دیگر باشد؛ بنابراین استخراج ویژگی های کلمات با چالش هایی مواجه بوده است. در این راستا به عنوان یک تلاش اولیه، مدل‌های تعبیه‌ی کلمات برای مسائلی که با متن سروکار دارند ارائه شدند هرچند این مدل‌ها عمیق نبوده و حاوی اطّلاعات کمی بودند و به عبارت دیگر، کاربرد آن‌ها مؤثر، امّا محدود بود. بدین سان در سال ۲۰۱۸، مهندسان گوگل مدل بزرگ و قدرتمندتری به نام BERT را با دادگان بسیاری آموزش داده، و آن را در دسترس عموم قرار دادند. این مدل در واقع دسته‌ای از رمزگذارهای آموزش‌دیده‌ی مدل ترنسفورمر است و هدف از این مطالعه، تشخیص نویسنده‌ها از روی متون فارسی با استفاده از یک مدل مبتنی بر این مدل به نام ParsBERT بوده و میزان کارآمد بودن آن نیز مورد تحلیل و تفسیر قرار می گیرد.

سپاس‌گزاری

با سپاس فراوان از استاد راهنمای گرامی، دکتر ساجدی که با راهنمایی های ارزشمند خود، من را در این پژوهش یاری نمودند. همچنین از والدینم و دوستان عزیزم یگانه و حسین نیز سپاسگزارم.

فهرست مطالب

فهرست تصاویر

فهرست جداول

پیشگفتار

تشخیص نویسنده یک متن می تواند عملی چالش برانگیز باشد؛ در این صورت باید با مطالعه و کشف نوع نوشتار هر نویسنده هدف، ابرازی بدست آورد تا بوسیله آن، متون دارای نویسنده مجهول بازشناسی و آن نویسنده را شناسایی کرد. از آنجا که اغلب نویسندگان از الگوهای خاص خود در نگارش بهره می برند و شناسایی این الگوها از توابع محسوس ریاضیاتی پیروی نمی کند، اگر از آن نویسندگان به مقدار کافی نوشته موجود باشد، می توان از علم یادگیری ماشین برای کشف چنین الگوها استفاده کرد.
یک نوشته، مجموعه ای از اجزا (کلمات) است که اغلب معنای واحدی نداشته و با در کنار یکدیگر قرار گرفتن، معانی مجزایی تولید میکنند. در این صورت باید از رویکرد مبتنی بر حافظه مانند شبکه های عصبی بازگشتی یا شبکه های عصبی با رویکرد نظارت و یادگیری دو سویه بهره برد که یکی از این مدل ها، مدل پیش آموزش دیده BERT [۱] است، مدلی که روی متون انگلیسی متعدد آموزش دیده و حال در حوزه های مختلف مرتبط با علم پردازش زبان طبیعی، کاربرد دارد.

فصل ۱

مفاهیم مقدماتی

ابتدا قصد داریم تا مفاهیمی پایه را چنان شرح دهیم تا مسیر اتصال بین مسئله مبتنی بر متن به زبان طبیعی انسان و سیستم های رایانه ای به خوبی تبیین گردد. این مفاهیم در دو بخش شبکه های عصبی و پردازش زبان طبیعی توصیف و تشریح می گردند.

۱.۱ شبکه های عصبی

شبکه‌های عصبی مصنوعی[۱] یا به زبان ساده‌تر شبکه‌های عصبی سیستم‌ها و روش‌های محاسباتی نوین برای یادگیری ماشینی، نمایش دانش و در انتها اعمال دانش به دست آمده در جهت بیش‌بینی پاسخ‌های خروجی از سامانه‌های پیچیده هستند. ایدهٔ اصلی این گونه شبکه‌ها تا حدودی الهام‌گرفته از شیوهٔ کارکرد سیستم عصبی زیستی برای پردازش داده‌ها و اطلاعات به منظور یادگیری و ایجاد دانش می‌باشد. عنصر کلیدی این ایده، ایجاد ساختارهایی جدید برای سامانهٔ پردازش اطلاعات است. این سیستم از شمار زیادی عناصر پردازشی فوق‌العاده بهم‌پیوسته با نام نورون تشکیل شده که برای حل یک مسئله با هم هماهنگ عمل می‌کنند و توسط سیناپس‌ها (ارتباطات الکترومغناطیسی) اطلاعات را منتقل می‌کنند. در این شبکه‌ها اگر یک سلول آسیب ببیند بقیه سلول‌ها می‌توانند نبود آن را جبران کرده، و نیز در بازسازی آن سهیم باشند. این شبکه‌ها قادر به یادگیری‌اند؛ مثلاً با اعمال سوزش به سلول‌های عصبی لامسه، سلول‌ها یادمی‌گیرند که به طرف جسم داغ نروند و با این الگوریتم سیستم می‌آموزد که خطای خود را اصلاح کند. یادگیری در این سیستم‌ها به صورت تطبیقی صورت می‌گیرد، یعنی با استفاده از مثال‌ها وزن سیناپس‌ها به گونه‌ای تغییر می‌کند که در صورت دادن ورودی‌های جدید، سیستم پاسخ درستی تولید کند.

---

[۱] Artificial Neural Network (ANN)

۱.۱.۱ زمینه پیدایش

فلسفهٔ اصلی شبکهٔ عصبی مصنوعی، مدل کردن ویژگی‌های پردازشی مغز انسان برای تقریب زدن روش‌های معمول محاسباتی با روش پردازش زیستی است. به بیان دیگر، شبکهٔ عصبی مصنوعی روشی است که دانش ارتباط بین چند مجموعهٔ داده را از طریق آموزش فراگرفته و برای استفاده در موارد مشابه ذخیره می‌کند. این پردازنده از دو جهت مشابه مغز انسان عمل می‌کند:

۱. یادگیری شبکهٔ عصبی از طریق آموزش صورت می‌گیرد.

۲. وزن‌دهی مشابه با سیستم ذخیره‌سازی اطلاعات، در شبکهٔ عصبی مغز انسان انجام می‌گیرد.

۲.۱.۱ تاریخچه شبکه‌های عصبی مصنوعی

از قرن نوزدهم به‌طور همزمان اما جداگانه، از سویی نوروفیزیولوژیست‌ها سعی کردند سیستم یادگیری و تجزیه و تحلیل مغز را کشف کنند، و از سوی دیگر ریاضیدانان تلاش کردند مدل ریاضی ای بسازند که قابلیت فراگیری و تجزیه و تحلیل عمومی مسائل را دارا باشد.
اولین کوشش‌ها در شبیه‌سازی با استفاده از یک مدل منطقی در اوایل دههٔ ۱۹۴۰ توسط وارن مک‌کالک و والتر پیتز انجام شد که امروزه بلوک اصلی سازنده اکثر شبکه‌های عصبی مصنوعی است. عملکرد این مدل مبتنی بر جمع ورودی‌ها و ایجاد خروجی با استفاده از شبکه‌ای از نورون‌ها است. اگر حاصل جمع ورودی‌ها از مقدار آستانه بیشتر باشد، اصطلاحاً نورون برانگیخته می‌شود. نتیجه این مدل اجرای ترکیبی از توابع منطقی بود.
در سال ۱۹۴۹ دونالد هب قانون یادگیری را برای شبکه‌های عصبی طراحی کرد. در سال ۱۹۵۸ شبکه پرسپترون توسط روزنبلات معرفی گردید. این شبکه نظیر واحدهای مدل شده قبلی بود. پرسپترون دارای سه لایه است که شامل لایهٔ ورودی، لایهٔ خروجی و لایهٔ میانی می‌شود. این سیستم می‌تواند یاد بگیرد که با روشی تکرارشونده وزن‌ها را به گونه‌ای تنظیم کند که شبکه توان بازتولید جفت‌های ورودی و خروجی را داشته‌باشد.
روش دیگر، مدل خطی تطبیقی نورون است که در سال ۱۹۶۰ توسط برنارد ویدرو و مارسیان هاف در دانشگاه استنفورد) به وجود آمد که اولین شبکه‌های عصبی به کار گرفته شده در مسائل واقعی بودند. آدالاین یک دستگاه الکترونیکی بود که از اجزای ساده‌ای تشکیل شده بود، روشی که برای آموزش استفاده می‌شد با پرسپترون فرق داشت.
در سال ۱۹۶۹ میسکی و پاپرت کتابی نوشتند که محدودیت‌های سیستم‌های تک لایه و چند لایه پرسپترون را تشریح کردند. نتیجه این کتاب پیش داوری و قطع سرمایه‌گذاری برای تحقیقات در زمینه شبیه‌سازی شبکه‌های عصبی بود. آن‌ها با طرح اینکه طرح پرسپترون قادر به حل هیچ مسئله جالبی نمی‌باشد، تحقیقات در این زمینه را برای مدت چندین سال متوقف کردند. با وجود اینکه اشتیاق عمومی و سرمایه‌گذاری‌های موجود به حداقل خود رسیده بود، برخی محققان تحقیقات خود را برای ساخت ماشین‌هایی که توانایی حل مسائلی از قبیل تشخیص الگو را داشته باشند، ادامه دادند. از جمله گراسبگ که شبکه‌ای تحت عنوان Avalanch را برای تشخیص صحبت پیوسته و کنترل دست ربات مطرح کرد. همچنین او با همکاری کارپنتر شبکه‌های نظریه تشدید انطباقی را بنا نهادند که با مدل‌های طبیعی تفاوت داشت. اندرسون و کوهونن نیز از اشخاصی بودند که تکنیک‌هایی برای یادگیری ایجاد کردند.

ورباس در سال ۱۹۷۴ شیوه آموزش پس انتشار خطا را ایجاد کرد که یک شبکه پرسپترون چندلایه البته با قوانین نیرومندتر آموزشی بود.
پیشرفت‌هایی که در سال ۱۹۷۰ تا ۱۹۸۰ به‌دست آمد، برای جلب توجه به شبکه‌های عصبی بسیار مهم بود. برخی فاکتورها نیز در تشدید این مسئله دخالت داشتند، از جمله کتاب‌ها و کنفرانس‌های وسیعی که برای مردم در رشته‌های متنوع ارائه شد. امروز نیز تحولات زیادی در تکنولوژی ANN ایجاد شده‌است.

۳.۱.۱ تعریف رسمی

یک شبکهٔ عصبی مصنوعی، از سه لایهٔ ورودی، خروجی و پردازش تشکیل می‌شود. هر لایه شامل گروهی از سلول‌های عصبی (نورون) است که عموماً با کلیهٔ نورون‌های لایه‌های دیگر در ارتباط هستند، مگر این که کاربر ارتباط بین نورون‌ها را محدود کند؛ ولی نورون‌های هر لایه با سایر نورون‌های همان لایه، ارتباطی ندارند.
نورون کوچک‌ترین واحد پردازشگر اطلاعات است که اساس عملکرد شبکه‌های عصبی را تشکیل می‌دهد. یک شبکهٔ عصبی مجموعه‌ای از نورون‌هاست که با قرار گرفتن در لایه‌های مختلف، معماری خاصی را بر مبنای ارتباطات بین نورون‌ها در لایه‌های مختلف تشکیل می‌دهند. نورون می‌تواند یک تابع ریاضی غیرخطی باشد، در نتیجه یک شبکهٔ عصبی که از اجتماع این نورون‌ها تشکیل می‌شود، نیز می‌تواند یک سامانهٔ کاملاً پیچیده و غیرخطی باشد. در شبکهٔ عصبی هر نورون به‌طور مستقل عمل می‌کند و رفتار کلی شبکه، برآیند رفتار نورون‌های متعدد است. به عبارت دیگر، نورون‌ها در یک روند همکاری، یکدیگر را تصحیح می‌کنند. یادگیری شبکهٔ عصبی از طریق آموزش صورت می‌گیرد. وزن‌دهی مشابه با سیستم ذخیره‌سازی اطلاعات، در شبکهٔ عصبی مغز انسان انجام می‌گیرد.

۴.۱.۱ کاربرد

با استفاده از دانش برنامه‌نویسی رایانه می‌توان ساختار داده‌ای طراحی کرد که همانند یک نورون عمل نماید. سپس با ایجاد شبکه‌ای از این نورون‌های مصنوعی به هم پیوسته، ایجاد یک الگوریتم آموزشی برای شبکه و اعمال این الگوریتم به شبکه آن را آموزش داد. این شبکه‌ها برای تخمین و تقریب، کارایی بسیار بالایی از خود نشان داده‌اند. گستره کاربرد این مدل‌های ریاضی بر گرفته از عملکرد مغز انسان، بسیار وسیع می‌باشد که به عنوان چند نمونه کوچک می‌توان استفاده از این ابزار ریاضی در پردازش سیگنال‌های بیولوژیکی، مخابراتی و الکترونیکی تا کمک در نجوم و فضانوردی را نام برد. اگر یک شبکه را هم‌ارز با یک گراف بدانیم، فرایند آموزش شبکه تعیین نمودن وزن هر یال و پایه اولیهٔ خواهد بود.

۵.۱.۱ یادگیری

یادگیری ماشین با نظارت به دنبال تابعی از میان یک سری توابع هست که تابع هزینه داده‌ها را بهینه سازد. به عنوان مثال در مسئله رگرسیون تابع هزینه می‌تواند اختلاف بین پیشینی و مقدار واقعی

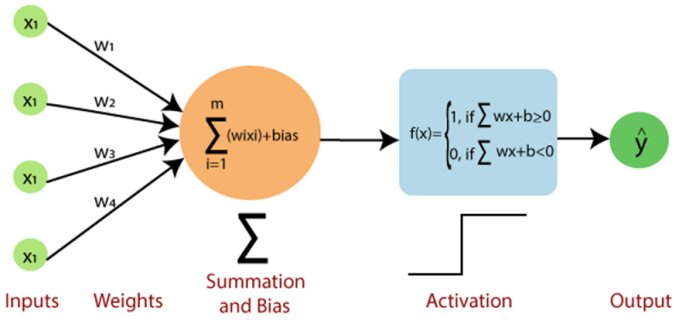
خروجی به توان دو باشد، یا در مسئله طبقه‌بندی ضرر منفی لگاریتم احتمال خروجی باشد. مشکلی که در یادگیری شبکه‌های عصبی وجود دارد این است که این مسئله بهینه‌سازی دیگر محدب نیست. ازین رو با مشکل کمینه‌های محلی روبرو هستیم. یکی از روش‌های متداول حل مسئله بهینه‌سازی در شبکه‌های عصبی بازگشت به عقب یا همان است. الگوریتم پس انتشار خطا گرادیانِ تابع هزینه را برای تمام وزن‌های شبکه عصبی محاسبه می‌کند و بعد از روش‌های گرادیان کاهشی برای پیدا کردن مجموعه وزن‌های بهینه استفاده می‌کند. روش‌های گرادیان کاهشی سعی می‌کنند به صورت متناوب در خلاف جهت گرادیان حرکت کنند و با این کار تابع هزینه را به حداقل برسانند. پیدا کردن گرادیانِ لایه آخر ساده است و با استفاده از مشتق جزئی بدست می‌آید. گرادیانِ لایه‌های میانی اما به صورت مستقیم بدست نمی‌آید و باید از روش‌هایی مانند قاعده زنجیری در مشتق‌گیری استفاده کرد. الگوریتم پس انتشار خطا از قاعده زنجیری برای محاسبه گرادیان‌ها استفاده می‌کند و همان‌طور که در پایین خواهیم دید، این روش به صورت متناوب گرادیان‌ها را از بالاترین لایه شروع کرده آن‌ها را در لایه‌های پایینتر پخش می‌کند.

