پردیس علوم
دانشکده ریاضی، آمار و علوم کامپیوتر

مروری بر پردازش زبان طبیعی و مدل‌های زبانی فشرده

نگارنده

یزدان زندیه وکیلی

استاد راهنما: دکتر هدیه ساجدی

پایان‌نامه برای دریافت درجه کارشناسی
در رشته علوم کامپیوتر

مرداد ۱۴۰۲

چکیده

در این کار، قلمرو مدل‌های زبان فشرده را مورد بررسی قرار گرفته، با تمرکز بر پیشرفت‌های پیشگامانه در پردازش زبان طبیعی (NLP) که تعادلی بین کارایی و عملکرد ارائه می دهد. با افزایش پیچیدگی و تقاضای برنامه‌های NLP، نیاز به مدل‌هایی که می‌توانند در پلتفرم‌های دارای محدودیت منابع، مانند دستگاه‌های موبایل و سیستم‌های محاسباتی لبه‌ای، مستقر شوند، به طور فزاینده‌ای حیاتی می‌شود. هدف اصلی این تحقیق بررسی و ارزیابی کارآمدی مدل‌های زبان فشرده از جمله Mo-bileBERT، TinyBERT، MiniLM و DistilBERT و مناسب بودن آن‌ها برای وظایف NLP در دنیای واقعی است.
روش تحقیق شامل یک بررسی متون گسترده است که یک پایه نظری برای بررسی اصول معماری و طراحی مدل های زبان فشرده ارائه می دهد. تجزیه و تحلیل کامل تکنیک‌های انتقال و فشرده سازی دانش به کار رفته در این مدل‌های فشرده، مبنایی را برای درک کارایی آن‌ها تشکیل می‌دهد. ارزیابی‌های تجربی و معیارها برای تحلیل عملکرد این مدل‌ها در وظایف و مجموعه داده‌های متنوع NLP، مقایسه آن‌ها با همتایان بزرگ‌تر و مدل‌های سنتی NLP انجام می‌شود. کاوش موارد کاربرد عملی، مزایای بالقوه مدل‌های زبان فشرده را در سناریوهای دنیای واقعی، به‌ویژه در زمینه محیط‌های محاسباتی موبایل و لبه، نشان می‌دهد.

اصالت اثر

اصالت اثر

هیچ قسمت از این پایان‌نامه، پیش از این در هیچ موسسه تحصیلات عالی برای دریافت درجه تحصیلی استفاده نشده است. همچنین، هیچ قسمت از این پایان‌نامه برگردان فارسی تمامی یا قسمتی از یک اثر دیگر علمی (مانند مقاله، پایان‌نامه، و غیره) به زبانی دیگر نمی‌باشد. ارائه این پایان‌نامه توسط نگارنده به معاونت آموزشی (معاونت پژوهشی و تحصیلات تکمیلی برای ارشد و دکتری) به منزله تعهد نگارنده به اصالت متن و محتوای ارائه شده بر اساس یک کار پژوهشی در مدت تحصیل در دانشگاه تهران می باشد. در صورت اثبات خلاف این امر، مدرک تحصیلی اخذ شده توسط این پایان‌نامه از دانشگاه تهران، معتبر نمی باشد.

حق مالکیت معنوی

حق مالکیت معنوی

پیشگفتار

پیگیری مدل‌های پردازش زبان طبیعی کارآمد و قدرتمند (NLP) یک نیروی محرکه در زمینه هوش مصنوعی بوده است. همانطور که برنامه های NLP همچنان در پیچیدگی و برجستگی رشد می کنند، تقاضا برای مدل هایی که تعادلی بین عملکرد و کارایی منابع ایجاد می کنند بسیار مهم شده است. این پایان نامه کاوش در مدل های زبان فشرده را آغاز می کند، با تمرکز بر پیشرفت های پیشگامانه مانند MobileBERT، TinyBERT، MiniLM و DistilBERT. هدف بررسی کارآیی این مدل‌ها در افزایش قابلیت‌های NLP در عین پرداختن به چالش‌های ناشی از محیط‌های محدود به منابع مانند دستگاه‌های تلفن همراه و سیستم‌های محاسبات لبه است.
هدف اصلی این پایان نامه بررسی و ارزیابی کارآمدی مدل های زبان فشرده و مناسب بودن آنها برای کاربردهای NLP در دنیای واقعی است. اهداف اولیه عبارتند از:

- درک مبانی نظری NLP و شبکه های عصبی برای ایجاد زمینه برای بررسی مدل های فشرده.
- تجزیه و تحلیل معماری، اصول طراحی و تکنیک های به کار گرفته شده توسط Mo- bileBERT، TinyBERT، MiniLM و DistilBERT برای دستیابی به فشرده سازی و کارایی مدل.
- بررسی عملکرد و تحلیل مقایسه‌ای این مدل‌های فشرده در برابر همتایان بزرگ‌تر و مدل‌های سنتی
- بررسی کاربردپذیری و موارد استفاده بالقوه مدل‌های زبان فشرده در سناریوهای دنیای واقعی، به‌ویژه در پلتفرم‌های با منابع محدود.

منابع اولیه که زیربنای این پایان نامه است شامل مقالات تحقیقاتی اساسی و اسناد رسمی در مورد MobileBERT، TinyBERT، MiniLM و DistilBERT است. این منابع به عنوان سنگ بنای درک جنبه های نظری، طراحی معماری و ارزیابی عملکرد مدل های زبان فشرده عمل می کنند.

فهرست مطالب

فهرست تصاویر

فصل ۱

مقدمه‌ای بر پردازش زبان طبیعی

پردازش زبان طبیعی (NLP[1]) یک حوزه میان رشته ای پویا و به سرعت در حال پیشرفت است که در تقاطع زبان شناسی، علوم کامپیوتر و هوش مصنوعی قرار دارد. هدف آن اعطای توانایی به ماشین‌ها برای درک، تفسیر و تولید زبان انسانی است، بنابراین ارتباط و تعامل یکپارچه بین انسان و رایانه را ممکن می‌سازد. در هسته NLP چالش پر کردن شکاف عمیق بین ماهیت ساختار یافته داده‌های قابل خواندن توسط ماشین و پیچیدگی و ابهام ذاتی زبان طبیعی نهفته است. با توسعه الگوریتم‌ها و مدل‌های محاسباتی پیچیده، NLP به دنبال استخراج بینش‌های معنادار از داده‌های متنی بدون ساختار، تسهیل درک زبان، و فعال کردن پاسخ‌های هوشمند، انقلابی در حوزه‌های مختلف، از جمله بازیابی اطلاعات، تجزیه و تحلیل احساسات، ترجمه ماشینی، و عوامل مکالمه است.