۶.۱.۱ انواع شبکه های عصبی

انواع مختلفی از شبکه‌های عصبی وجود دارند که به لحاظ ساختار، جریان داده، تعداد و نوع نورون‌های لایه‌ها، تعداد لایه‌ها و سایر موارد با یکدیگر تفاوت دارند. در ادامه، به توضیح ساختار درونی به همراه مزایا و معایب رایج‌ترین و پرکابردترین انواع شبکه‌های عصبی، پرداخته می‌شود:

شبکه عصبی پرسپترون

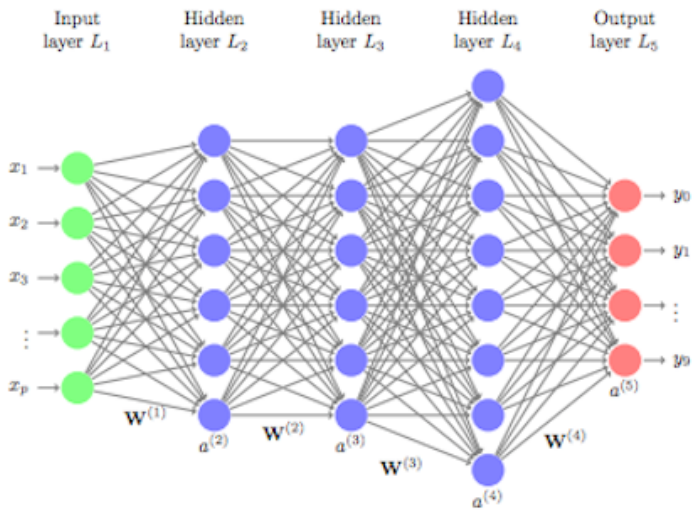
مدل پرسپترون یکی از ساده‌ترین و قدیمی‌ترین مدل‌های شبکه عصبی است که از آن در مسائلی با رویکرد یادگیری نظارت شده برای دسته‌بندی داده‌ها به دو گروه مشخص استفاده می‌شود. مدل پرسپترون تنها دارای دو لایه ورودی و خروجی است. داده‌ها از طریق لایه نخست به شبکه وارد می‌شوند و هر یک از مقادیر، با وزن‌های مدل (که در ابتدای آموزش مدل با مقادیر تصادفی مقداردهی شده‌اند) ضرب می‌شوند. سپس حاصل جمع تمامی ضرب‌ها به لایه آخر منتقل می‌شود. لایه آخر دارای یک گره با تابع فعالسازی است که حد آستانه‌ای برای مقدار ورودی خود در نظر می‌گیرد. چنانچه مقدار ورودی گره بیشتر از عدد ۰ باشد، خروجی تابع فعالسازی برابر با عدد ۱ و در غیر این صورت برابر با عدد ۰ خواهد بود.
مزیت مدل پرسپترون این است که به دلیل سادگی، به انجام محاسبات پیچیده‌ای نیاز ندارد و برای پیاده‌سازی عملیات منطقی نظیر OR AND، و NAND مناسب است.
از معایب اصلی این مدل این است که نمی‌توان از آن برای دسته‌بندی داده‌ها با بیش از دو گروه استفاده کرد و صرفاً این مدل برای دسته‌بندی دودویی استفاده می‌شود. همچنین از این مدل برای تقسیم‌بندی داده‌ها به صورت غیرخطی نمی‌توان استفاده کرد.



شکل ۱.۱: شبکه عصبی پرسپترون [۳]

شبکه عصبی پیش‌خور

شبکه عصبی پیش‌خور از چندین لایه متوالی تشکیل شده است که هر لایه خروجی خود را در قالب بردار، به لایه بعد منتقل می‌کند. بر اساس میزان پیچیدگی مسئله، تعداد لایه‌های پنهان این مدل می‌تواند یک لایه یا بیش از یک لایه باشد.
در این مدل، جریان داده‌ها فقط به‌صورت یک‌طرفه اتفاق می‌افتد. به عبارتی، وزن‌های این مدل، استاتیک هستند و داده‌ها از طریق لایه ورودی، به شبکه وارد می‌شوند و پس از عبور از لایه‌های پنهان و اعمال عملیات محاسباتی بر روی آن‌ها، خروجی نهایی در لایه آخر مشخص می‌شود و دیگر مرحله پس انتشار در این مدل انجام نمی‌شود. مزیت شبکه عصبی پیش‌خور این است که بار محاسباتی کمی دارد و پیاده‌سازی آن به‌سادگی انجام می‌شود. همچنین، به دلیل آن که مرحله پس انتشار در این مدل رخ نمی‌دهد، سرعت اجرای مدل بالا است. مهم‌ترین نقطه ضعف مدل پیش‌خور، عدم

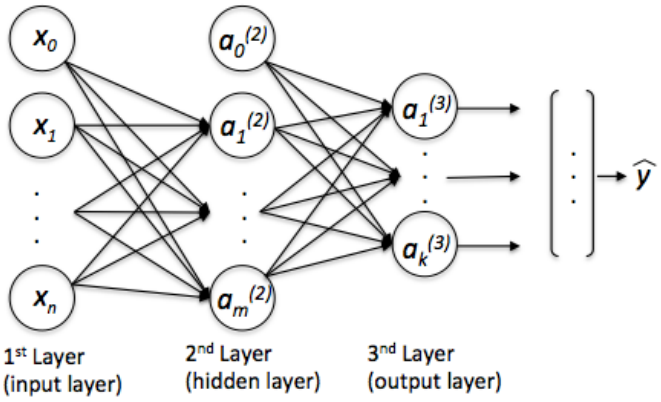


شکل ۲.۱: شبکه عصبی پیش‌خور [۴]

وجود مرحله پس انتشار است که به عنوان یکی از مهم‌ترین مرحله در یادگیری مدل‌های یادگیری عمیق محسوب می‌شود.

شبکه عصبی چند لایه پرسپترون

شبکه عصبی چند لایه پرسپترون، ساختاری مشابه با ساختار مدل پرسپترون دارد؛ اما تعداد لایه‌های پنهان آن بیش از یک لایه است. همچنین، این مدل نوعی شبکه پیش‌خور محسوب می‌شود با این تفاوت که در مدل چند لایه پرسپترون، تعداد تمامی گرههای هر لایه با هم برابر است و ارتباط کاملی بین گرههای هر لایه وجود دارد. خروجی هر لایه در قالب بردار، به عنوان ورودی به لایه بعد منتقل می‌شود. در این مدل، از توابع فعالسازی غیرخطی نظیر Tanh Sigmoid، ReLU، و سایر توابع مشابه استفاده می‌شود. با مدل چند لایه پرسپترون می‌توان عملگرهای منطقی XOR NOT، XNOR، NOR، AND، و OR را پیاده‌سازی کرد.
مزیت مدل چند لایه پرسپترون این است که از این مدل می‌توان برای مدل‌سازی مسائل غیرخطی استفاده کرد. به‌علاوه، این مدل عملکرد خوبی در مسائلی دارد که میزان حجم داده‌های مسئله، کم است.
از معایب این مدل می‌توان به پیچیدگی محاسباتی زیاد و زمان محاسبات بالا اشاره کرد. همچنین در این مدل نمی‌توان به‌راحتی به میزان تاثیر متغیرهای وابسته بر روی متغیرهای مستقل پی برد.



شکل ۳.۱: شبکه عصبی چند لایه پرسپترون [۵]

شبکه عصبی پیچشی

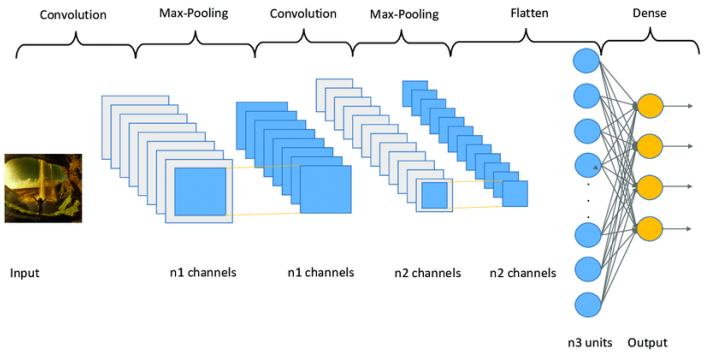
از شبکه عصبی پیچشی برای استخراج ویژگی‌هایی داده ورودی استفاده می‌شود. این شبکه دارای دو بخش اصلی به شرح زیر است:

۱. **لایه پیچشی**: در این لایه، فیلتری به منظور استخراج ویژگی بر روی داده ورودی اعمال می‌شود و سپس نتایج حاصل از این فیلتر، از تابع فعالسازی ReLU عبور کرده تا اعداد

منفی حاصل شده، به عدد صفر نگاشته شوند.

۲. **لایه فشرده‌ساز**: ورودی این لایه، خروجی لایه پیچشی است و از آن به منظور کاهش تعداد پارامترهای شبکه استفاده می‌شود.

مزیت شبکه عصبی پیچشی در این است که از آن می‌توان به منظور استخراج ویژگی با ابعاد پایین استفاده کرد. همچنین، زمانی که از این شبکه به منظور استخراج ویژگی از داده‌ها در مسئله خاصی استفاده می‌شود، در مقایسه با سایر مدل‌ها از دقت بالاتری برخوردار است. به‌علاوه، تعداد پارامترهای این شبکه در مقایسه با سایر شبکه‌ها کم‌تر است. بدین‌ترتیب، این شبکه به محاسبات کم‌تری در حین یادگیری احتیاج دارد.
از معایب اصلی این شبکه این است که برای رسیدن به دقت بالا، به حجم زیادی از داده آموزشی احتیاج دارد. جمع‌آوری و تهیه داده‌های برچسب خورده نیازمند هزینه مالی و هزینه زمانی بالایی است. علاوه‌براین، هرچقدر از تعداد لایه‌های میانی بیشتری در این شبکه استفاده شود، زمان یادگیری شبکه به مراتب بیشتر می‌شود.

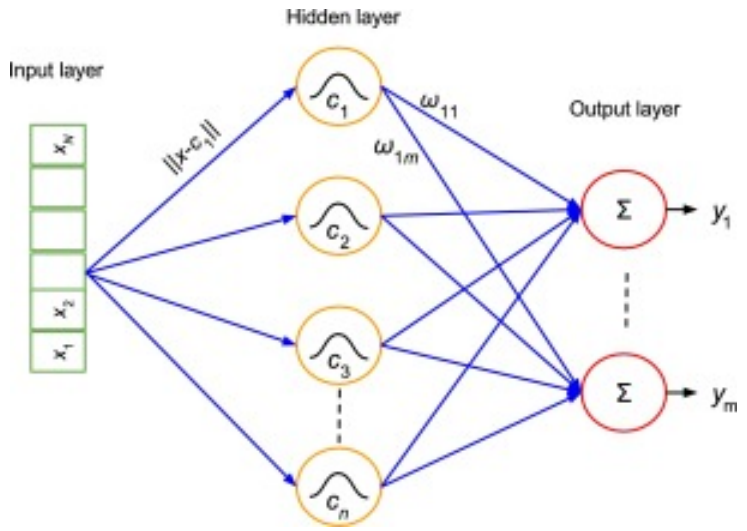


شکل ۴.۱: نمونه ای از شبکه های عصبی پیچشی [۶]

شبکه عصبی تابع پایه شعاعی

ساختار شبکه عصبی تابع شعاعی پایه، مشابه با ساختار شبکه چند لایه پرسپترون است و تنها تفاوتی که با مدل چند لایه پرسپترون دارد، محدودیت در تعداد لایه‌های میانی آن است. تعداد لایه میانی شبکه عصبی تابع شعاعی پایه، یک عدد است و از آن به منظور دسته‌بندی غیرخطی ورودی‌ها استفاده می‌شود.
شبکه عصبی تابع شعاعی پایه به منظور دسته‌بندی داده‌ها، از معیار میزان شباهت داده جدید با مجموعه داده‌های آموزشی استفاده می‌کند. به عبارتی، لایه میانی این شبکه از نورون‌هایی تشکیل شده است که ویژگی‌های داده‌های آموزشی را در خود ذخیره می‌کنند. زمانی که داده جدیدی به مدل وارد می‌شود تا گروه آن مشخص شود، مدل با محاسبه فاصله اقلیدسی داده نسبت به داده‌های آموزشی، نزدیک‌ترین گروه را برای داده مشخص می‌کند.

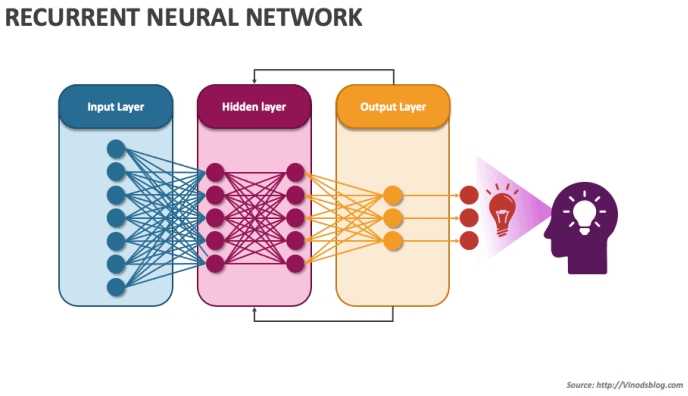
کم بودن تعداد لایه‌های شبکه عصبی تابع شعاعی پایه به عنوان مهم‌ترین مزیت این شبکه محسوب می‌شود که بار محاسباتی یادگیری شبکه را کاهش می‌دهد. افزون‌براین، با اینکه این مدل تنها دارای یک لایه پنهان است، اما می‌توان از آن برای دسته‌بندی داده‌ها با بیش از دو کلاس نیز استفاده کرد. با وجود اینکه یادگیری شبکه عصبی تابع شعاعی پایه به‌سرعت اتفاق می‌افتد و بار محاسباتی مدل کم است، با این حال روند دسته‌بندی داده‌ها با سرعت پایین انجام می‌شود زیرا این مدل برای تشخیص گروه داده جدید، بر پایه سنجش میزان شباهت آن با داده‌های آموزشی خود عمل می‌کند. بدین‌ترتیب، سرعت عملکرد این مدل در مقایسه با سرعت عملکرد شبکه عصبی چند لایه پرسپترون بیشتر است.



شکل ۵.۱: شبکه عصبی تابع پایه شعاعی [۷]

شبکه عصبی بازگشتی

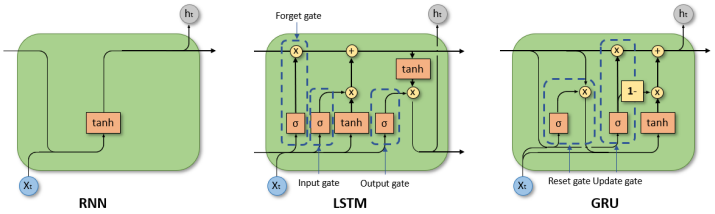
از شبکه عصبی بازگشتی در مسائلی نظیر ترجمه ماشین یا برچسب‌گذاری اجزای کلام استفاده می‌شود که ترتیب در داده‌ها اهمیت داشته باشد. به عنوان مثال، در تشخیص اجزای کلام در جملات، نحوه قرارگیری فاعل، مفعول، فعل و سایر اجزای تشکیل‌دهنده جمله اهمیت دارد. برخلاف سایر مدل‌ها، این نوع شبکه عصبی دارای حافظه‌ای است که اطلاعات داده‌های قبلی خود را ذخیره می‌کند. در شبکه‌های عصبی قبلی، فرض بر این بود که اجزای ورودی مدل هیچ گونه وابستگی به یکدیگر ندارند. اما خروجی مدل بازگشتی، به ورودی‌های قبلی آن وابسته است. مدل‌های بازگشتی دارای انواع دیگری نظیر مدل حافظه بلند کوتاه مدت و مدل واحد بازگشتی گیت هستند که به لحاظ ساختار درونی، تفاوت جزئی با یکدیگر دارند.
مهم‌ترین و اصلی‌ترین مزیت شبکه عصبی بازگشتی، ویژگی بازگشتی بودن آن است که این قابلیت را به مدل می‌دهد تا ترتیب ورودی‌های خود را در حافظه خود نگه دارد و به نوعی، وابستگی داده‌ها را در زمان یادگیری مدنظر قرار دهد. دو مورد از اصلی‌ترین نقاط ضعف شبکه‌های عصبی بازگشتی،



شکل ۱.۶: شبکه عصبی بازگشتی [۸]

محوشدگی گرادیان و انفجار گرادیان است. زمانی مسئله محوشدگی گرادیان اتفاق می‌افتد که تعداد لایه‌های شبکه عصبی زیاد باشد که در پی آن، مقدار گرادیان تابع هزینه به صفر نزدیک‌تر می‌شود. در پی این رخداد، فرآیند یادگیری شبکه عصبی دشوار است. کوچک بودن خروجی مشتق تابع در شبکه‌های عمیق در طی مرحله پس انتشار، باعث می‌شود مقدار گرادیان به‌صورت نمایی کاهش پیدا کند و به عدد صفر نزدیک شود. همین امر سبب می‌شود پارامترهای شبکه عصبی نظیر مقادیر وزن‌ها و بایاس‌ها در لایه‌های ابتدایی شبکه به‌روزرسانی نشوند و بنابراین یادگیری شبکه به‌درستی انجام نگیرد.

انفجار گرادیان نیز زمانی رخ می‌دهد که مقادیر گرادیان خطا روی هم انباشته شوند و بدین‌ترتیب این مقادیر، خیلی بزرگ شوند. این مسئله باعث می‌شود مقدار نهایی وزن‌ها بسیار بزرگ شود که همین امر منجر به ناپایداری شبکه خواهد شد. در برخی مواقع، ممکن است مقادیر بزرگ وزن‌ها باعث سرریز شدن وزن‌ها و رسیدن به مقادیر NaN شوند. در پی انفجار گرادیان‌ها، یادگیری شبکه متوقف شده و وزن‌های شبکه تغییر نخواهند کرد.

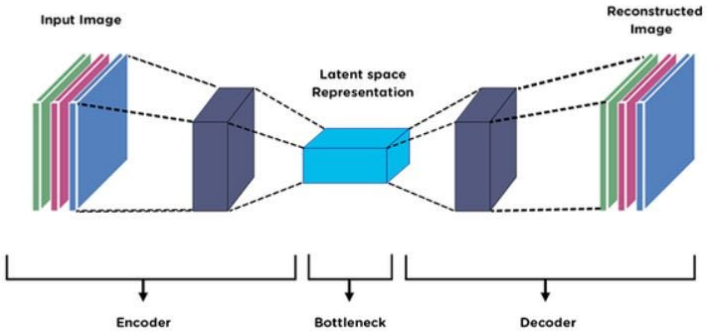


شکل ۱.۷: معماری های شبکه های عصبی بازگشتی [۹]

مدل‌های رمزگذار-رمزگشا

مدل رمزگذار-رمزگشا از دو شبکه عصبی بازگشتی ساخته می‌شود. اولین مدل بازگشتی، داده‌های ورودی را کدگذاری می‌کند و دومین مدل بازگشتی، خروجی مدل بازگشتی نخست را کدگشایی

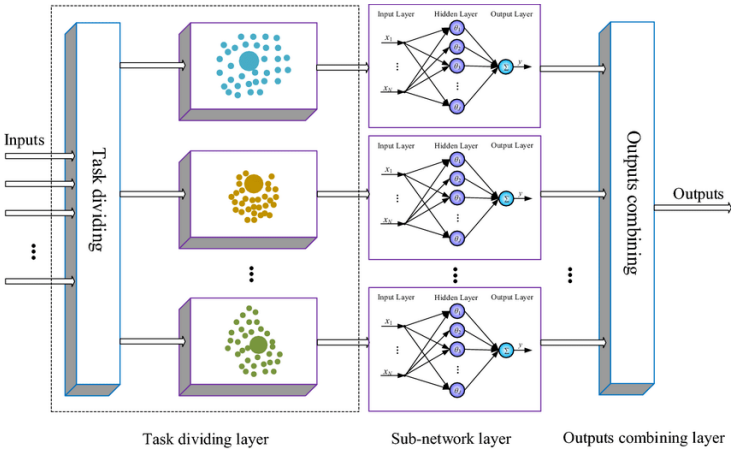
می‌کند. از این مدل برای مسائلی نظیر ترجمه ماشین، سیستم‌های پرسش و پاسخ و چت‌بات استفاده می‌شود که طول ورودی شبکه با طول خروجی شبکه یکسان نیست.
مزیت اصلی مدل رمزگذار_رمزگشا این است که با استفاده از آن‌ها می‌توان خروجی‌هایی تولید کرد که طول آن‌ها وابسته به طول ورودی مدل نباشند. بدین‌ترتیب، این مدل محدودیت مدل‌های بازگشتی را ندارند.
مشکل اصلی مدل رمزگذار_رمزگشا در این است که هر چقدر تعداد لایه‌های این شبکه بیشتر باشد، بار محاسباتی یادگیری شبکه بیشتر می‌شود. همچنین، چنانچه از این مدل در مسئله ترجمه ماشین استفاده شود و ورودی این مدل، از جملات طولانی تشکیل شده باشد، ممکن است مدل با مسئله محوشدگی گرادیان روبه‌رو شود.



شکل ۸.۱: شبکه عصبی رمزگذار_رمزگشا [۱۰]

شبکه عصبی ماژولار

شبکه عصبی ماژولار از چندین شبکه عصبی به عنوان ماژول ساخته می‌شود که هر کدام از این ماژول‌ها بخشی از مسئله را یاد می‌گیرند. در نهایت، خروجی‌های هر یک از این شبکه‌های عصبی با یکدیگر یکپارچه شده تا خروجی نهایی مدل حاصل شود.
از مزیت‌های اصلی مدل‌های ماژولار این است که در این روش، می‌توان یک مسئله پیچیده را به زیر بخش‌های ساده‌تر تقسیم کرد تا به یادگیری مدل کمک شود. همچنین، در این روش می‌توان از شبکه‌های عصبی مختلفی برای ماژول‌ها استفاده کرد. به‌علاوه، در این روش می‌توان یادگیری چندین وظیفه مختلف را به‌طور همزمان پیش برد.
مدل‌های ماژولار عملکرد ضعیفی در حل مسائلی دارند که باید هدف متحرکی را در فضای مشخص شده شناسایی کنند. مسائلی نظیر شناسایی شئ متحرک در فضا، کنترل ترافیک، دنبال کردن هدف در بازی‌های کامپیوتری از این دست مسائل هستند.



شکل ۹.۱: شبکه عصبی ماژولار [۱۱]

۲.۱ پردازش زبان طبیعی

پردازش زبان طبیعی یا NLP، یکی از زیرشاخه‌های مهم در حوزهٔ علوم رایانه، هوش مصنوعی است که به تعامل بین کامپیوتر و زبان‌های (طبیعی) انسانی می‌پردازد؛ بنابراین پردازش زبان‌های طبیعی بر ارتباط انسان و رایانه، متمرکز است. چالش اصلی و عمده در این زمینه، درک زبان طبیعی و ماشینی کردن فرایند درک و برداشت مفاهیم بیان‌شده با یک زبان طبیعیِ انسانی است. به تعریف دقیق‌تر، پردازش زبان‌های طبیعی عبارت است از استفاده از رایانه برای پردازش زبان گفتاری و زبان نوشتاری. بدین معنی که رایانه‌ها را قادر سازیم که گفتار یا نوشتار تولید شده در قالب و ساختار یک زبان طبیعی را تحلیل و درک نموده یا آن را تولید نمایند.
هدف اصلی در پردازش زبان طبیعی، ایجاد تئوری‌هایی محاسباتی از زبان، با استفاده از الگوریتم‌ها و ساختارهای داده‌ای موجود در علوم رایانه است. بدیهی است که در راستای تحقق این هدف، نیاز به دانشی وسیع از زبان است و علاوه بر محققان علوم رایانه، نیاز به دانش زبان شناسان نیز در این حوزه می‌باشد. با پردازش اطلاعات زبانی می‌توان آمار مورد نیاز برای کار با زبان طبیعی را استخراج کرد.

۱.۲.۱ تاریخچه

به‌طور کلی تاریخچه پردازش زبان طبیعی از دهه ۱۹۵۰ میلادی شروع می‌شود. در ۱۹۵۰ آلن تورینگ مقاله معروف خود را دربارهٔ آزمایش تورینگ که امروزه به عنوان ملاک هوشمندی شناخته می‌شود، منتشر ساخت.
نخستین تلاش‌ها برای ترجمه توسط رایانه ناموفق بودند، به‌طوری‌که ناامیدی بنگاه‌های تأمین بودجهٔ پژوهش از این حوزه را نیز در پی داشتند. پس از اولین تلاش‌ها آشکار شد که پیچیدگی زبان بسیار بیشتر از چیزی است که پژوهشگران در ابتدا می پنداشتند. بنابراین حوزه‌ای که پس از آن برای استعانت مورد توجه قرار گرفت زبان‌شناسی بود؛ اما در آن دوران نظریهٔ زبان‌شناسی وجود نداشت

که بتواند کمک شایانی به پردازش زبان‌ها بکند. در سال ۱۹۵۷ کتاب ساختارهای نحوی اثر نوام چامسکی زبان‌شناس جوان آمریکایی که از آن پس به شناخته‌شده‌ترین چهرهٔ زبان‌شناسی نظری تبدیل شد به چاپ رسید. از آن پس پردازش زبان با حرکت‌های تازه‌ای دنبال شد اما هرگز قادر به حل کلی مسئله نشد.

پردازش زبان طبیعی مبتنی بر قوانین دست‌نویس (دهه ۱۹۵۰ - اوایل دهه ۱۹۹۰)

دهه ۱۹۵۰: آزمایش جورج تاون در سال ۱۹۵۴ شامل ترجمه کاملاً خودکار بیش از شصت جمله روسی به انگلیسی بود. نویسندگان ادعا کردند که ظرف سه یا پنج سال، ترجمه ماشینی یک مشکل حل شده خواهد بود. با این حال، پیشرفت واقعی بسیار کندتر بود، و پس از گزارش ALPAC در سال ۱۹۶۶، که نشان می‌داد تحقیقات ده ساله نتوانسته‌است انتظارات را برآورده کند، بودجه برای ترجمه ماشینی به‌طور چشمگیری کاهش یافت. تحقیقات کمی در مورد ترجمه ماشینی در آمریکا انجام شد (اگرچه برخی از تحقیقات در جاهای دیگر مانند ژاپن و اروپا) تا اواخر دهه ۱۹۸۰ که اولین سیستم‌های ترجمه ماشینی آماری توسعه یافتند، ادامه یافت.
دهه ۱۹۶۰: برخی از سیستم‌های پردازش زبان طبیعی موفق که در دهه ۱۹۶۰ توسعه یافتند عبارت بودند از ،SHRDLU یک سیستم زبان طبیعی که در جهان‌های بلوکی محدود با واژگان محدود کار می‌کرد، و ،ELIZA شبیه‌سازی یک روان‌درمانگر بود، که توسط جوزف وایزنبام بین سال‌های ۱۹۶۴ و ۱۹۶۶ نوشته شده بود. الایزا با استفاده از تقریباً هیچ اطلاعاتی در مورد افکار یا احساسات انسان، گاهی تعامل شگفت‌انگیزی شبیه انسان ارائه می‌داد. ولی وقتی "بیمار" از پایگاه دانش بسیار کوچک فراتر می‌رفت، ELIZA ممکن بود یک پاسخ عمومی ارائه دهد، برای مثال، به "سرم درد می‌کند" با "چرا می‌گویی سرت درد می‌کند؟" پاسخ دهد.
دهه ۱۹۷۰: در طول دهه ۱۹۷۰، بسیاری از برنامه نویسان شروع به نوشتن هستی‌شناسی‌های مفهومی کردند، که اطلاعات دنیای واقعی را به داده‌های قابل فهم کامپیوتری ساختار می‌داد. در طول این مدت، اولین ربات‌های گفتگو (به عنوان مثال، PARRY) نوشته شدند.
دهه ۱۹۸۰: دهه ۱۹۸۰ و اوایل دهه ۱۹۹۰ دوران اوج روش‌های دست‌نویس در حوزه پردازش زبان طبیعی است. حوزه‌های مورد توجه در آن زمان شامل تحقیق در مورد تجزیه مبتنی بر قاعده (مانند توسعه HPSG به عنوان عملیاتی محاسباتی گرامر مولد)، مورفولوژی (مانند مورفولوژی دو سطحی)، معناشناسی (مانند الگوریتم Lesk) بودند.