مطالعه NLP با چالش‌های چند وجهی مواجه است که از ماهیت پیچیده زبان انسانی ناشی می‌شود. زبان تنوع گسترده ای را نشان می دهد که شامل واژگان متنوع، قواعد دستور زبان و معناشناسی وابسته به زمینه است. علاوه بر این، تفاوت‌های ظریف در معنا، عبارات اصطلاحی و مفاهیم فرهنگی، دشواری تحلیل زبان محاسباتی را تشدید می‌کند. محققان NLP تلاش می‌کنند تا سیستم‌های قوی و سازگاری را ایجاد کنند که قادر به درک ظرافت‌های بافت زبان، ابهام‌زدایی متنی از معانی کلمات، و ایجاد پاسخ‌های منسجم و متناوب مناسب باشد. NLP با الهام از زبان‌شناسی، مدل‌سازی آماری و یادگیری ماشینی، تلاش می‌کند تا رازهای زبان انسان را کشف کند و فناوری‌های جدیدی را ایجاد کند که از تعاملات سنتی انسان و رایانه فراتر می‌رود، و نوید آینده‌ای را می‌دهد که در آن ماشین‌های هوشمند به‌طور یکپارچه با انسان‌ها ارتباط برقرار کرده و با انسان‌ها تعامل دارند. زمانی به قلمرو داستان های گمانه زنی تنزل داده شد.

---

[1] Natural Language Processing

۱.۱ تاریخچه

ریشه‌های NLP به اوایل روزهای پیدایش کامپیوترها و ظهور هوش مصنوعی در دهه‌های ۱۹۵۰ و ۱۹۶۰ برمی‌گردد. تمرکز اولیه NLP بر روی ترجمه ماشین بود و پژوهش‌گران تلاش کردند تا سیستم‌هایی را ایجاد کنند که به طور خودکار یک زبان را به زبان دیگر ترجمه کنند. یکی از اولین تلاش‌های قابل ذکر در این راستا آزمایش Georgetown-IBM در سال ۱۹۵۴ بود، که از یک کامپیوتر اولیه برای ترجمه جملات روسی به انگلیسی استفاده شد. در دهه‌های ۱۹۶۰ و ۱۹۷۰، پژوهشگران در توسعه سیستم‌های مبتنی بر قوانین برای درک و ترجمه زبان تلاش‌های قابل‌توجهی انجام دادند. اما این تلاش‌های اولیه به دلیل پیچیدگی و تنوع زبان‌های طبیعی محدودیت‌هایی داشت که ساخت مجموعه جامعی از قوانین برای هر ساختار زبانی ممکن را دشوار می‌ساخت.
در دهه ۱۹۸۰، تحقیقات NLP به سمت روش‌های آماری تغییر مسیر داد. به جای تکیه صرفا بر قوانین ساخت‌یافته، پژوهش‌گران شروع به استفاده از یادگیری ماشین و مدل‌های احتمالی برای تحلیل و پردازش داده‌های زبانی کردند. این تغییر امکان انعطاف‌پذیری بیشتر در برخورد با تغییرات در زبان فراهم کرد و منجر به بهبودهای قابل‌توجهی در وظایف مختلف NLP، مانند تشخیص گفتار و درک زبان شد.
در دهه ۱۹۹۰ و اوایل دهه ۲۰۰۰، ارتقاء توانایی محاسباتی و به‌دست آمدن مجموعه‌های بزرگ از زبان، پیشرفت NLP را به شدت تسریع کرد. در این دوره، ظهور الگوریتم‌های یادگیری ماشین مانند ماشین‌های بردار پشتیبان (SVM[2])، مدل‌های مخفی مارکوف (HMM[3]) و مدل‌های احتمالی پیچیده‌تر مانند میدان‌های تصادفی مشروط (CRF[4]) در برخی از برنامه‌های NLP مشاهده شد.

۲.۱ پیش پردازش

پردازش زبان طبیعی شامل چندین مؤلفه مختلف است که هر کدام وظایف خاصی را در کل زنجیره فهم و پردازش زبان انجام می‌دهند. در ادامه برخی از اجزای کلیدی NLP آورده شده‌اند:

۱. توکن‌بندی (Tokenization) توکن‌بندی فرآیند تقسیم متن به واحدهای کوچک‌تر به نام توکن‌ها است. این توکن‌ها می‌توانند واژه‌ها، زیرواژه‌ها یا حروف باشند. توکن‌بندی مرحله اولیه مهمی در NLP است، زیرا به کامپیوتر امکان فهم و پردازش عناصر جداگانه متن را می‌دهد.
معمولاً، اولین مرحله در فرایند پیش‌پردازش مجزا سازی کلمات می‌باشد. در این مرحله متن را به کلمات مجزا تقسیم می‌کنیم که به هرکدام یک Token نیز گفته می‌شود [۱].
الگوریتم‌های مختلفی برای این کار مورداستفاده قرار می‌گیرد ازجمله Rule-Based،

[2] Support Vector Machine
[3] Hidden Markov Model
[4] Conditional Random Fields

Dictionary-Based Statistical، و Hybrid.
در روش Rule-Based با استفاده از قوانین از پیش تعیین‌شده کلمات را از هم جدا می‌کنیم. این قوانین می‌تواند شامل محل قرارگیری علائم نگارشی مثل نقطه یا علامت سؤال باشد.
الگوریتم‌های Statistical از روش‌های یادگیری ماشین برای این کار بهره می‌برند؛ به این شکل که ابتدا روی داده‌های عظیمی آموزش داده‌شده و درنهایت می‌توانند مرز میان کلمات و جملات را تشخیص دهند.
در روش Dictionary-Based نیز ابتدا فهرستی از پیش تهیه‌شده از کلمات را به سیستم می‌دهیم و سیستم هنگام برخورد با آن‌ها در متن به شکل Token استخراجشان می‌کند.
در روش Hybrid نیز از چندین روش مختلف برای بهبود سرعت و دقت استفاده می‌شود. در این روش با توجه به نوع متن می‌توان نوع روش را انتخاب کرد، به‌عنوان مثال برای پیام‌های شبکه‌های اجتماعی از نوع Rule-Based و برای متن‌های علمی از روش Statistical استفاده می‌کنیم.
توکن‌بندی، به تفکیک میزان دقت مورد نیاز برای وظیفه خاص، می‌تواند در سطوح مختلفی انجام شود که در ادامه به سه حالت از آن اشاره شده است:

(آ) توکن‌بندی واژگان (Word Tokenization):
توکن‌بندی واژگان روش معمولی توکن‌بندی است. در این رویکرد، متن به واژه‌های تکی تقسیم می‌شود. به عنوان مثال، جمله "هوا در تابستان گرم است" به توکن‌های زیر توکن‌بندی می‌شود: ["هوا"، "در"، "تابستان"، "گرم"، "است"]

(ب) توکن‌بندی زیرواژگان (Subword Tokenization):
توکن‌بندی زیرواژگان متن را به واحدهای کوچکتر تقسیم می‌کند که ممکن است به واژه‌های کامل مربوط نشوند. این روش به ویژه برای مقابله با ساخت‌یافت‌های پیچیده واژگان در زبان‌ها و برخورد با واژه‌های خارج از واژگان کاربرد دارد. روش‌های توکن‌بندی زیرواژگان محبوب شامل رمزنگاری بر پایه بایت (Byte Pair Encoding - BPE SentencePiece) می‌شوند.

(ج) توکن‌بندی کاراکتر (Charecter Tokenization):
در توکن‌بندی کاراکتر، متن به کاراکترهای تکی تقسیم می‌شود. این سطح توکن‌بندی برای مدل‌سازی زبان در سطح کاراکتر و وظایفی که املای دقیق واژگان مهم است، مفید است.