پردازش زبان طبیعی مبتنی بر روشهای آماری (۱۹۹۰–۲۰۱۰)

تا دهه ۱۹۸۰، اکثر سیستم‌های پردازش زبان طبیعی بر اساس مجموعه‌های پیچیده‌ای از قوانین دست نوشته بودند. از اواخر دهه ۱۹۸۰، با معرفی الگوریتم‌های یادگیری ماشینی برای پردازش زبان، انقلابی در پردازش زبان طبیعی رخ داد. این امر هم به دلیل افزایش مداوم در قدرت محاسباتی و هم کاهش تدریجی تسلط نظریات زبان‌شناسی چامسکی بود.
دهه ۱۹۹۰: بسیاری از موفقیت‌های اولیه قابل توجه در روش‌های آماری در NLP در زمینه ترجمه ماشینی رخ داد، به‌ویژه به دلیل تحقیقات ،IBM مانند مدل‌های هم‌ترازی .IBM این سیستم‌ها می‌توانستند از مجموعه‌های متنی چندزبانه موجود که توسط پارلمان کانادا و اتحادیه اروپا تهیه شده

بود استفاده کنند. این مجموعه‌های متنی در نتیجه قوانینی که خواستار ترجمه تمام اقدامات دولتی به همه زبان‌های رسمی نظام‌های دولتی مربوطه بودند، تهیه شده بود. با این حال، بیشتر سیستم‌ها به مجموعه‌هایی وابسته بودند که به‌طور خاص برای وظایف پیاده‌سازی شده توسط این سیستم‌ها توسعه یافته بودند، که یک محدودیت عمده در موفقیت این سیستم‌ها بود. در نتیجه، تحقیقات زیادی روی روش‌های یادگیری مؤثرتر از مقادیر محدود داده انجام شد.

دهه ۲۰۰۰: با رشد وب، از اواسط دهه ۱۹۹۰، مقدار فزاینده ای از داده‌های خام در دسترس قرار گرفت؛ بنابراین تحقیقات به‌طور فزاینده ای بر روی الگوریتم‌های یادگیری بدون نظارت و نیمه نظارتی متمرکز شد. چنین الگوریتم‌هایی می‌توانستند از داده‌هایی که به صورت دستی با پاسخ‌های مورد نظر یا با استفاده از ترکیبی از داده‌های بدون برچسب بیاموزند. به‌طور کلی، این کار بسیار دشوارتر از یادگیری تحت نظارت است و معمولاً نتایج دقیق کمتری را برای مقدار معینی از داده‌های ورودی ایجاد می‌کند. با این حال، حجم عظیمی از داده‌های بدون برچسب در دسترس است (از جمله، کل محتوای شبکه جهانی وب)، که اگر الگوریتم مورد استفاده پیچیدگی زمانی کافی داشته باشد، اغلب می‌تواند نتایج ضعیف‌تر را جبران کند.

پردازش زبان طبیعی مبتنی بر شبکه‌های عصبی (در حال حاضر)

در دهه ۲۰۱۰، روش‌های یادگیری بازنمایی و یادگیری ماشینی به سبک شبکه عصبی عمیق در پردازش زبان طبیعی رایج شد. این محبوبیت تا حدی به دلیل انبوهی از نتایج بود که نشان می‌داد چنین تکنیک‌هایی می‌توانند به نتایج پیشرفته‌ای در بسیاری از کارهای زبان طبیعی، مانند مدل‌سازی زبان و تجزیه دست یابند. این امر به‌طور فزاینده ای در پزشکی و مراقبت‌های بهداشتی مهم است.

۲.۲.۱ نحوه عملکرد

به‌طور کلی، ۴ گام اصلی روش‌های پردازش زبان طبیعی شامل ۴ عبارت اند از:

۱. **تحلیل لغوی**: این مرحله که تحلیل نام دارد، شامل فرایندی است که طی آن یک جمله، به کلمات یا واحدهای کوچکی به نام «نشانه‌ها» شکسته می‌شود تا معنای آن و رابطه‌اش با کل جمله تشخیص داده شود.

۲. **تحلیل نحوی**: مرحله تحلیل نحوی، به فرایندی اشاره دارد که در آن، مورادی که در ادامه بیان شده، صورت می‌گیرد.

ارتباط بین عبارات و کلمات گوناگون درون جمله، تشخیص داده می‌شود. ساختار این کلمات استانداردسازی می‌شود. روابط به‌صورت ساختار سلسله‌مراتبی بیان می‌شود.

۳. **تحلیل معنایی**: مرحله تحلیل معنایی، فرایندی است که ساختارهای نحوی را به معانی مستقل از زبانشان مرتبط می‌سازد و این کار از سطوح عبارات و بندها (بخشی از جملات)، جملات و پاراگراف‌ها تا مرحله کلی نوشتار صورت می‌گیرد.

۴. **تبدیل خروجی**: گام تبدیل خروجی، فرایندی است که در آن، نتیجه‌ای (خروجی) بر مبنای تحلیل معنایی متن یا گفتار، تولید می‌شود که متناسب با هدف اپلیکیشن است.

۳.۲.۱ ابزار و رویکرد

برای بهره برداری از مزایای مفاهیم موجود در حوزه NLP بایستی از ابزار و رویکردهایی خاص استفاده کرد که در زیر شرح داده شده اند.

ابزار زبان طبیعی

زبان برنامه نویسی پایتون طیف وسیعی از ابزارها و کتابخانه‌ها را برای به کارگیری در وظایف خاص پردازش زبان طبیعی فراهم می‌کند. بسیاری از این موارد در کتابخانه ابزار زبان طبیعی یا NLTK، مجموعه ای منبع باز از کتابخانه ها، برنامه‌ها و منابع آموزشی برای ساخت برنامه‌های NLP پیدا می‌شوند. NLTK شامل کتابخانه‌هایی برای بسیاری از وظایف این حوزه و نیز کتابخانه‌هایی برای وظایف فرعی، مانند تجزیه جملات، تقسیم‌بندی کلمات، ریشه‌یابی و ریشه یابی و نشانه سازی است. پایتون همچنین شامل کتابخانه‌هایی برای پیاده‌سازی قابلیت‌هایی مانند استدلال معنایی، توانایی رسیدن به نتایج منطقی بر اساس حقایق استخراج‌شده از متن است.

رویکردهای آماری، یادگیری ماشین و یادگیری عمیق

اولین برنامه‌های پردازش طبیعی متن، سیستم‌های مبتنی بر قواعد و کدگذاری دستی بودند که می‌توانستند وظایف NLP خاصی را انجام دهند، اما نمی‌توانستند به راحتی مقیاس‌پذیر شوند تا جریان به ظاهر بی‌پایانی از استثناها یا حجم فزاینده متن را در خود جای دهند. NLP آماری الگوریتم‌های کامپیوتری را با مدل‌های یادگیری ماشین و یادگیری عمیق ترکیب می‌کند تا به طور خودکار عناصر متن را استخراج، طبقه‌بندی و برچسب‌گذاری کند و سپس احتمال آماری را به هر معنای احتمالی آن عناصر اختصاص دهد. امروزه، مدل‌های یادگیری عمیق و تکنیک‌های یادگیری مبتنی بر شبکه‌های عصبی پیچشی و شبکه‌های عصبی بازگشتی سیستم‌های NLP را قادر می‌سازند که در حین کار یاد بگیرند و معنای دقیق‌تری را از حجم عظیمی از متن خام، بدون ساختار و بدون برچسب استخراج کنند.

۴.۲.۱ کاربرد

پردازش زبان طبیعی، این امکان را برای کامپیوترها فراهم می‌کند تا گفتار انسان‌ها را بفهمند و نمونه‌ای از آن را تولید کنند. به‌همین دلیل موارد استفاده زیادی دارند. کاربردهای پردازش زبان طبیعی به دو دسته کلی قابل تقسیم است:

۱. **کاربردهای نوشتاری**: از کاربردهای نوشتاری آن می‌توان به استخراج اطلاعاتی خاص از یک متن، ترجمه یک متن به زبانی دیگر یا یافتن مستنداتی خاص در یک پایگاه داده نوشتاری (مثلاً یافتن کتاب‌های مرتبط به هم در یک کتابخانه) اشاره کرد.

۲. **کاربردهای گفتاری**: نمونه‌هایی از کاربردهای گفتاری پردازش زبان عبارتند از سیستم‌های پرسش و پاسخ انسان با رایانه، سرویس‌های اتوماتیک ارتباط با مشتری از طریق تلفن،

سیستم‌های آموزش به فراگیران یا سیستم‌های کنترلی توسط صدا که در سالهای اخیر این حوزه تحقیقاتی توجه دانشمندان را به خود جلب کرده‌است و تحقیقات قابل ملاحظه‌ای در این زمینه صورت گرفته است.

۵.۲.۱ مزایا و معایب NLP

در ادامه برخی از مزیت‌ها و عیب‌هایی که می‌توان برای پردازش زبان طبیعی بیان کرد را آورده‌ایم.

مزایا

پردازش زبان طبیعی این امکان را می‌دهد تا داده‌ها را از منابع ساختارمند و بدون ساختار تحلیل کنیم. این عمل بسیار سریع و از نظر زمانی مقرون به‌صرفه و کارآمد است. پردازش زبان طبیعی، پاسخ‌های دقیق و جامعی را برای پرسش‌ها فراهم می‌کند. به‌همین دلیل، در زمان زیادی که صرف اطلاعات ناخواسته و بی مورد می‌شود صرفه‌جویی می‌کند. NLP به کاربران امکان می‌دهد تا سوالاتی را در موضوع دلخواه و گوناگون بپرسند و در کسری از ثانیه پاسخ خود را دریافت کنند.

معایب

فرایند آموزش مدل‌های پردازش زبان طبیعی نیازمند داده و محاسبات بسیار زیادی است. NLP هنگام رویارویی با عبارات غیررسمی، کنایه و اصطلاحات، دچار مشکلات متعددی می‌شود. گاهی نتایج حاصل از پردازش زبان طبیعی به اندازه کافی دقیق نیست. به عبارت دیگر می‌توان گفت که دقت آن به‌طور مستقیم با دقت داده‌ها تناسب دارد. NLP برای وظایف مشخص و محدودی طراحی شده است، بنابراین نمی‌تواند با حوزه‌های جدید سازگار شود و عملکرد محدودی را در این مورد ارائه می‌دهد.

۶.۲.۱ چالش ها

پرداختن به زبان‌طبیعی، موضوع بسیار دشواری است. حتی به‌عنوان یک انسان، گاهی اوقات ممکن است در تفسیر جملات یکدیگر یا تصحیح اشتباه تایپی متن خود به مشکل بر بخوریم. پردازش زبان طبیعی در مسیر خود با چالش‌های بسیاری همراه است که کاربردهایش را در معرض اشتباه و عدم موفقیت قرار می‌دهد. برخی از چالش‌های اساسی عبارت اند از:

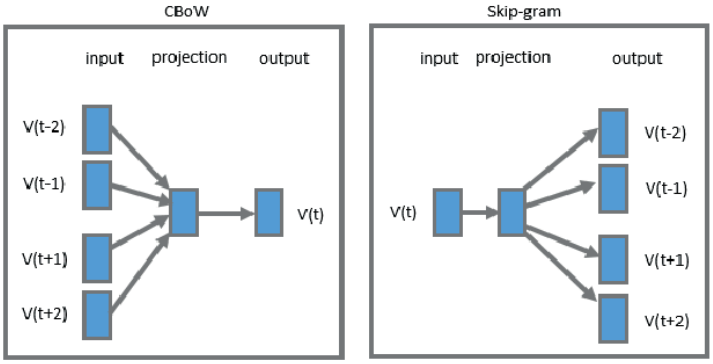
۱. طعنه

۲. تعدد معنایی یا مبهم بودن عبارت

۳. ادبیات عامیانه یا کوچه بازاری

۴. ادبیات مربوط به حوزه خاص

۵. سوگیری موجود در داده‌های آموزشی



شکل ۱۰.۱: CBOW vs. Skip-gram [۱۳]

هرچند که امروزه، با پیشرفت‌های صورت گرفته در زمینه فهم زبان طبیعی به‌عنوان یکی از شاخه‌های هوش مصنوعی، یادگیری عمیق و داده‌های آموزشی جامعه، دریچه‌ای برای الگوریتم‌ها ایجاد شده است تا گفتار و متن واقعی را مشاهده از آن یاد بگیرند. بدین ترتیب به چالش‌های این حوزه رسیدگی شده است.

۷.۲.۱ استخراج ویژگی

این مرحله در پردازش زبان طبیعی با استفاده از الگوریتم های یادگیری ماشین الزامی است. در این فرایند متن خود را به بردارهایی از اعداد تبدیل کرده که آماده پردازش توسط مدل های طراحی شده می شوند. الگوریتم های مختلفی برای استخراج ویژگی ها مورد استفاده قرار می گیرد که برخی از مشهورترین آنان به شرح زیر هستند.

Word۲Vec

مدل Word۲Vec [۱۲] از زمان معرفی در سال ۲۰۱۳ تاکنون به محبوبیت بالایی در کاربردهای مختلف پردازش زبان طبیعی از جمله تحلیل احساسات دست پیدا کرده است. این الگوریتم که مورد استفاده Skip-Gram و CBOW[۲] به طور معمول با دو معماری قرار میگیرد. از شبکه های عصبی برای تبدیل متن به بردارهای عددی بهره می گیرد.

GloVe

مدل GloVe [۱۴] از ترکیب دو روش تجزیه ماتریس سراسری و روش Skip-gram استفاده می کند. در این مدل از تعداد تکرار کلمات در هر متن برای یافتن کلمات هم معنی استفاده شده به این صورت که به جای استفاده از احتمال رخداد برای رسیدن به معنای کلمه از نسبت احتمالات هم رخدادی استفاده کنیم. Glove برخلاف CBoW و Skip-gram به جای استفاده از آنتروپی

[۲] Continues Bag-of-Words

متقاطع از مجموع حداقل مربعات وزن دار شده برای پیش بینی استفاده میکند. مدل Glove به ازای استفاده از مرجع آموزش تعداد کلمات و زمان یادگیری یکسان عملکرد بهتری را نسبت به Word۲Vec از خود نشان می دهد.

FastText

مدل FastText (ارائه توسط شرکت متا[۳] در ۲۰۱۶) منبع باز نیز بر پایه شبکه های عصبی ساخته شده و توسط کمپانی فیس بوک در جهت بهبود مدل Word۲Vec منتشر شد و از معماری مشابه Skip-gram استفاده می کند. این مدل هر یک از کلمات را نیز به بخش های کوچک تری تبدیل کرده و به کمک این روش اطلاعات بیشتری را از هر کلمه استخراج میکند. این الگوریتم برای درک کلمات کمیاب کلمات دارای ایراد نگارشی با کلماتی که در زبان طبیعی شناسایی نشده اند [۴] کمک فراوانی می کند.

ELMo

نوعی دیگر از مدلها برای تعبیه سازی کلمات، الگوریتم ELMo [۱۵] می باشد که بر پایه شبکه های عصبی ساخته شده است. این مدل با بهره گیری از معماری حافظه کوتاه و بلند مدت دولایه جملات را از راست و چپ مورد بررسی قرار میدهد تا نقش کلمات بعد و قبل کلمه مورد نظر را بررسی کند. این مدل کلمه موجود در متن های مختلف برداری را به آن اختصاص داده و مشکل وابستگی معنی برای هر کلمه به متن را بر طرف میکند.

---

[۳] Meta

[۴] Out Of Vocabulary (OOV)

فصل ۲

پیشینه پژوهش

مسئله تشخیص نویسنده در زبان انگلیسی بسیار مورد توجه و مدلسازی قرار گرفته است. در مقاله [۱۶]، مدلی بر پایه مدل زبانی بزرگ برای تشخیص نویسنده به نام مدل زبانی نویسندگی ارائه شد که روی هر دو مجموعه داده Blogs۵۰ [۱۷] و CCAT۵۰ [۱۸] نسبت به برخی دیگر از مدل های مورد بررسی عملکرد مطلوبی را ارائه داد.
مقاله [۱۹] بر تقویت استحکام و کارایی مدلسازی در راستای حل این مسئله هنگامی که نویسندگانی موجود باشند که در حوزه های مختلف، می نگارند، با استفاده از رویکردهای تحریف متن[۱] تاکید دارد؛ تاکیدی که از مدلسازی و آزمایش روی دو مجموعه داده عظیم CCAT-۱۰ (زیرمجموعه ای از مجموعه متنی رویترز[۲]) [۲۰] و Guardian [۲۱] از روزنامه بریتانیایی به همین اسم بدست آمده است.
در مقاله [۲۲] نشان داده شده است که هر مدل زبانی مجهز به رویکرد Linguistically Informed Prompting یا [LIP در حل این مسئله به خوبی کمک می کند.
در مقاله [۲۳] مقایسه ای بین عملکرد و قدرت دو مدل BERT و DistilBERT [۲۴] برای این مسئله صورت گرفت و برتر بودن مدل DistilBERT از نظر دقت و کارایی در مدل سازی گزارش شد.
در مقاله [۲۵] نشان داده شده است که در صورت وجود یک مدل کارآمد، میزان کارایی با افزایش تعداد نویسندگان در مجموعه دادگان، کاهش می یابد.
مقاله [۲۶] نشان داده است که رویکرد مبتنی بر شباهت ساده می تواند پیچیده ترین مسائل تشخیص نویسنده را نیز حل کند.
مقاله [۲۷] یک مرور اساسی روی روش های مختلف حل این مسئله را در میان می گذار. یک نتیجه مهم که تکمیل کننده نتیجه مقاله سایز [۲۵] است، آن است که کاهش دقت با افزایش تعداد نویسندگان برای مدل های سنتی حوزه یادگیری ماشین مانند مدل ماشین بردار پشتیبان، شدیدتر از این کاهش برای مدل های عمیق مانند مدل حافظه طولانی کوتاه مدت یک سویه [۲۸] و یا دو سویه است.
پژوهش در راستای مدلسازی حل این مسئله در زبان های غیر انگلیسی نیز انجام شده است. از

[۱] در اینجا منظور از تحریف، حذف کلمات مرتبط با موضوع نوشته اما مستقل از شیوه نگارش نویسنده است
[۲] Reuters Corpus

مقاله [۲۹] نتیجه می شود که دسته بند های بیز ساده مبتنی بر مدل های ۵ الی ۷ گرمی، مطلوب ترین نتیجه و بالاترین دقت را برای حل این مسئله در هر گویش آلمانی که در بخش آلمانی زبان سوئیس صحبت می شود، در میان مدل‌های n-گرمی مورد بررسی، بدست می دهد.
در مقاله [۳۱]، این مسئله برای سه زبان انگلیسی، فرانسوی و آلمانی در سه رویکرد تحلیل مولفه های اصلی [۳۰]، قانون دلتا (ارائه شده توسط Burrows در ۲۰۰۲) و یک دسته بند مبتنی بر Z-scores (ارائه شده توسط Muller در ۱۹۹۲) مدلسازی شده و برتری استفاده از رویکرد سوم نسبت به دو رویکرد اول، گزارش شد.
همچنین یک رویکرد مبتنی بر گراف در مقاله [۳۲] به جهت حل این مسئله در چهار زبان انگلیسی، اسپانیایی، هلندی و یونانی ارائه و تحلیل شده است.

فصل ۳

ابزار مدلسازی

۱.۳ مجموعه دادگان

مجموعه دادگان مورد بررسی، مجموعه ای متشکل از ۳۰ عدد نوشته حدود ۵۰۰ کلمه ای به زبان فارسی به ازای هر نویسنده است. این نوشته ها از فضای اینترنت استخراج شده و برابری تقریبی تعداد کلمات آثار از هر نویسنده به منظور ایجاد یک تعادل در مجوعه دادگان آموزشی و جلوگیری از متعصب شدن مدل نسبت به یک حوزه صحبت یا یک نویسنده است. نام خانوادگی هر یک از این نویسندگان به همراه حوزه صحبت به شرح زیر است:

جدول ۱.۳: مجموعه دادگان

نام خانوادگی نویسنده	حوزه صحبت
دهقان	خودروها
شتری	خودروها
متین	سینما
راستی کردار	فونت های زبان فارسی
دهقانی	مدل های زبانی
حصارکی	هوش مصنوعی و یادگیری ماشین
بهرامی جو	بازی های رایانه ای
حاجی محمدی	سینما
شیخی	فناوری
زاهدی	بازی های رایانه ای

گزینش متون از حوزه های مختلف به منظور افزایش تنوع و پیچیده تر کردن فرآیند مدل سازی است. همچنین برخی از کلمات موجود در مجموعه دادگان به زبان انگلیسی است تا بر تنوع دادگان و پیچیدگی آنان افزوده شود. به منظور پیش پردازش دادگان مورد مطالعه از ابزارهای کتابخانه 'hazm' [۳۳] در زبان برنامه نویسی پایتون و نیز حذف کننده هایی استفاده شده است.

۲.۳ مدل BERT

۱.۲.۳ سرآغاز شکل‌گیری

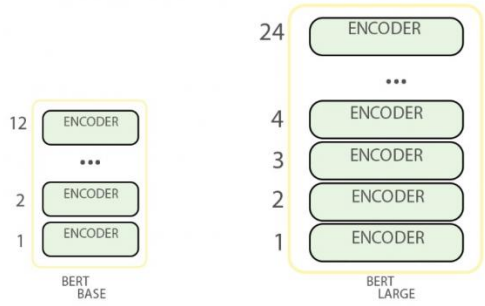
هنگامی که برای اولین بار یک شبکه عصبی پیچشی توانست در مسابقه ImageNet [۳۴] برنده شود، نگاه‌ها به‌سمت یادگیری ماشین و یادگیری عمیق جلب شد. به‌طوری که بسیاری از شرکت‌ها به دنبال حل مشکلات با استفاده از الگوریتم‌های هوشمند بودند، در حالی که غافل از این نکته بودند که شبکه عصبی پیچشی شرکت‌کننده در این مسابقه بر مبنای حجم گسترده‌ای از داده‌ها آموزش داده شده بود؛ درصورتی‌که برای حل بسیاری از مشکلات، کاربران به چنین حجمی از اطلاعات دسترسی ندارند. در سویی دیگر، آموزش یک شبکه عمیق با داده‌های زیاد کاری نیست که همه شرکت‌ها توانایی انجام آن‌را داشته باشند. زیرا نیازمند حجم زیادی از داده‌ها و سامانه‌هایی است که توانایی پردازش اطلاعات را داشته باشند. درست در همین نقطه بود که استفاده از مدل‌های از قبل‌آموزش‌دیده به یاری کسب‌وکارها و افرادی آمد که داده و توان پردازشی محدودی در اختیار داشتند. در چنین شرایطی کافی است یک شبکه را با استفاده از ویژگی‌های مختلفی مثل استخراج ویژگی و تنظیم دقیق و غیره برای کار خودمان آماده کرده و به‌صورت اختصاصی آموزش دهیم.
مدل BERT[۱] یک مدل زبانی عمیق مبتنی بر معماری ترنسفورمر است که در سال ۲۰۱۸ توسط تیم هوش مصنوعی گوگل توسعه داده شد. این مدل قادر است برای یک واژه یا جمله، یک بردار ویژگی با اندازه ثابت تولید کند که قابل استفاده در وظایف پردازش زبان طبیعی، مانند تشخیص احساسات، ترجمه ماشینی و پاسخ به سوالات باشد. مهم‌ترین ویژگی مدل BERT این است که یک مدل زبانی دوطرفه است، به این معنی که برای پیش‌بینی هر کلمه در جمله، به تمام کلمات قبل و بعد از آن مراجعه می‌کند و از اطلاعات موجود در آن‌ها استفاده می‌کند. این ویژگی باعث می‌شود که مدل BERT برای تمام وظایف پردازش زبان طبیعی، از جمله تشخیص احساسات، ترجمه ماشینی و پاسخ به پرسش‌ها کارآمد باشد.

۲.۲.۳ عملکرد و معماری

این مدل در واقع ترکیبی از دو رویکرد مختلف به‌نام‌های مدل زبانی آموزش بدون نظارت و یادگیری چند وظیفگی همزمان است.
در رویکرد مدل زبانی آموزش بدون نظارت، شبکه با استفاده از کلان‌داده‌ها و بدون برچسب‌گذاری داده‌ها، روند یادگیری زبان طبیعی را آغاز می‌کند. در این مرحله، شبکه می‌تواند بدون ناظر یاد بگیرد که هر کلمه در یک جمله چه معنایی دارد و چگونه با کلمات دیگر در جمله ارتباط دارد. در رویکرد یادگیری همزمان، شبکه به‌صورت همزمان برای چند وظیفه مختلف آموزش داده می‌شود که از آن جمله باید به تشخیص ترتیب جملات، پرسش و پاسخ، تشخیص نوع جمله و تشخیص موجودیت‌ها اشاره کرد. با این کار، شبکه می‌تواند بهترین ویژگی‌های مربوط به هر کدام از این وظایف را یاد بگیرد و این ویژگی‌ها را با هم ترکیب کند تا بتواند وظایف دیگری را انجام دهد.
مدل BERT با ترکیب این دو رویکرد و در اختیار داشتن حجم گسترده‌ای از داده‌ها و توانایی انجام چند کار به‌طور همزمان قادر است به دقیق‌ترین شکل ممکن وظایف محوله را انجام دهد. همین

[۱] Bidirectional Encoder Representations from Transformers

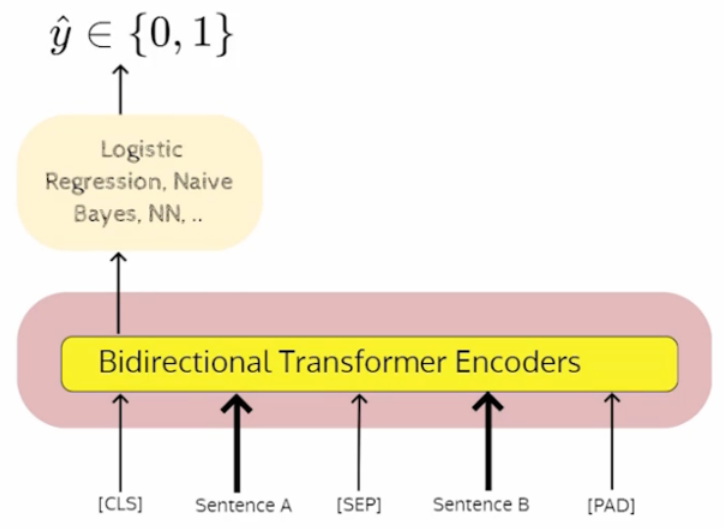
مسئله باعث شده تا BERT به یکی از بهترین و قدرتمندترین مدل‌های پردازش زبان طبیعی تبدیل شود.
در حقیقت این مدل در دو اندازه متفاوت آموزش داده می‌شود که BERT پایه و BERT بزرگ نام دارند.
مدل BERT در واقع دسته‌ای از رمزگذارهای مدل ترنسفورمر است که آموزش دیده‌اند. هر دو مدل BERT شامل لایه‌های رمزگذار است. به‌طور مثال، مدل BERT پایه ۱۲ لایه رمزگذار و مدل بزرگ‌تر ۲۴ لایه رمزگذار دارد. مدل پایه در مجموع ۱۱۰ میلیون پارامتر و مدل بزرگ ۳۴۵ میلیون پارامتر دارد که آموزش هر یک از آن‌ها به چهار روز زمان نیاز دارد (به‌شرطی که تجهیزات سخت‌افزاری قدرتمندی موجود باشند). مدل پایه ۷۶۸ و مدل بزرگ‌تر ۱۰۲۴ گره پنهان در لایه شبکه پیشخور خود دارند و تعداد لایه‌های توجه در اولی ۱۲ و در دومی ۱۶ عدد است (شکل ۱.۲). در مدل فوق، اولین نشانه ورودی با یک نشانه خاص که CLS نام دارد در اختیار مدل قرار می‌گیرد که شباهت زیادی به رمزگذار معماری ترنسفورمر دارد. به بیان دقیق‌تر، دنباله‌ای از کلمات به‌عنوان ورودی در اختیار مدل قرار می‌گیرند. این‌ها در طول لایه‌های رمزگذار حرکت می‌کنند. هر لایه رمزگذار یک لایه خودتوجه و یک لایه شبکه پیشخور دارد که ورودی‌ها از آن‌ها عبور می‌کنند و سپس به لایه رمزگذار بعدی وارد می‌شوند. هر بردار موقعیت، گره‌های لایه پنهان را در خروجی نشان می‌دهد. به‌طور مثال، در مدل BERT پایه اندازه لایه پنهان ۷۶۸ است، بنابراین در خروجی بردارهایی به اندازه ۷۶۸ خواهیم داشت. در مسئله طبقه‌بندی فقط بردار خروجی اول مهم است که ورودی آن همان توکن CLS است. این بردار خروجی در مسئله طبقه‌بندی به‌عنوان ورودی به لایه طبقه‌بندی وارد می‌شود تا بتواند در خروجی نتیجه را نشان دهد.