۲. تجزیه‌سازی مورفولوجیکال (Morphological Analysis) تجزیه‌سازی مورفولوژیک به مطالعه ساختار و اشکال واژه‌ها می‌پردازد. این وظایف شامل کوتاه‌کردن (Stemming) و ریشه‌یابی (Lemmatization) واژه‌ها هستند.

۳. برچسب‌گذاری قسمت‌های سخن (Part-of-Speech Tagging) برچسب‌گذاری قسمت‌های

سخن شامل اختصاص برچسب قسمت‌های سخن (اسم، فعل، صفت و غیره) به هر واژه در یک جمله است. این اطلاعات در درک ساختار گرامری و معنای یک جمله مفید است.

۴. شناسایی موجودیت‌های نام‌دار (Named Entity Recognition) هدف از NER شناسایی و طبقه‌بندی موجودیت‌های نام‌دار (مانند نام افراد، سازمان‌ها، مکان‌ها، تاریخ‌ها و غیره) در یک متن است. این کار به استخراج اطلاعات و موجودیت‌ها مفید است.

۵. نحو و تحلیل جملات (Syntax and Parsing) این مؤلفه بر روی تحلیل ساختار گرامری جملات تمرکز دارد. تجزیه‌سازی شامل تجزیه جملات به ساختار سلسله‌ای مانند درخت‌های تجزیه است که روابط بین کلمات را نشان می‌دهد.

۶. برچسب‌گذاری نقش‌های معنایی (Semantic Role Labeling) فرآیند شناسایی نقش‌هایی است که کلمات مختلف در یک جمله در مورد فعل اصلی ایفا می‌کنند. این به درک معنای معنایی و روابط درون جمله کمک می‌کند.

۳.۱ استخراج ویژگی

این مرحله در پردازش زبان طبیعی با استفاده از الگوریتم‌های یادگیری ماشین الزامی است. در این فرایند متن خود را به بردارهایی از اعداد تبدیل کرده که آماده پردازش توسط مدل‌های طراحی‌شده می‌شوند. الگوریتم‌های مختلفی برای استخراج ویژگی‌ها مورداستفاده قرار می‌گیرد که برخی از آن‌ها عبارت‌اند از:

Word2Vec ۱.۳.۱

الگوریتم Word2Vec از زمان معرفی در سال ۲۰۱۳ تاکنون به محبوبیت بالایی در کاربردهای مختلف پردازش زبان طبیعی ازجمله تحلیل احساسات دست پیداکرده است. این الگوریتم که به‌طورمعمول با دو معماری Continues Bag-of-Words (CBoW) و Skip-Gram مورداستفاده قرار می‌گیرد، از شبکه‌های عصبی برای تبدیل متن به بردارهای عددی بهره می‌گیرد.

GloVe ۲.۳.۱

روش GloVe از ترکیب دو روش تجزیه ماتریس سراسری و روش Skip-gram استفاده می‌کند. در این مدل از تعداد تکرار کلمات در هر متن برای یافتن کلمات هم‌معنی استفاده‌شده، به این شکل که به‌جای استفاده از احتمال رخداد برای رسیدن به معنای کلمه از نسبت احتمالات هم رخدادی استفاده کنیم. GloVe برخلاف CBoW و Skip-gram، به‌جای استفاده از آنتروپی متقاطع از مجموع حداقل مربعات وزن‌دار شده برای پیش‌بینی استفاده می‌کند. مدل GloVe به ازای استفاده

از مرجع آموزش، تعداد کلمات و زمان یادگیری یکسان عملکرد بهتری را نسبت به Word2Vec از خود نشان می‌دهد.

FastText ۳.۳.۱

این الگوریتم منبع باز نیز بر پایه شبکه‌های عصبی ساخته‌شده و توسط کمپانی فیس‌بوک در جهت بهبود مدل Word2Vec منتشر شد و از معماری مشابه Skip-gram استفاده می‌کند. این مدل هر یک از کلمات را نیز به بخش‌های کوچک‌تری 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 یا کلمات OOV کمک فراوانی می‌کند.

ELMo ۴.۳.۱

Elmo نوعی دیگر از مدل‌ها برای تعبیه سازی کلمه می‌باشد که بر پایه شبکه‌های عصبی ساخته‌شده است. این مدل با بهره‌گیری از معماری حافظه کوتاه و بلندمدت دولایه، جملات را از راست و چپ مورد بررسی قرار می‌دهد تا نقش کلمات بعد و قبل کلمه موردنظر را بررسی کند. این مدل برای هر کلمه موجود در متن‌های مختلف برداری را به آن اختصاص داده و مشکل وابستگی معنی کلمه به متن را برطرف می‌کند.

۴.۱ اهمیت موضوع مورد مطالعه

اهمیت مدل‌های زبانی در شکل‌دهی به آینده به‌دلیل نقش محوری آن‌ها در باز کردن پتانسیل کامل زبان بشری به عنوان رسانه اولیه ارتباط با سیستم‌های هوشمند، انکارناپذیر است. با تسریع پیشرفت در هوش مصنوعی و پردازش زبان طبیعی، مدل‌های زبان در حال تکامل هستند تا مهارت قابل توجهی در درک و تولید زبان انسانی با دقت و طبیعی بودن بی‌سابقه نشان دهند. این مدل‌ها به‌عنوان ابزارهای قدرتمندی برای پردازش حجم وسیعی از داده‌های متنی، تسهیل توسعه برنامه‌های کاربردی پیچیده در بازیابی اطلاعات، تحلیل احساسات، ترجمه زبان و حوزه‌های متعدد دیگر پدیدار شده‌اند. علاوه بر این، ظهور مدل‌های زبانی از قبل آموزش‌دیده‌شده، مانند BERT[5] و GPT[6]، عصر تحول‌آفرینی از یادگیری انتقالی را آغاز کرده است که امکان انتقال کارآمد دانش را در بین وظایف و حوزه‌ها فراهم می‌کند. این الگوی یادگیری انتقال پیامدهای عمیقی برای این زمینه دارد که باعث افزایش کارایی، قابلیت استفاده مجدد و سازگاری در طیف متنوعی از برنامه‌های NLP می‌شود.
در آینده، مدل‌های زبانی نقش اصلی را در متحول کردن تعاملات انسان_رایانه ایفا می‌کنند و ارتباط شهودی و یکپارچه‌تری را بین افراد و ماشین‌ها ممکن می‌سازند. با ظهور عوامل مکالمه پیچیده و

[5] Bidirectional Encoder Representation from Transformers
[6] Generative Pre-trained Transformers

دستیاران مجازی، مدل‌های زبانی به عنوان ستون فقرات تعاملات طبیعی و تعاملی با توانایی درک و پاسخگویی به سؤالات و درخواست‌های انسانی به شیوه‌ای مرتبط و منسجم عمل خواهند کرد. علاوه بر این، این مدل‌ها توسعه فن‌آوری‌های فراگیرتر و در دسترس‌تر را تسریع می‌کنند، موانع زبانی را از بین می‌برند و ارتباط مؤثر میان گویشوران زبان‌های مختلف را تسهیل می‌کنند. با پیشرفت تحقیقات هوش مصنوعی و NLP، مدل‌های زبان در پیشتاز باقی می‌مانند و به عنوان بلوک‌های اساسی برای نوآوری‌های آینده که جامعه را توانمند می‌کنند، زندگی انسان‌ها را غنی می‌کنند و مرزهای آنچه ماشین‌ها می‌توانند از طریق مهارت خود در درک و تولید زبان انجام دهند را بازتعریف می‌کنند. پذیرش پتانسیل کامل مدل‌های زبانی در آینده بدون شک چشم‌اندازی دگرگون‌کننده از کاربردها و تجربیات هوشمند را ایجاد می‌کند، که از شیوه‌های سنتی تعامل فراتر می‌رود و جامعه را به عصر همکاری و ارتباطات یکپارچه انسان و ماشین سوق می‌دهد.

۵.۱ چالش پردازش زبان طبیعی با روش‌های سنتی

مدل‌های NLP مبتنی بر ترانسفورماتور سنتی، در حالی که در قابلیت‌های درک زبان انقلابی هستند، چالش‌های قابل توجهی را هنگام استفاده در برنامه‌های کاربردی دنیای واقعی ارائه می‌کنند. این مدل‌ها، مانند نسخه‌های اولیه BERT، GPT-۲ و RoBERTa، از نظر اندازه عظیم هستند و صدها میلیون پارامتر را شامل می‌شوند و آنها را به حافظه فشرده و از نظر محاسباتی نیازمند می‌سازد. این محدودیت‌های عملی ایجاد می‌کند، به‌ویژه در پلتفرم‌های دارای منابع محدود مانند دستگاه‌های تلفن همراه و سیستم‌های محاسبات لبه. حافظه و نیازهای محاسباتی این ترانسفورماتورهای بزرگ مانع استنتاج بلادرنگ می‌شوند و منجر به کندی زمان پاسخ و مصرف انرژی بیش از حد می‌شوند. علاوه بر این، دانلود، ذخیره و به روز رسانی چنین مدل های بزرگی می تواند در تنظیمات محدود به پهنای باند غیر عملی شود. در پرتو این چالش ها، نیاز به مدل های ترانسفورماتور فشرده تر آشکار می شود. مدل های فشرده مانند DistilBERT، TinyBERT MiniLM و MobileBERT به عنوان راه حلی برای ایجاد تعادل بین پیچیدگی و کارایی مدل ظاهر شده اند. هدف این مدل‌های فشرده حفظ عملکرد همتایان بزرگ‌تر خود در عین کاهش قابل توجه اندازه مدل و سربار محاسباتی، امکان استقرار یکپارچه برنامه‌های قدرتمند NLP در حین حرکت و تسهیل پذیرش گسترده NLP در حوزه‌های مختلف است.

فصل ۲

شبکه‌های عصبی

روش‌های مبتنی بر شبکه‌های عصبی جایگزین مناسبی برای روش‌های سنتی یادگیری ماشین مثل Naïve Bayes می‌باشند. ساختار این شبکه‌ها با الهام گرفتن از نحوه عملکرد مغز انسان، از لایه‌هایی تشکیل‌شده که درون آن‌ها واحدهای محاسبه‌گری به نام نورون وجود دارد. شبکه عصبی با تغییر وزن میان نرون‌های متصل به یکدیگر عملیات‌های مختلفی مثل رگرسیون یا طبقه‌بندی داده‌های ورودی را انجام می‌دهد. به لایه‌های میان لایه ورودی و خروجی لایه مخفی گفته می‌شود و شبکه‌های عصبی عمیق از تعداد لایه‌های مخفی بیشتری برخوردار هستند. برخی از چالش‌های استفاده از شبکه‌های عصبی عمیق، نیاز به مجموعه داده گسترده و سیستم‌های پردازشی قدرتمند است که با پیشرفت GPU در سال‌های اخیر این مشکلات کمتر شده است [۲]. شبکه‌های عصبی از معماری‌های مختلفی مثل شبکه‌های عصبی پیچشی (CNN[1]) و حافظه کوتاه و بلندمدت (LSTM[2]) استفاده می‌کنند. در این فصل نگاه عمیقی به معماری‌های مرسوم برای پردازش زبان طبیعی و تحلیل احساسات پرداخته و تعدادی از کارهای اخیر در این زمینه را موردبررسی قرار می‌دهیم.

در اواخر دهه ۹۰ محبوبیت شبکه‌های عصبی عمیق به دلیل حجم سنگین محاسبات کمتر شده بود اما پیشرفت‌های صنعت سخت‌افزار ازجمله پردازنده‌های گرافیکی باعث پیشرفت چشمگیر این زمینه و دسترسی به نتایج بی‌نظیری در پردازش تصویر، پردازش صوت و پردازش زبان طبیعی شد [۳]. به شکل کلی، شبکه‌های عصبی از چندین لایه با عناصر پردازشی غیرخطی تشکیل‌شده که لایه‌های نزدیک به ورودی جزئیات ساده داده‌های ورودی را استخراج کرده و لایه‌های جلوتر جزئیات پیچیده‌تر لایه‌های پیشین را استخراج می‌کنند.

---

[1] Convolutional Neural Network

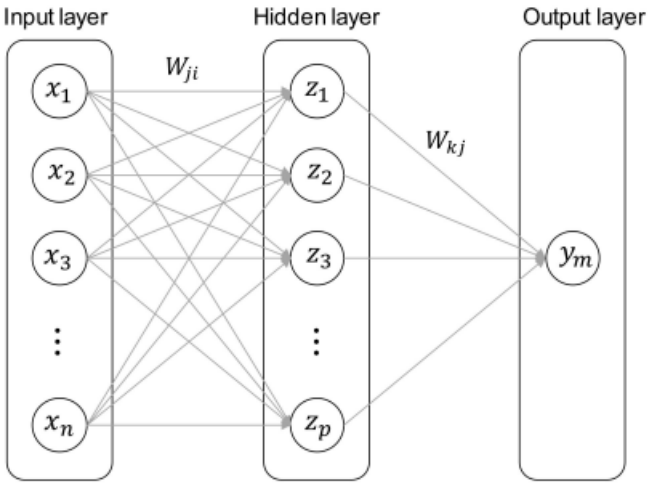
[2] Long Short-Term Memory

۱.۲ اجزاء شبکه‌های عصبی

مدل (ANN[3]) ساختار مغز را تقلید می کند و از سه لایه (ورودی، پنهان و خروجی) تشکیل شده است. این مدل با معادلات شامل پارامترهای وزن، سوگیری و توابع فعال‌سازی توصیف می‌شود. هدف اصلی مدل ANN یافتن بهترین پارامترهای وزن از طریق الگوریتم های آموزشی است که انتشار پس از آن یک روش معمول برای تنظیم است. تعداد بهینه گره های پنهان از طریق آزمون و خطا تعیین می شود و کوچکتر یا مساوی تعداد گره های ورودی اغلب نتایج بهتری را به همراه دارد. توابع فعال سازی مختلف به ANN اجازه می دهد تا روابط غیر خطی بین ورودی و خروجی را یاد بگیرد. فرآیند کلی شامل ساخت مدلی است که خطاها را در مجموعه آموزشی به حداقل می رساند و سپس آن را در مجموعه آزمایشی اعمال می کند.