شکل ۱.۳: BERTLARGE vs. BERTBASE [۳۵]

۳.۲.۳ آموزش

مدل BERT بر مبنای رویکرد بدون ناظر و یادگیری انتقالی آموزش داده می‌شود. در این رویکرد، مدل با استفاده از مجموعه داده بزرگی که از منابع مختلفی مثل وب جمع‌آوری شده است، بدون نیاز به برچسب‌های دقیق آموزش می‌بیند. در مرحله اول، برای آموزش مدل، هر جمله به چند قطعه کوچک‌تر تقسیم می‌شود. سپس، برای هر قطعه، یک بردار ویژگی ایجاد می‌شود که شامل تمام کلمات و موقعیت آن‌ها در جمله است. در مرحله بعد، با استفاده از شبکه‌های ترنسفورمر برای هر ویژگی، بردار جدیدی به‌نام مدل زبانی نقاب‌دار ایجاد می‌شود. در این مرحله، برای برخی از کلمات



شکل ۲.۳: معماری مدل BERT برای عمل دسته بندی دودویی [۳۶]

در هر پاراگراف از مترادف‌ها یا نمادهای ویژه استفاده می‌شود و سپس مدل سعی می‌کند که با توجه به کلمات دیگر در همان جمله، کلمات جایگزین را پیش‌بینی کند. در مرحله آخر، با استفاده از رویکرد مدل زبانی نقاب‌دار، مدل برای تمام جملات و نوشتارهای جدید، بردار ویژگی جدیدی ایجاد می‌کند. سپس، با استفاده از این بردارهای ویژگی، مدل با محوریت رویکرد یادگیری انتقالی، برای وظایف دیگری مانند تشخیص موجودیت‌ها و پرسش و پاسخ آموزش داده می‌شود.

۴.۲.۳ تنظیم دقیق

تنظیم دقیق در مدل BERT، به معنی آموزش مجدد این مدل برای انجام وظیفه‌ای خاص با استفاده از داده‌های برچسب‌دار است. در این روش، ابتدا مدل بدون برچسب آموزش داده می‌شود و در قسمت تنظیم دقیق با استفاده از داده‌های برچسب‌دار، برای انجام وظیفه‌ای خاص دوباره آموزش داده می‌شود. وظایف خاص می‌توانند تشخیص احساسات، تشخیص موجودیت‌ها یا پرسش و پاسخ و تعامل با کاربران باشند. در ادامه، برای آموزش دوباره مدل از الگوریتم‌های بهینه‌سازی مانند Adam استفاده می‌شود و پارامترهای مدل با استفاده از داده‌های برچسب‌دار به‌روزرسانی می‌شوند. همچنین، برای جلوگیری از برازش بیش‌ازحد از تکنیک‌هایی مانند Dropout و L۲ regularization استفاده می‌شود.

۵.۲.۳ استخراج ویژگی

استخراج ویژگی در مدل BERT، به معنی استفاده از بخش‌هایی از مدل است که از قبل آموزش دیده‌اند، بدون آن‌که نیازی به اجرای دوباره تنظیم دقیق برای استخراج ویژگی‌ها از متون ضروری باشد. در این روش، مدل با داده‌های برچسب‌دار آموزش داده نمی‌شود و به‌جای آن، از بخش‌های ازپیش‌آموزش‌دیده برای استخراج ویژگی‌ها از متن‌ها استفاده می‌شود.

برای استخراج ویژگی، ابتدا متن‌ها به قطعات کوچکتر تقسیم می‌شوند و سپس با استفاده از بخش‌هایی از این مدل که پیش‌آموزش دیده‌اند، برای هر بردار ویژگی جدیدی ایجاد می‌شود. این بردارهای ویژگی معمولا برای وظایفی مانند تحلیل احساسات، تشخیص موجودیت‌ها و خلاصه‌سازی متن استفاده می‌شوند. مزیت اصلی استخراج ویژگی در مدل ،BERT عدم نیاز به آموزش دوباره مدل برای هر وظیفه خاص است. این روش به‌عنوان یک روش سریع و موثر در زمینه پردازش زبان طبیعی در سیستم‌هایی که نیاز به پردازش بسیار زیادی از اطلاعات دارند، مانند موتورهای جست‌وجو و سیستم‌های پردازش زبان طبیعی، استفاده می‌شود.

در این‌جا از رویکرد نقاب‌زنی یا ماسک‌گذاری نیز استفاده می‌شود که به معنی مخفی کردن بخشی از ورودی‌ها در فرایند پیش‌آموزش است. در این روش، برای هر جمله ورودی، بخشی از کلمات به‌صورت تصادفی مخفی می‌شوند و مدل سعی می‌کند کلمات مخفی‌شده را با توجه به سایر کلمات ورودی پیش‌بینی کند. به‌طور دقیق‌تر، در تکنیک ماسک‌گذاری، برای هر جمله ورودی، ۱۵ درصد از کلمات به‌صورت تصادفی مخفی می‌شوند. سپس، در مرحله پیش‌آموزش، مدل سعی می‌کند برای هر کلمه مخفی‌شده، کلمه متناظر را پیش‌بینی کند. به‌عنوان مثال، در جمله «من به دانشگاه رفتم و کتاب‌هایم را با خودم بردم»، کلمه «رفتم» در این تکنیک به‌صورت تصادفی مخفی می‌شود و مدل سعی می‌کند کلمه متناظر با آن یعنی «به» را پیش‌بینی کند. رویکرد ماسک‌گذاری در مدل به‌دلیل این‌که مدل در پیش‌آموزش با کلماتی که مخفی شده‌اند برخورد داشته است، به بهبود عملکرد مدل در وظایف پردازش زبان طبیعی کمک می‌کند. همچنین، این تکنیک باعث می‌شود مدل برای فهم بهتر جملات، به ترتیب کلمات و ارتباط بین آن‌ها دقت کند.

۶.۲.۳ ویژگی ها

مدل BERT به‌دلیل ویژگی‌های کلیدی که دارد مورد توجه مهندسان یادگیری ماشین و پردازش زبان طبیعی قرار دارد. از جمله این ویژگی‌ها به موارد زیر باید اشاره کرد:

۱. **مبتنی بر معماری ترنسفورمر**: مدل BERT بر مبنای معماری ترنسفورمر طراحی شده است که یک شبکه عصبی پیشرفته برای پردازش زبان طبیعی است و امکان پردازش همزمان دو جهت دنباله کلمات را دارد.

۲. **پیش‌آموزش با مجموعه داده بزرگ**: مدل BERT با استفاده از مجموعه داده بزرگی که شامل متون مختلف و فاقد برچسب‌گذاری است، پیش‌آموزش داده شده است. این پیش‌آموزش به مدل BERT امکان فهم بهتر مفهوم کلمات و جملات را می‌دهد.

۳. **قابلیت تشخیص موجودیت‌ها**: مدل BERT در وظایف تشخیص موجودیت‌ها مانند شخص، سازمان، محصول، مکان و غیره بسیار قوی عمل می‌کند.

۴. **قابلیت تحلیل احساسات**: مدل BERT راهکاری بسیار قدرتمند برای تحلیل احساسات متون دارد و حتا قادر است احساسات مثبت، منفی و خنثی را در یک جمله تشخیص دهد.

۵. **محاسبه جملات مشابه**: مدل BERT با استفاده از رویکردهایی مانند توجه چندسر، امکان محاسبه شباهت بین دو جمله یا پاسخ به پرسش‌های متنی را دارد.

۶. **قابلیت تفسیر**: مدل BERT امکان تفسیر نتایجی را که خود تولید می‌کند دارد و می‌تواند به‌دلیل وزن‌دهی به کلمات، نزدیک‌ترین مکالمه به زبان طبیعی را با انسان‌ها برقرار کند. از دیگر ویژگی‌های این مدل می‌توان به آموزش با استفاده از روش مدل زبانی ماسک‌زده‌شده اشاره کرد. در این روش، برای آموزش مدل، برخی از کلمات در جملات با کلمات دیگر جایگزین می‌شوند و مدل باید بتواند کلمات جایگزین را شناسایی کند. این روش آموزش، باعث می‌شود که مدل BERT بتواند بهتر درک کند که کلمات در چه زمینه‌هایی با هم مرتبط هستند که نقش موثری در پاسخ‌دهی دقیق‌تر به پرسش‌ها و ترجمه ماشینی دارد. در حال حاضر، مدل BERT یکی از محبوب‌ترین و کارآمدترین مدل‌های زبانی در حوزه پردازش زبان طبیعی است و توسط بسیاری از شرکت‌ها و سازمان‌ها برای حل مسائل پردازش زبان طبیعی استفاده می‌شود. با توجه به این ویژگی‌ها، مدل BERT به‌عنوان یکی از بهترین مدل‌های پردازش زبان طبیعی محسوب می‌شود که در بسیاری از وظایف مانند تحلیل احساسات، تشخیص موجودیت‌ها، پرسش و پاسخ و خلاصه‌سازی متن، عملکرد بسیار خوبی دارد.

۷.۲.۳ در مقام مقایسه

امروزه، بحث داغی در محافل علمی پیرامون مقایسه مدل BERT با دیگر مدل‌های پردازش زبان طبیعی در جریان است و برخی بر این باور هستند که این مدل عملکرد بهتری نسبت به رقبا دارد. برخی از مدل‌هایی که با مدل BERT قابل مقایسه هستند به‌شرح زیر هستند:

ELMo: مدل ELMo یکی از مدل‌های کارآمد در حوزه پردازش زبان طبیعی است که در سال ۲۰۱۸ میلادی معرفی شد. بررسی‌های انجام‌شده نشان می‌دهد که مدل BERT در بسیاری از وظایف پردازش زبان طبیعی از ELMo بهتر عمل می‌کند.

GPT-۲: مدل GPT-۲ [۳۷] یکی دیگر از مدل‌های حوزه پردازش زبان طبیعی است که در سال ۲۰۱۹ میلادی معرفی شد. در مقام مقایسه با مدل BERT، GPT-۲ در مورد وظایفی مثل تولید متن و ترجمه ماشینی، بهتر عمل می‌کند، اما در وظایفی مانند تحلیل احساسات و تشخیص موجودیت‌ها، BERT بهتر عمل می‌کند.

Transformer-XL: مدل Transformer-XL [۳۸] نیز یکی دیگر از مدل‌های شاخص در حوزه پردازش زبان طبیعی است که در سال ۲۰۱۹ میلادی معرفی شد. در هر دو مورد وظایف مثل پرسش و پاسخ متنی، و نیز تحلیل احساسات و تشخیص موجودیت‌ها،مدل BERT بهتر است.

مدل BERT در بسیاری از وظایف پردازش زبان طبیعی، از جمله تشخیص موجودیت‌ها، تحلیل احساسات، پرسش و پاسخ متنی، خلاصه‌سازی متن و ترجمه ماشینی، عملکرد بسیار خوبی دارد و به‌عنوان یکی از محبوب‌ترین و پرکاربردترین مدل‌های پردازش زبان طبیعی محسوب می‌شود. به اعتقاد بسیاری از کارشناسان، BERT یک مدل زبانی قدرتمند است که نقطه عطفی در حوزه پردازش زبان‌های طبیعی به‌شمار می‌شود. این مدل امکان استفاده از تکنیک یادگیری انتقالی را در حوزه پردازش زبان‌های طبیعی به‌وجود آورده و در بسیاری از وظایف این حوزه عملکرد خوبی ارائه کرده است و بدون تردید در آینده نزدیک در انجام طیف گسترده‌ای از کارها کمک خوهد کرد.

۸.۲.۳ کاربردها

مدل BERT بدلیل قابلیت‌های بسیار بالایی که در پردازش زبان طبیعی دارد، در انجام انواع مختلفی از وظایف مرتبط با پردازش زبان طبیعی مورد استفاده قرار می‌گیرد. برخی از روش‌های استفاده از این مدل به شرح زیر هستند:

۱. **تشخیص موجودیت‌ها**: مدل BERT قادر است به خوبی نام اشخاص، محل‌ها و شرکت‌ها را در جملات تشخیص دهد.

۲. **پرسش و پاسخ**: می‌توان از مدل BERT برای پاسخ‌گویی به پرسش‌هایی استفاده کرد که پاسخ دقیقی برای آن‌ها وجود دارد. در این روش، مدل با دریافت یک پرسش به‌صورت خودکار بهترین پاسخ را پیدا می‌کند. این همان تکنیکی است که ChatGPT از آن استفاده می‌کند.

۳. **تشخیص احساسات**: با استفاده از مدل BERT می‌توان به خوبی احساسات مثبت و منفی متن‌ها را تشخیص داد.

۴. **خلاصه‌سازی متن**: با استفاده از مدل BERT می‌توان متن‌های طولانی را خلاصه‌سازی کرد و مهم‌ترین اطلاعات را استخراج کرد.

۵. **ترجمه ماشینی**: مدل BERT برای ترجمه ماشینی نیز استفاده می‌شود و با توجه به قابلیت‌های بالایی که در فهم زبان و ترجمه‌های دقیق‌تر دارد مورد استفاده قرار می‌گیرد. امروزه، ابزارهایی مثل گوگل ترنسلیت و بینگ مایکروسافت از این ویژگی برای ترجمه متون به زبان‌های مختلف استفاده می‌کنند.

۶. **پردازش زبان طبیعی در بازیابی اطلاعات**: با استفاده از مدل BERT می‌توان بهترین صفحات وب را برای پاسخ به پرسش‌های کاربران پیدا کرد.

کاربرد اصلی مدل BERT در زمینه پردازش زبان طبیعی است، زیرا بهترین نتیجه را ارائه می‌دهد. به‌طور مثال، در بحث تشخیص موجودیت‌ها، مدل BERT می‌تواند موجودیت‌هایی مانند نام افراد، شرکت‌ها، مکان‌ها، ساختمان‌ها و غیره را به بهترین شکل تشخیص دهد. همچنین، در پرسش و پاسخ متنی، مدل BERT می‌تواند به پرسش‌های مختلف پاسخ دهد و بر مبنای اصل استنتاج پرسش‌های مرتبط با پرسش اصلی کاربر را به او پیشنهاد دهد. در خلاصه‌سازی متن، این مدل می‌تواند یک متن طولانی را به یک خلاصه کوتاه تبدیل کند. همچنین، در ترجمه ماشینی نیز بسیار قوی عمل می‌کند و می‌تواند متون را به روشی دقیق و خوانا به زبان دیگری ترجمه کند. این مدل با در برداشتن چنین توانمندی ای می تواند در بهینه سازی نتایج جستجو را نیز موثر واقع شود. درک زبان طبیعی شامل تفسیر یک کلمه ، طبقه بندی تمایلات کاربران و پایان بندی عامیانه جملات است که می تواند برای کاربران اهمیت و کاربرد زیادی را داشته باشد. گوگل از این الگوریتم برای امتیاز و رتبه دهی به سایت ها از این الگوریتم استفاده می کند و درصورتی که محتوا و نوشته های سایت شما به زبان ساده و عامیانه باشد، کاربران راحت تر ارتباط برقرار می کنند و گوگل رتبه بهتری به این سایت ها می دهد. این مدل تأثیر بسیاری را بر نتایج جستجوی صوتی و متنی گذاشته

است. رویکردهای ادراکی زبان طبیعی یا عامیانه این مدل، گوگل را حساس به خطا کرده است. به دلیل مهارتی که این مدل در درک متن ها پیدا کرده است، بدون درک کامل زبان استاندارد، الگوی خود را به زبان های مختلف به اشتراک می گذارد و کلمات را تفسیر می کند. به همین خاطر مدل BERT در سئو بین المللی تاثیر زیادی می گذارد.
باوجود اینکه این مدل شرایط را برای بسیاری از سایت ها سخت کرده و رتبه سایت آنها را پایین آورده است. ولی می توان به این الگوریتم دید مثبت داشت. مدل BERT متن باز (Open Sourse) است؛ یعنی هرکسی می تواند از آن استفاده کند.
گوگل ادعا می کند که کاربران می توانند تنها ۳۰ دقیقه در واحد پردازش (TPU) محتوای اطلاعاتی خود را تغییر دهند و البته طی چند ساعت در واحد گرافیکی یک سیستم پیشرفته پرسش و پاسخ ترتیب دهند. در حال حاضر بسیاری از سازمان ها و گروه های تحقیقاتی و گروه های وابسته به گوگل در حال تنظیم معماری BERT برای آموزش با نظارت هستند تا از آن برای بهینه سازی انجام کارهایی خاص یا پیش آموزش (مثلاً تغییر نرخ یادگیری) استفاده کنند. برخی از مواردی که از این مدل استفاده شده واست عبارت اند از:
PatentBERT [۳۹]: این مدل از یک مدل BERT دقیق برای طبقه بندی و دسته بندی قانون حق اختراع استفاده می کند.
SciBERT [۴۰]: یک مدل پیش فرض است که بیشتر برای مطالب علمی استفاده می شود.
VideoBERT [۴۱]: این مدل BERT به شکل بصری و زبانی است که برای فهمیدن جملات درون ویدئو های بدون تگ در یوتیوب است. مدل VideoBERT بر روی بیش از یک میلیون فیلم آموزشی در گروه های مختلف مانند آشپزی ، باغبانی و تعمیر وسایل نقلیه انجام شده است.
TinyBERT [۴۲]: این مدل نتایج بهتری را نسبت به خود BERT اصلی دارد، ۵.۷ برابر کوچک تر و ۴.۹ سریع تر از خود مدل BERT است.

ParsBERT ۹.۲.۳

آموزش و هر استفاده از مدل BERT روی متون انگلیسی انجام میشود. در نتیجه این مدل با پارامترهای آموزش دیده روی متون به زبان فارسی، کارایی مورد انتظار را بدست نمی دهد. در نتیجه مدل ParsBERT [۴۳] ارائه شد که مدلی آموزش دیده روی مجموعه های عظیمی از کلمات نوشتاری و محاوره ای به زبان فارسی و مبتنی بر معماری مدل BERT است. در این مطالعه از تنظیم دقیق پارامترهای مدل ParsBERT برای ساخت یک مدل قدرتمند تشخیص نویسنده توسط مجموعه دادگان استفاده می گردد.

فصل ۴

مدلسازی

۱.۴ تنظیم پارامترها و مقادیر اولیه

هر مدلسازی و تنظیم دقیق بایستی با مقدار دهی برخی از پارامترها آغاز می شود. بنابراین پارامترهای اولیه برای مدل ParsBERT عبارت اند از بیشینه طول مجاز نشانه ها، اندازه های دسته بچ برای مجموعه دادگان آموزشی و مجموعه دادگان اعتبارسنجی مشتق شده از آن، تعداد دورهای آموزشی (یا تنظیم دقیق) و نیز نرخ یادگیری که بترتیب برابر با ۵۱۲[۱]، ۸، ۴، ۵ و 2×10^{-5} هستند. همچنین برای یاگیری و نیز تنظیم دقیق از الگوریتم بهینه سازی Adam با هدف کمینه سازی تابع هزینه Sparse Categorical Cross Entropy استفاده می شود. این تابع بدلیل ماهیت مسئله که دسته بندی و توزیع هر یک از نوشته ها بین چندین دسته است، به عنوان تابع هزینه انتخاب شده است.

۲.۴ معیارهای ارزیابی

معیارهای ارزیابی کارایی مدل ParsBERT در مسئله دسته بندی مورد مطالعه عبارت اند از:
- **صحت**: نسبت نمونه‌های مثبت درست به تمامی نمونه‌های مثبت پیش‌بینی شده را «صحت» گویند. در این معیار مخرج، پیش‌بینی مثبت مدل برای تمامی نمونه‌های موجود در مجموعه‌داده است.

$$\text{صحت} = \frac{\text{تعداد پاسخ‌های مثبت درست}}{\text{تعداد پاسخ‌های مثبت نادرست} + \text{تعداد پاسخ‌های مثبت درست}} \quad (۱.۴)$$

- **بازخوانی**: نسبت نمونه‌های مثبت درست به تمامی نمونه‌هایی که در حقیقت مثبت هستند را معیار «بازخوانی» گویند. مخرج در این معیار، جمع تمامی نمونه‌های مثبت در مجموعه‌داده است.

$$\text{Recall} = \frac{\text{تعداد پاسخ‌های مثبت درست}}{\text{تعداد پاسخ‌های منفی نادرست} + \text{تعداد پاسخ‌های مثبت درست}} \quad (۲.۴)$$

[۱] مدل BERT و در نتیجه ParsBERT توانایی پردازش عبارات و جملات با حداکثر ۵۱۲ نشانه یا کلمه را دارند.

- **امتیاز F1**: امتیاز F1 ترکیبی از دو معیار دقت و بازیابی است. از آن‌جایی که هر دو معیار دقت و بازیابی در محاسبه امتیاز F1 نقش دارند، امتیاز F1 بالاتر نشان‌دهنده عملکرد بهتر است. همان‌طور که در فرمول این معیار نیز مشخص است، به‌دلیل وجود عملگر ضرب در صورت مخرج، اگر از میزان یکی از معیارهای دقت یا بازیابی کاسته شود، امتیاز F1 بسیار نزولی می‌شود. در نتیجه امتیاز F1 یک مدل یادگیری ماشین بالاست، اگر نمونه‌های مثبت پیش‌بینی شده در حقیقت نیز مثبت بوده و هیچ نمونه مثبتی به اشتباه منفی پیش‌بینی نشده باشد. این معیار وزن یکسانی به دو معیار دقت و بازیابی می‌دهد.

$$\text{امتیاز F1} = 2 \times \frac{\text{بازخوانی} \times \text{صحت}}{\text{بازخوانی} + \text{صحت}} \qquad (۳.۴)$$

۳.۴ دستاوردها

برای تنظیم دقیق مدل ParsBERT تا وصول به یک نتیجه مطلوب، برای هر مجموعه مقدار پارامترها، هر بار به روش اعتبارسنجی ضربدری ۵ لا میزان کارایی مدل را با توجه به معیارهای ارزیابی برآورد، و در صورت لزوم، تغییراتی در مقداردهی ها یا فرآیند یادگیری اعمال کردیم. مراحل سعی و خطا تا وصول به یک نتیجه مطلوب برای مدلسازی مورد نیاز مسئله مورد نظر به شرح زیر است:

۱. با توجه به پارامترهای اولیه در ۱.۴، میانگین امتیاز F1 بدست آمده در این حالت برابر با ۸۸.۴ است.

۲. این بار مقدار بیشینه طول مجاز نشانه ها را ۵۱۲ به ۱۲۸ کاهش دادیم. میانگین امتیاز F1 بدست آمده در این حالت برابر با ۹۱.۵ است و در نتیجه با کاهش مقدار بیشینه طول مجاز نشانه ها، نتیجه بهتری حاصل شده است. با وجود این نتیجه مطلوب، چند تنظیم مقادیر دیگر انجام شد با این امید که نتیجه بهتری حاصل شود.

۳. با حفظ آخرین تغییرات مقادیر، این بار اندازه های دسته بچ برای مجموعه دادگان آموزشی و مجموعه دادگان اعتبارسنجی مشتق شده را بترتیب از ۸ و ۴ به ۱۶ و ۸ تغییر دادیم. میانگین امتیاز F1 بدست آمده در این حالت برابر با ۸۴.۶ است که این، از هر دو نتیجه قبلی بدست آمده نامطلوب تر است. بنابراین مقادیر پیشین اندازه های بچ را حفظ می کنیم.

۴. به عنوان یک سعی و خطای دیگر، با حفظ مقادیر تلاش دوم، لحاظ کلمات توقف بری یادگیری و تنظیم دقیق مدل را حذف کردیم. میانگین امتیاز F1 بدست آمده در این حالت برابر با ۸۵.۲ است. این نتیجه به اندازه نتیجه تلاش دوم مطلوب نبوده و حاکی از آن است که وجود کلمات توقف در فرآیند یادگیری یا تنظیم دقیق مدل BERT یا ParsBERT باعث اقزایش کارایی آنان می شود.

۵. به عنوان پنجمین و آخرین تلاش، نرخ یادگیری را کمی کاهش داده و از مقدار 2×10^{-5} به مقدار 1.8×10^{-5} رساندیم. با حفظ مقادیر دیگر پارامترها، میانگین امتیاز F1 بدست

آمده در این حالت برابر با ۸۹.۸ است که مقدار مطلوبی است هرچند همچنان به مطلوبیت نتیجه حاصل در تلاش دوم نیست.

بنابراین مقادیر پارامتر تعیین شده در تلاش دوم به عنوان مقادیر مطلوب، و مدل تنظیم دقیق شده حاصل به عنوان مدل مطلوب، اختیار می شوند.

۴.۴ نتیجه گیری

این مطالعه نشان داد که مدل ParsBERT بطور کلی می تواند با مورد تنظیم دقیق واقع شدن، هر مجموعه دادگان متنی از حوزه های مخالف را بین برچسب های متناظر (نویسندگان آنها) با دقت مطلوبی دسته بندی و توزیع کند[۲]. در مطالعه بعدی مقایسه ای بین عملکردهای دو مدل از پیش آموزش دیده ParsBERT و مدل ParsGPT [۴۴] انجام خواهیم داد.

[۲] مشابه چنین نتیجه ای برای مدل BERT روی متون به زبان انگلیسی گزارش شده است.

واژه‌نامه فارسی به انگلیسی

Word	Meanings
BERT بزرگ	BERTLARGE
BERT پایه	BERTBASE
ادبیات عامیانه یا کوچه بازاری	Slang or Street Language
ادبیات مربوط به حوزه خاص	Domain-Specific Language
استخراج ویژگی	Feature Extraction
اعتبارسنجی ضربدری	Cross Validation
الگوریتم پس انتشار خطا	Backpropagation Algorithm
الگوریتم‌های دسته بندی	Classification Algorithms
الگوریتم‌های بهینه سازی	Optimizaion Algorithms
امتیاز F۱	F1-score
انفجار گرادیان	Exploding Gradient
آزمایش تورینگ	Turing Test
بازخوانی	Recall
بازیابی اطلاعات	Information Retrieval
برازش بیش‌ازحد	Overfitting
بدون برچسب	Unannotated
برچسب	Label
برچسب‌گذاری اجزای کلام	Part-Of-Speech Tagging (POS Tagging)
بهینه سازی محدب	Convex Optimization
پاسخ‌های مثبت درست	True Positives
پاسخ‌های مثبت نادرست	False Positives
پاسخ‌های منفی درست	True Negatives
پاسخ‌های منفی نادرست	False Negatives
پایه	Base
پردازش زبان طبیعی	Natural Language Processing (NLP)
پرسش و پاسخ	Question and Answer
پیش پردازش	Pre-process
تابع فعالساز	Activation Function

تابع هزینه	Loss Function
تبدیل	Transformation
تجزیه کردن مبتنی بر قاعده	Rule-based Parsing
تحلیل لغوی	Lexical Analysis
تحلیل مولفه های اصلی	Principle Components Analysis
تحلیل معنایی	Semantic Analysis
تحلیل نحوی	Syntactic Analysis
تشخیص موجودیت‌ها	Named Entity Recognition
تشخیص نویسنده‌ها	Authors Identification
تعدد معنایی	Polysemy
تنظیم دقیق	Fine-tuning
توجه چندسر	Multihead Attention
جهان بلوکی	Block Universe
چت‌بات	ChatBot
سوگیری (تعصب) موجود در داده‌های آموزشی	Bias in Training Data
حوزه صحبت	Genre
خروجی	Output
خلاصه‌سازی متن	Text Summarization
دسته	Class
دسته بَچ	Batch
دسته بند بیز ساده	Naïve Bayes Classifier
دودویی	Binary
رمزگشا	Decoder
رویکرد	Technique
سرریز	Overflow
دقت	Accuracy
دور	Epoch
رمزگذار	Encoder
رمزگذار دو سویه نمایشی از ترنسفورمرها	Bidirectional Encoder Representations from Transformers (BERT)
شبکه عصبی پرسپترون	Perceptron Neural Network
شبکه عصبی پیچشی	Convolutional Neural Network
شبکه عصبی پیش‌خور	Feed Forward Neural Network
شبکه عصبی بازگشتی	Recurrent Neural Network
شبکه عصبی تابع پایه شعاعی	Radial Basis Function Neural Network
شبکه عصبی چند لایه پرسپترون	Multilayer Perceptron
شبکه عصبی ماژولار	Modular Neural Network
شبکه عصبی مصنوعی	Artificial Neural Network
صحت	Precision
طعنه	Sarcasm

فاصله اقلیدسی	Euclidean Distance
فهم زبان طبیعی	Natural Language Understanding (NLU)
کارایی	Efficiency
کاهش گرادیان	Gradient Descent
کلمات توقف	Stop Words
گره‌های لایه پنهان	Hidden Nodes
لا	Fold
لایه خودتوجه	Self-Attention Layer
لایه شبکه پیشخور	Feed Forward Network Layer
لایه فشرده ساز	Pooling Layer
ماسک‌گذاری	Masking
مبدل	Transformer
مبهم بودن عبارت	Phrase Ambiguity
مجموعه دادگان	Dataset
مجموعه متنی	Corpus
محوشدگی گرادیان	Vanishing Gradient
مدل پیش‌آموزش داده شده	Pre-trained Model
مدل حافظه بلند کوتاه مدت	Long Short-Term Memory (LSTM) Model
مدل زبانی آموزش بدون نظارت	Unsupervised Language Model
مدل زبانی عمیق	Deep Language Model
مدل زبانی ماسک‌زده‌شده	Masked Language Model
مدل واحد بازگشتی گیت	Gated Recurrent Unit Model
مدل‌های n-گرمی	n-grams Models
مدل‌های تعبیه‌ی کلمات	Word Embedding Models
مدل‌های رمزگذار–رمزگشا	Encoder-Decoder Models
معماری	Architecture
مولفه تعیین کننده	Parameter
نشانه	Token
ورودی	Input
وزن	Weight
وظیفه	Task
هوش مصنوعی	Artificial Intelligence
یادگیری انتقالی	Transfer Learning
یادگیری با نظارت	Supervised Learning
یادگیری چند وظیفگی همزمان	Simultaneous Multi-Task Learning
یادگیری عمیق	Deep Learning
یادگیری ماشین	Machine Learning