$$\hat{y} = f_y(\sum_{j=1}^{p} z_j W_{kj} + b_k) \tag{۱.۲}$$

$$z_j = f_z(\sum_{j=1}^{p} x_i W_{ij} + c_j) \tag{۲.۲}$$

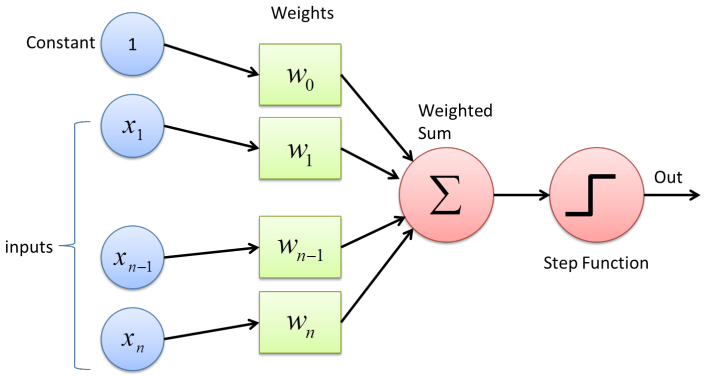


شکل ۱.۲: نمایی از معماری یک شبکه عصبی مصنوعی با یک لایه مخفی[۱]

[3] Artificail Neural Network

۲.۲ تاریخچه

- اولین مدل نورون مصنوعی (۱۹۴۳) : مفهوم شبکه های عصبی مصنوعی به سال ۱۹۴۳ باز می گردد، زمانی که وارن مک کالوچ، فیزیولوژیست عصبی و والتر پیتس، ریاضیدان، اولین مدل ریاضی یک نورون مصنوعی را معرفی کردند. آنها یک مدل ساده برای تقلید رفتار یک نورون بیولوژیکی با استفاده از منطق باینری پیشنهاد کردند. پرسپترون (۱۹۵۷) : فرانک روزنبلات Perceptron را با الهام از مدل نورون McCulloch-Pitts توسعه داد. پرسپترون یکی از قدیمی ترین و ساده ترین اشکال شبکه های عصبی مصنوعی است که قادر به یادگیری و طبقه بندی دودویی است. کار روزنبلات پایه و اساس شبکه های عصبی تک لایه را گذاشت.



شکل ۲.۲: تک نورون شبکه عصبی[۱]

- زمستان هوش مصنوعی (۱۹۶۹-۱۹۸۰) : موفقیت اولیه Perceptron باعث ایجاد خوش بینی در زمینه هوش مصنوعی و ANN شد. با این حال، زمانی که محققان متوجه محدودیت‌های پرسپترون‌های تک لایه در حل مسائل پیچیده شدند، شور و شوق کاهش یافت. بودجه برای تحقیقات هوش مصنوعی کاهش یافت و منجر به به اصطلاح "زمستان هوش مصنوعی" شد.
- الگوریتم پس انتشار(۱۹۸۶) : در سال ۱۹۸۶، روملهارت، هینتون و ویلیامز الگوریتم پس انتشار را معرفی کردند. این پیشرفت، پرسپترون‌های چند لایه (همچنین به عنوان شبکه‌های عصبی پیش‌خور شناخته می‌شوند) را قادر می‌سازد تا الگوهای پیچیده را با تنظیم وزن‌ها در شبکه به طور موثر یاد بگیرند. این الگوریتم نشان‌دهنده تجدید علاقه به شبکه‌های عصبی مصنوعی بود.
- موج دوم شبکه های عصبی (۱۹۸۰) : توسعه پس انتشار و پیشرفت در قدرت محاسباتی منجر به تجدید علاقه به شبکه های عصبی مصنوعی در دهه ۱۹۸۰ شد. محققان معماری

ها و الگوریتم های یادگیری جدید را بررسی کردند و ANN ها در کاربردهای مختلف محبوبیت یافتند.

- شبکه‌های عصبی کانولوشن (CNN) و شبکه‌های عصبی بازگشتی (RNN[4]) (۱۹۹۰) : دهه ۱۹۹۰ شاهد ظهور انواع جدیدی از ANN ها بود. در سال ۱۹۹۸، LeCun و همکاران شبکه‌های عصبی کانولوشن (CNN) را برای وظایف پردازش تصویر معرفی کرد. CNN ها بینایی کامپیوتری و تشخیص الگو را متحول کردند. تقریباً در همان زمان، محققان شبکه‌های عصبی بازگشتی (RNN) را برای پردازش متوالی داده‌ها توسعه دادند و آن‌ها را برای کارهایی مانند پردازش زبان طبیعی مناسب ساختند.

- تسلط ماشین‌های بردار پشتیبانی (SVM) (۲۰۰۰) : در حالی که ANN ها به پیشرفت خود ادامه دادند، ماشین های بردار پشتیبان (SVM) به دلیل تعمیم و کارایی خوبشان محبوبیت پیدا کردند. SVM ها اغلب در بسیاری از برنامه ها در این دوره عملکرد بهتری از هاANN داشتند.

- یادگیری عمیق و پیشرفت ها (۲۰۱۰) : دهه ۲۰۱۰ با ظهور Deep Learning شاهد یک دوره تحول برای ANN ها بود. محققان شبکه‌های عصبی عمیق را با لایه‌های متعدد (شبکه‌های عصبی عمیق یا DNN) توسعه دادند و از قدرت محاسباتی رو به رشد و در دسترس بودن مجموعه‌های داده عظیم بهره بردند. پیشرفت هایی در بینایی کامپیوتر، پردازش زبان طبیعی، تشخیص گفتار و موارد دیگر با استفاده از یادگیری عمیق حاصل شد.

Deep Learning و ANN همچنان در خط مقدم تحقیقات هوش مصنوعی قرار دارند. پیشرفت در سخت افزار، الگوریتم‌ها و در دسترس بودن داده‌ها، پیشرفت را در این زمینه تسریع کرده است. مدل های مبتنی بر ANN، مانند ترانسفورمر، BERT، GPT-۳ و دیگران، نتایج قابل توجهی را در حوزه های مختلف به دست آورده اند. تحقیقات در حال انجام با هدف رسیدگی به چالش‌هایی مانند تفسیرپذیری مدل، استحکام و مقیاس پذیری است.

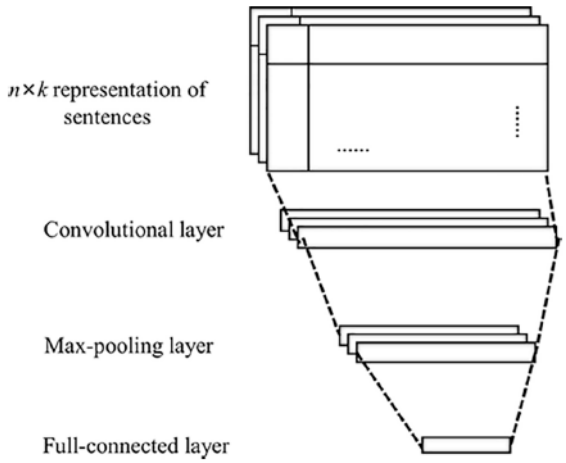
امروزه، شبکه‌های عصبی مصنوعی جزء اساسی سیستم‌های هوش مصنوعی مدرن هستند که کاربردهای پیشگامانه را در زمینه‌های مختلف، از مراقبت‌های بهداشتی و مالی گرفته تا وسایل نقلیه خودران و روباتیک، ممکن می‌سازند. تاریخچه شبکه‌های عصبی مصنوعی نشان‌دهنده اهمیت پایدار آن‌ها و پیگیری مداوم ساختن سیستم‌های هوشمندتر و سازگارتر است. ترانسفورمر (Transformer) : دسته ای از مدل های یادگیری عمیق هستند که انقلابی در پردازش زبان طبیعی (NLP) و سایر وظایف ترتیب به دنباله ایجاد کرده اند. آنها در مقاله سال ۲۰۱۷ با عنوان "توجه تنها چیزی است که نیاز دارید" توسط واسوانی و همکاران معرفی شدند. از آن زمان، ترانسفورمرها به ستون فقرات مدل های مختلف NLP تبدیل شده اند. نوآوری کلیدی Transformers در مکانیزم توجه آنها نهفته است، که به آنها اجازه می دهد تا کل توالی داده ها را به طور همزمان پردازش کنند، نه به

[4] Recurrent Neural Network

صورت متوالی مانند شبکه های عصبی بازگشتی سنتی .(RNN) این موازی سازی به طور قابل توجهی سرعت آموزش را افزایش می دهد و ترانسفورمرها را برای مدیریت وابستگی های دوربرد در داده های متوالی بسیار کارآمد می کند.

۳.۲ شبکه‌های عصبی CNN

شبکه‌های عصبی پیچشی (CNN) نوع خاصی از شبکه‌های عصبی بوده که ابتدا برای پردازش تصویر مورداستفاده قرار می‌گرفتند. فیلترهای پیچشی ۲ بعدی در این شبکه عصبی بر روی تصویر حرکت کرده و جزئیات مهم را استخراج می‌کنند. شبکه‌های عصبی پیچشی ۱ بعدی در پردازش زبان طبیعی برای پردازش و طبقه‌بندی متن استفاده شده است. CNN ها با استفاده از دو عملیات پیچش و ادغام عملیات استخراج ویژگی را انجام می‌دهند. عملیات پیچش به ضرب فیلتر با سایز مشخص در بردار متن ورودی و تغییر مکان آن بر اساس اندازه گام گفته می‌شود. پس از این مرحله نیز معمولاً یک‌لایه ادغام قرار می‌دهیم که خود دارای چند نوع می‌باشد و باعث کاهش اندازه ورودی خود می‌شود.

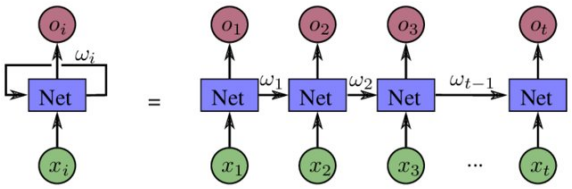


شکل ۳.۲: ساختار معماری شبکه CNN یک بعدی[۱]

۴.۲ شبکه‌های عصبی بازگشتی

شبکه‌های عصبی بازگشتی نوعی از معماری‌های مختلف شبکه‌های عصبی بوده که از نوعی حافظه داخلی بهره برده و به همین دلیل برای پردازش داده‌های دنباله‌دار بسیار مناسب می‌باشند. منظور از

حافظه در این نوع شبکه اثرپذیری خروجی عناصر یک دنباله از ورودی‌های پیشین می‌باشد. تعداد لایه‌ها در این شبکه بر اساس میزان طول دنباله ورودی مشخص می‌شود. به این شکل که اگر جمله ورودی به این شبکه از ۶ کلمه تشکیل‌شده باشد، از یک شبکه عصبی ۶ لایه برای پردازش آن استفاده می‌کنیم [۲]. این شبکه‌ها به دلیل وجود مشکلاتی مثل محوشدگی و یا انفجار گرادیان محدود به پردازش جملات کوتاه می‌باشند. برای غلبه بر این محدودیت‌ها، معماری‌های دیگری که خود از انواع RNN می‌باشند، مثل شبکه‌های عصبی بازگشتی دوطرفه، شبکه‌های عصبی بازگشتی دوطرفه عمیق، شبکه حافظه کوتاه و بلندمدت و شبکه عصبی واحد بازگشتی گیتی (GRU[5]) استفاده می‌شود که در ادامه دو معماری پراستفاده LSTM و GRU را با جزئیات بیشتری موردبررسی قرار می‌دهیم.



شکل ۴.۲: معماری شبکه عصبی بازگشتی به شکل باز و بسته[۲]

۱.۴.۲ شبکه‌های حافظه کوتاه‌ـبلند مدت

LSTM (Long Short-Term Memory) یک نوع معماری شبکه عصبی بازگشتی است که برخی از محدودیت‌های RNNهای سنتی در مواجهه با وابستگی‌های بلند مدت در داده‌ها را برطرف می‌کند. این مدل در سال ۱۹۹۷ توسط دو دانشمند، Hochreiter و Schmidhuber ،معرفی شد. در RNNهای سنتی، به دلیل مشکل از بین رفتن گرادیان، شبکه با مشکل حفظ اطلاعات مرتبط با طول توالی‌های بلند مدت مواجه می‌شود. مشکل از بین رفتن گرادیان زمانی اتفاق می‌افتد که گرادیان‌های استفاده شده برای به‌روزرسانی پارامترهای شبکه در فرآیند پس‌انتشار، به اندازه‌ی کافی کوچک می‌شوند و باعث می‌شود که شبکه در یادگیری از گام‌های زمانی اولیه دشواری داشته باشد. به عبارت دیگر، RNNها توانایی درک وابستگی‌های انتشار یافته بر روی فاصله‌های طولانی در داده‌های ورودی را دارا نیستند. شبکه‌های LSTM با داشتن سلول‌های حافظه و مکانیزم‌های دروازه‌ای طراحی شده‌اند تا مشکل از بین رفتن گرادیان را برطرف سازند. این اجزاء به شبکه‌های LSTM امکان به‌خاطرسپاری یا فراموشی انتخابی اطلاعات در طول زمان را می‌دهند که باعث می‌شود این شبکه‌ها بتوانند اطلاعات مهم را در طول توالی‌های بلند مدت حفظ کنند.