واژه‌نامه انگلیسی به فارسی

کلمه	تعریف
Accuracy	دقت
Activation Function	تابع فعالساز
Architecture	معماری
Artificial Intelligence	هوش مصنوعی
Artificial Neural Network	شبکه عصبی مصنوعی
Authors Identification	تشخیص نویسنده‌ها
Backpropagation Algorithm	الگوریتم پس انتشار خطا
Base	پایه
Batch	دسته بچ
BERTBASE	BERT پایه
BERTLARGE	BERT بزرگ
Bias in Training Data	سوگیری (تعصب) موجود در داده‌های آموزشی
Bidirectional Encoder Representations from Transformers (BERT)	رمزگذار دو سویه نمایشی از ترنسفورمرها
Binary	دودویی
Block Universe	جهان بلوکی
ChatBot	چت‌بات
Class	دسته
Classification Algorithms	الگوریتم‌های دسته بندی
Convex Optimization	بهینه سازی محدب
Convolutional Neural Network	شبکه عصبی پیچشی
Corpus	مجموعه متنی
Cross Validation	اعتبارسنجی ضربدری
Dataset	مجموعه دادگان
Decoder	رمزگشا
Deep Language Model	مدل زبانی عمیق
Deep Learning	یادگیری عمیق
Domain-Specific Language	ادبیات مربوط به حوزه خاص

Efficiency	کارایی
Encoder	رمزگذار
Encoder-Decoder Models	مدل‌های رمزگذار_رمزگشا
Epoch	دور
Euclidean Distance	فاصله اقلیدسی
Exploding Gradient	انفجار گرادیان
F1-score	امتیاز F۱
False Negatives	پاسخ‌های منفی نادرست
False Positives	پاسخ‌های مثبت نادرست
Feature Extraction	استخراج ویژگی
Feed Forward Network Layer	لایه شبکه پیشخور
Feed Forward Neural Network	شبکه عصبی پیش‌خور
Fine-tuning	تنظیم دقیق
Fold	لا
Gated Recurrent Unit Model	مدل واحد بازگشتی گیت
Genre	حوزه صحبت
Gradient Descent	کاهش گرادیان
Hidden Nodes	گره‌های لایه پنهان
Information Retrieval	بازیابی اطلاعات
Input	ورودی
Label	برچسب
Lexical Analysis	تحلیل لغوی
Long Short-Term Memory (LSTM) Model	مدل حافظه بلند کوتاه مدت
Loss Function	تابع هزینه
Machine Learning	یادگیری ماشین
Masked Language Model	مدل زبانی ماسک‌زده‌شده
Masking	ماسک‌گذاری
Modular Neural Network	شبکه عصبی ماژولار
Multilayer Perceptron	شبکه عصبی چند لایه پرسپترون
Multihead Attention	توجه چندسر
n-grams Models	مدل‌های n-گرمی
Naïve Bayes Classifier	دسته بند بیز ساده
Named Entity Recognition	تشخیص موجودیت‌ها
Natural Language Processing (NLP)	پردازش زبان طبیعی
Natural Language Understanding (NLU)	فهم زبان طبیعی
Optimizaion Algorithms	الگوریتم‌های بهینه سازی
Output	خروجی
Overfitting	برازش بیش‌ازحد
Overflow	سرریز

Parameter	مولفه تعیین کننده
Part-Of-Speech Tagging (POS Tagging)	برچسب‌گذاری اجزای کلام
Perceptron Neural Network	شبکه عصبی پرسپترون
Phrase Ambiguity	مبهم بودن عبارت
Polysemy	تعدد معنایی
Pooling Layer	لایه فشرده ساز
Pre-process	پیش پردازش
Pre-trained Model	مدل پیش‌آموزش داده شده
Precision	صحت
Principle Components Analysis	تحلیل مولفه های اصلی
Question and Answer	پرسش و پاسخ
Radial Basis Function Neural Network	شبکه عصبی تابع پایه شعاعی
Recall	بازخوانی
Recurrent Neural Network	شبکه عصبی بازگشتی
Rule-based Parsing	تجزیه کردن مبتنی بر قاعده
Sarcasm	طعنه
Self-Attention Layer	لایه خودتوجه
Semantic Analysis	تحلیل معنایی
Simultaneous Multi-Task Learning	یادگیری چند وظیفگی همزمان
Slang or Street Language	ادبیات عامیانه یا کوچه بازاری
Stop Words	کلمات توقف
Supervised Learning	یادگیری با نظارت
Syntactic Analysis	تحلیل نحوی
Task	وظیفه
Technique	رویکرد
Text Summarization	خلاصه‌سازی متن
Token	نشانه
Transfer Learning	یادگیری انتقالی
Transformation	تبدیل
Transformer	مبدل
True Negatives	پاسخ‌های منفی درست
True Positives	پاسخ‌های مثبت درست
Turing Test	آزمایش تورینگ
Unannotated	بدون برچسب
Unsupervised Language Model	مدل زبانی آموزش بدون نظارت
Vanishing Gradient	محوشدگی گرادیان
Weight	وزن
Word Embedding Models	مدل‌های تعبیه‌ی کلمات

کتاب‌نامه

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv.org, May 24, 2019. https://arxiv.org/abs/1810.04805

[2] M. Koppel, J. Schler, and E. Bonchek-Dokow. Measuring Differentiability: Unmasking Pseudonymous Authors. Journal of Machine Learning Research, 8:12611276, 2007.

[3] İ. Kılıç, "Perceptron Model: The Foundation of Neural Networks," Medium, Sep. 15, 2023. https://medium.com/@ilyurek/perceptron-model-the-foundation-of-neural-networks-4db25b0148d

[4] "Feedforward Deep Learning Models · AFIT Data Science Lab R Programming Guide," Github.io, 2016. https://afit-r.github.io/feedforward-DNN (accessed Aug. 05, 2024).

[5] S. Raschka, "MultilayerPerceptron: A simple multilayer neural network - mlxtend," Github.io, 2014. https://rasbt.github.io/mlxtend/user-guide/classifier/MultiLayerPerceptron/ (accessed Aug. 05, 2024).

[6] M. T. García-Ordás, J. A. Benítez-Andrades, I. García-Rodríguez, C. Benavides, and H. Alaiz-Moretón, "Detecting Respiratory Pathologies Using Convolutional Neural Networks and Variational Autoencoders for Unbalancing Data," Sensors, vol. 20, no. 4, p. 1214, Feb. 2020, doi: https://doi.org/10.3390/s20041214.

[7] H. Faris, Ibrahim Aljarah, and Seyedali Mirjalili, "Evolving Radial Basis Function Networks Using Moth–Flame Optimizer," Elsevier eBooks, pp. 537–550, Jan. 2017, doi: https://doi.org/10.1016/b978-0-12-811318-9.00028-4.

[8] H. Bhat, "Recurrent Neural Network: Applications and Advancements," AlmaBetter, Aug. 01, 2023. https://www.almabetter.com/bytes/articles/recurrent-neural-network

[9] Suvankar Maity, "Have you ever heard of RNN, LSTM, and GRU? Don't worry if those sound like a jumble of letters right now. I'm here to explain what they are and why they're important! Okay, let's start with RNN.," Linkedin.com, Mar. 20, 2024. https://www.linkedin.com/pulse/rnn-lstm-gru-why-do-we-need-them-suvankar-maity-joegc/ (accessed Aug. 05, 2024).

[10] Ahmadsabry, "A Perfect guide to Understand Encoder Decoders in Depth with Visuals," Medium, Jun. 24, 2023. https://medium.com/@ahmadsabry678/a-perfect-guide-to-understand-encoder-decoders-in-depth-with-visuals-30805c23659b

[11] J.-F. Qiao, X. Meng, W.-J. Li, and B. M. Wilamowski, "A novel modular RBF neural network based on a brain-like partition method," Neural Computing and Applications, vol. 32, no. 3, pp. 899–911, Oct. 2018, doi: https://doi.org/10.1007/s00521-018-3763-z.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," arXiv.org, Sep. 06, 2013. http://arxiv.org/abs/1301.3781

[13] M. Bilgin and I. F. Senturk, "Sentiment analysis on Twitter data with semi-supervised Doc2Vec," 2017 International Conference on Computer Science and Engineering (UBMK), Oct. 2017, doi: https://doi.org/10.1109/ubmk.2017.8093492.

[14] J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543, 2014, doi: https://doi.org/10.3115/v1/d14-1162.

[15] allenai, "GitHub - allenai/allennlp-models: Officially supported AllenNLP models," GitHub, Oct. 19, 2022. https://github.com/allenai/allennlp-models (accessed Aug. 05, 2024).

[16] W. Huang, A. Murakami, and J. Grieve, "ALMs: Authorial Language Models for Authorship Attribution," arXiv (Cornell University), Jan. 2024, doi: https://doi.org/10.48550/arxiv.2401.12005.

[17] Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. 2006. Effects of age and gender on blogging. In AAAI spring symposium: Computational approaches to analyzing weblogs, volume 6, page 199–205.

[18] David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. Journal of machine learning research, 5:361–397.

[19] Efstathios Stamatatos, "Authorship Attribution Using Text Distortion," Jan. 2017, doi: https://doi.org/10.18653/v1/e17-1107.

[20] Reuters Editorial, "Business, Financial News, U.S and International Breaking News | Reuters," Reuters, 2024. https://www.reuters.com/

[21] The guardian, "News, sport and opinion from the Guardian's UK edition | The Guardian," the Guardian, 2023. https://www.theguardian.com/uk

[22] B. Huang, C. Chen, and K. Shu, "Can Large Language Models Identify Authorship?," arXiv (Cornell University), Mar. 2024, doi: https://doi.org/10.48550/arxiv.2403.08213.

[23] A. Abbasi, A. R. Javed, F. Iqbal, Z. Jalil, T. R. Gadekallu, and N. Kryvinska, "Authorship identification using ensemble learning," Scientific Reports, vol. 12, no. 1, Jun. 2022, doi: https://doi.org/10.1038/s41598-022-13690-4.

[24] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," arXiv.org, Feb. 29, 2020. https://arxiv.org/abs/1910.01108v4

[25] K. Luyckx and W. Daelemans, "The effect of author set size and data size in authorship attribution," Literary and Linguistic Computing, vol. 26, no. 1, pp. 35–55, Aug. 2010, doi: https://doi.org/10.1093/llc/fqq013.

[26] M. Koppel, J. Schler, and S. Argamon, "Authorship attribution in the wild," Language Resources and Evaluation, vol. 45, no. 1, pp. 83–94, Jan. 2010, doi: https://doi.org/10.1007/s10579-009-9111-2.

[27] X. He, Arash Habibi Lashkari, Nikhill Vombatkere, and Dilli Prasad Sharma, "Authorship Attribution Methods, Challenges, and Future Research Directions: A Comprehensive Survey,"

Information, vol. 15, no. 3, pp. 131–131, Feb. 2024, doi: https://doi.org/10.3390/info15030131.

[28] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: https://doi.org/10.1162/neco.1997.9.8.1735.

[29] R. Oppliger, “Automatic authorship attribution based on character n-grams in Swiss German.,” no. 16, pp. 177–185, Sep. 2016, doi: https://doi.org/10.5167/uzh-169627.

[30] A. Maćkiewicz and W. Ratajczak, “Principal components analysis (PCA),” Computers and Geosciences, vol. 19, no. 3, pp. 303–342, Mar. 1993, doi: https://doi.org/10.1016/0098-3004(93)90090-r.

[31] J. Savoy, “Authorship Attribution: A Comparative Study of Three Text Corpora and Three Languages,” Journal of Quantitative Linguistics, vol. 19, no. 2, pp. 132–161, May 2012, doi: https://doi.org/10.1080/09296174.2012.659003.

[32] H. Gómez-Adorno, G. Sidorov, D. Pinto, and I. Markov, “A Graph Based Authorship Identification Approach Notebook for PAN at CLEF 2015.” Accessed: Aug. 05, 2024. [Online]. Available: https://ceur-ws.org/Vol-1391/135-CR.pdf

[33] ”hazm: A Python library for digesting Persian text.” PyPI.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” 2009 IEEE Conference on Computer Vision and Pattern Recognition, Jun. 2009, doi: https://doi.org/10.1109/cvpr.2009.5206848.

[35] “BERT base vs BERT large,” OpenGenus IQ: Computing Expertise Legacy, Jan. 12, 2021. https://iq.opengenus.org/bert-base-vs-bert-large/

[36] S. Kumar, “Bidirectional Encoder Representations from Transformers (BERT),” Medium, Dec. 17, 2023. https://medium.com/@shravankoninti/bidirectional-encoder-representations-from-transformers-bert-ccfe032ccb3d (accessed Aug. 01, 2024).

[37] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language Models are Unsupervised Multitask Learners,” 2018.

Available: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models-are-unsupervised-multitask-learners.pdf

[38] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context," arxiv.org, Jan. 2019, doi: https://doi.org/10.48550/arXiv.1901.02860.

[39] J.-S. Lee and J. Hsiang, "PatentBERT: Patent Classification with Fine-Tuning a pre-trained BERT Model," arXiv (Cornell University), Jan. 2019, doi: https://doi.org/10.48550/arxiv.1906.02124.

[40] Iz Beltagy, K. Lo, and A. Cohan, "SciBERT: A Pretrained Language Model for Scientific Text," arXiv (Cornell University), Mar. 2019, doi: https://doi.org/10.48550/arxiv.1903.10676.

[41] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "VideoBERT: A Joint Model for Video and Language Representation Learning," Apr. 2019, doi: https://doi.org/10.48550/arxiv.1904.01766.

[42] X. Jiao et al., "TinyBERT: Distilling BERT for Natural Language Understanding," Association for Computational Linguistics, 2020. Accessed: Mar. 13, 2024. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.372.pdf

[43] Mehrdad Farahani, Mohammad Gharachorloo, Marzieh Farahani, Mohammad Manthouri. "ParsBERT: Transformer-based Model for Persian Language Understanding." ArXiv, 2020.

[44] Hooshvare Team, "ParsGPT2, a Persian version of GPT2," GitHub repository, 2021, Available: https://github.com/hooshvare/parsgpt

Abstract

The classification of members into predefined categories is a fundamental application in artificial intelligence and machine learning. One specific instance of this is the task of author identification for texts. Words often carry multiple meanings, especially in different contexts, making feature extraction challenging.
Initial efforts to address text-related problems introduced word embedding models. Although effective, these models were shallow and contained limited information. In 2018, Google engineers developed BERT, a more robust model trained on extensive data, and made it publicly available.
This model is actually a group of trained encoders of the transformer model, and this study aims to identify authors of Persian texts using ParsBERT, a model based on it. The efficiency of ParsBERT in author recognition is analyzed and interpreted.

College of Science
School of Mathematics, Statistics, and Computer Science

Identifying the Author of Persian Texts Using BERT Model

Sepehr Abbaspour

Supervisor: Dr. Hedieh Sajedi

A thesis submitted in partial fulfillment of the requirements for the degree of B.Sc. in Computer Science

July, 2024