[5] Gated Recurrent Unit

اجزاء کلیدی یک شبکه LSTM به شرح زیر است:

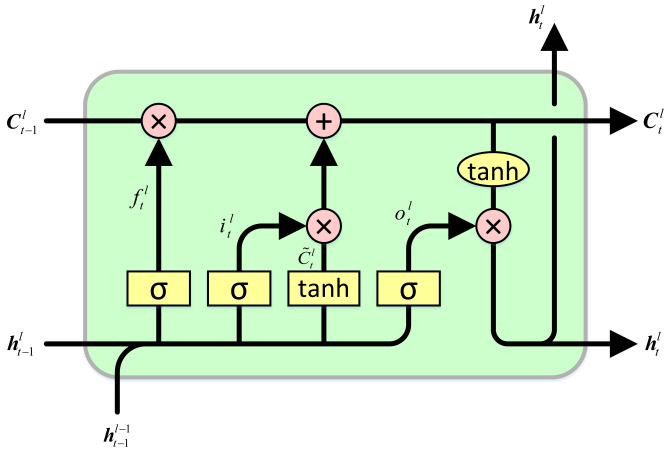
۱. حالت سلول (Cell State): این قسمت "حافظه" شبکه LSTM است که در طول تمام توالی اجرا می‌شود. حالت سلول ممکن است اطلاعات را از یک گام زمانی به گام بعدی ببرد. این حالت می‌تواند از طریق استفاده از ساختارهای دروازه‌ای، اطلاعات را به‌طور انتخابی فراموش یا حفظ کند.

۲. حالت پنهان (Hidden State): همچنین به عنوان خروجی سلول LSTM نیز شناخته می‌شود. این نسخه فیلترشده حالت سلول است و برای انجام پیش‌بینی‌ها یا انتقال اطلاعات به لایه‌ها یا گام‌های زمانی بعدی استفاده می‌شود.

۳. سه دروازه : شبکه‌های LSTM از سه نوع دروازه برای کنترل جریان اطلاعات استفاده می‌کنند:

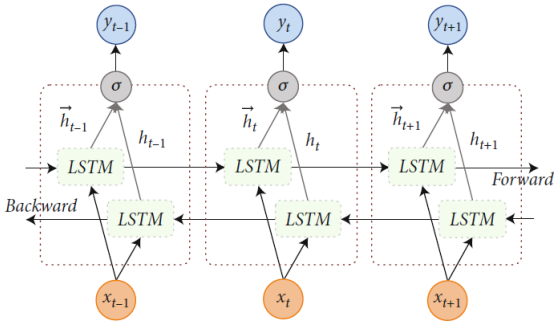
- دروازه ورودی (Input Gate): تعیین می‌کند کدام اطلاعات به حالت سلول اضافه شوند.
- دروازه فراموشی (Forget Gate): تصمیم می‌گیرد کدام اطلاعات از حالت سلول حذف شوند.
- دروازه خروجی (Output Gate): خروجی را براساس حالت سلول به‌روزرسانی می‌کند.

این دروازه‌ها از شبکه‌های عصبی sigmoid تشکیل شده‌اند که مقادیری بین ۰ و ۱ را تولید می‌کنند و نشان‌میدهند چه میزان اطلاعات اجازه عبور از طریق LSTM را دارد.
معماری LSTM به شبکه‌ها امکان یادگیری این را می‌دهد که چه زمانی باید اطلاعات را بر اساس داده‌های ورودی و متناسب با زمینه کنونی، به‌یاد بیاورد، فراموش کند یا به‌روزرسانی کند. در نتیجه شبکه‌های LSTM برای وظایف متوالی مثل پردازش زبان طبیعی، تشخیص گفتار و پیش‌بینی دنباله‌های زمانی استفاده می‌شوند.



شکل ۵.۲: ساختار درونی سلول LSTM[۲]

برخی افراد برای بهبود دقت این معماری از Bi-LSTM استفاده می‌کنند. در این معماری از دولایه مخفی برای پردازش ورودی از دو جهت استفاده می‌شود، سپس خروجی هر دو از یک تابع sigmoid دیگر عبور داده می‌شود. شکل ۶.۲ نمایش بسیار مناسبی از این مدل می‌باشد.



شکل ۶.۲: ساختار درونی سلول LSTM[۲]

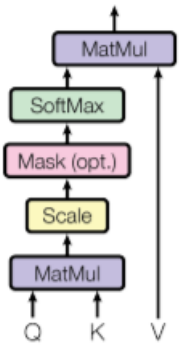
۵.۲ شبکه‌های ترانسفورمری

شبکه‌های ترانسفورمر (Transformer)، یک نوع معماری شبکه عصبی عمیق هستند که برای مدل‌سازی و پردازش داده‌های دنباله‌ای، مانند متن، استفاده می‌شوند. این معماری از ایده‌های نوآورانه‌ای برخوردار است که در مقاله "Attention Is All You Need" که در سال ۲۰۱۷ توسط Vaswani و همکارانش ارائه شد، معرفی شد. ترانسفورمر به دلیل توانایی‌های خود در

پردازش متن‌ها و داده‌های دنباله‌ای به صورت موثر و مقیاس‌پذیر، بسیار موفق بوده و از زمان ارائه‌اش، در بسیاری از پژوهش‌ها و کاربردها مورد استفاده قرار گرفته است.
اصلی‌ترین عنصر شبکه‌های ترانسفورمر، لایه مکانی‌توجه (Self-Attention Layer) است. در این لایه، ارتباطات بین عناصر داده‌های ورودی بر اساس اهمیت آن‌ها برقرار می‌شود. به عبارت دیگر، هر عنصر در دنباله با توجه به سایر عناصر، وزن‌های مخصوص خود را دارد. این وزن‌ها نشان‌دهنده اهمیت هر عنصر نسبت به سایرین است. این مکانیزم موسوم به مکانیزم توجه (Attention Mechanism) در شبکه‌های ترانسفورمر معرفی شده و باعث می‌شود شبکه بتواند اطلاعات مهم و مرتبط در دنباله‌ها را شناسایی و بهبود قابل‌توجهی در عملکرد آن‌ها داشته باشد.
شبکه‌های ترانسفورمر عموماً از چند لایه مکانی‌توجه (Self-Attention Layers) و لایه‌های کاملاً متصل (Fully connected layers) برای پردازش داده‌های ورودی استفاده می‌کنند. اطلاعات ویژگی‌ها از هر لایه عبور می‌کنند و هر لایه به اطلاعات بدست آمده از لایه‌های قبلی اضافه می‌کند. این مکانیزم اجازه می‌دهد تا شبکه به تدریج اطلاعات پیچیده‌تر و مفهومی‌تری از داده‌های ورودی را بدست آورد و این امر به بهبود قابل‌توجهی در توانایی‌های مدل‌سازی و تفسیر دنباله‌ها منجر می‌شود. شبکه‌های ترانسفورمر به دلیل این قابلیت‌های خاص و اثرگذار، در بسیاری از کاربردها از جمله ترجمه ماشینی، تولید متن، خلاصه‌سازی متن، پرسش و پاسخ، تحلیل احساسات، تولید شرح بر تصاویر و بسیاری دیگر از وظایف NLP مورد استفاده قرار می‌گیرند. علاوه بر این، معماری ترانسفورمر به علت قابلیت‌های مقیاس‌پذیری و توانایی آموزش موازی، از دیگر مزایای مهم آن‌هاست که به محبوبیت بیشتر آن‌ها در جامعه‌ی تحقیقاتی و صنعتی کمک کرده است.

۱.۵.۲ مکانیزم توجه

مدل ترانسفورمرها می‌تواند با استفاده از مکانیزم توجه بر اساس جمله‌ی در حال پردازش، تمرکز را روی اطلاعات مرتبط با آن تنظیم کند. این مدل با توجه به حالات کد گذار و کد گشا، وزن‌های لایه توجه را محاسبه می‌کند. البته مکانیزم توجه خود چندین نوع دارد که به‌عنوان مثال می‌توان به توجه عمومی ، توجه به خود و توجه چند سر اشاره کرد. در ساختار معماری ترانسفورمرها از توجه چند سر استفاده می‌شود. این کار باعث شده که بتوان چندین بخش ورودی را به‌طور هم‌زمان به این لایه داد تا با بررسی جملات طولانی، نتایج قابل‌توجهی در تمامی زمینه‌های NLP به دست آورده شود.



شکل ۷.۲: ساختار لایه‌های خود توجه[۳]

ساختار مکانیزم توجه به خود باعث می‌شود که کلمات هر جمله با توجه به کلمات مهم اطرافشان کدگذاری بشوند. همان‌طور که در شکل ۷.۲ مشاهده می‌شود، تابع توجه سه بردار ورودی Q، K و V را دریافت کرده که به ترتیب نماد کلمات Query، Keys و Value می‌باشند. انتخاب این کلمات از سیستم‌های جستجو گرفته‌شده، به این شکل که کلمه جستجو شده (Query) با کلیدهای (Keys) مختلف مقایسه شده و درنهایت موارد مشابه به‌عنوان خروجی (Values) بازگردانی می‌شوند. مکانیزم توجه نیز عملکرد مشابهی دارد. نحوه مقداردهی به این سه بردار در کاربردهای مختلف متفاوت است، به‌عنوان مثال در مدل‌های زبانی مثل BERT یا GPT هر سه بردار از یک منبع گردآوری‌شده، اما در ماشین‌های ترجمه بردارهای K و V از زبان مبدأ و بردار Q از زبان مقصد به دست آورده می‌شود.
در مرحله اول ضرب ماتریسی دو بردار Q و K محاسبه‌شده که امتیاز میان بردارهای ورودی مختلف را به ما بازگردانی می‌کند. این امتیاز در اصل نشان‌دهنده میزان توجه به کلمات دیگر برای کلمه هدف می‌باشد. در مرحله دوم امتیازهای به‌دست‌آمده را نرمال می‌کنیم تا عملیات گرادیان در حین یادگیری بهتر انجام شود. در مرحله بعد هم توسط تابع Softmax امتیازهای نرمال شده را به‌احتمال تبدیل می‌کنیم. درنهایت بردار Values نیز در خروجی تابع Softmax ضرب شده و مقدار نهایی لایه توجه محاسبه می‌شود. روابط بیان شده به صورت زیر است:

$$S = Q.K^T \tag{۳.۲}$$

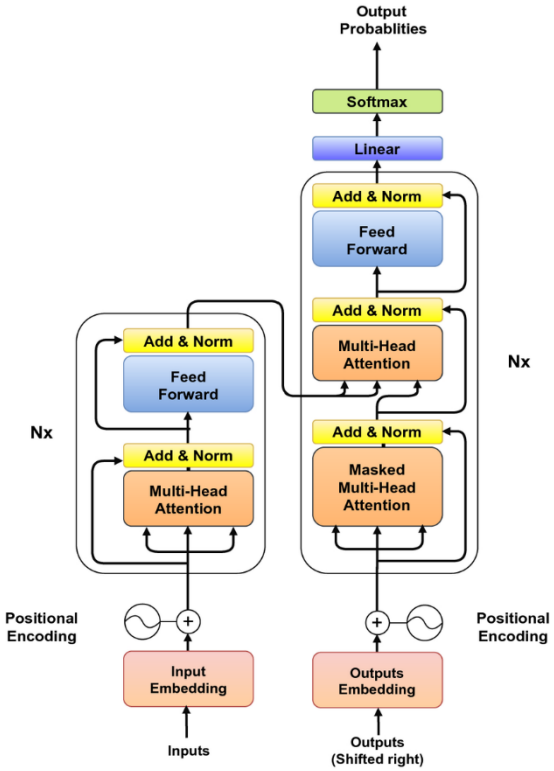
$$S_n = \frac{S}{\sqrt{d_K}} \tag{۴.۲}$$

$$P = softmax(S_n) \tag{۵.۲}$$

$$Z = P.V \tag{۶.۲}$$

۲.۵.۲ معماری ترانسفورمرها

ساختارTransformer در حوزه پردازش زبان طبیعی انقلابی بوده و در طیف گسترده‌ای از وظایف NLP به نتایج برتر دست یافته است. مدل‌های Transformer به واسطه عملکرد استثنایی در درک الگوهای پیچیده در داده‌های متوالی، به طور گسترده مورد استفاده و تطبیق قرار گرفته‌اند و در کاربردهای مختلفی خارج از NLP نیز مورد استفاده قرار گرفته‌اند، مانند بینایی ماشین و تشخیص گفتار. شکل ۸.۲ نشان‌دهنده اجزاء مختلف تراسفورمر است.



شکل ۸.۲: معماری یک شبکه ترانسفورمری[۳]

اجزاء تشکیل‌دهنده ترانسفورمر عبارتند از:

کد گذار

یک اجزای مهم در معماری Transformer است که مسئول پردازش دنباله ورودی و تولید نمایشی مرتبط برای هر جزء دنباله است. کد گذار شامل چندین لایه است و هر لایه شامل مکانیزم خودتوجه

(self-attention) و شبکه‌های عصبی feed-forward است. نحوه کارکرد این بخش در ادامه آورده شده است:

۱. تعبیه کلمات ورودی:
در ابتدای المان‌های سری ورودی به صورت متوالی در یک بردار قرار می‌گیرند که به این کار تعبیه سازی کلمات می‌گویند. هر کلمه یا توکن در دنباله ورودی با استفاده از یک ماتریس تعبیه، به یک بردار فشرده نگاشت می‌شود. هر یک از این بردارها معنای مربوط به کلمه را استخراج کرده و به عنوان ورودی اولیه به لایه کد گذار می‌دهد.

۲. توجه چند سر:
توجه چند سر دراصل تعدادی لایه توجه به خود می‌باشد که به شکل موازی با یکدیگر قرار گرفته‌اند. عملکرد اصلی در ترنسفورمرها برعهده مکانیزم خودتوجه است که به هر عنصر در دنباله ورودی اجازه می‌دهد تا به همه عناصر دیگر به صورت همزمان توجه کند. مکانیزم خودتوجه شامل سه مرحله است: رمزگذاری پرسش، کلید و مقدار (Q, K, V)، توجه نقطه‌ای و جمع‌بندی وزن‌دار مقادیر که در مباحث پیشین به آن پرداختیم.

۳. شبکه عصبی:
لایه Feed-Forward در معماری ترانسفورمر یکی از اجزای مهم لایه‌های مکرر (Re-current Layer) می‌باشد و در هر لایه از معماری ترانسفورمر وجود دارد. این لایه باعث افزایش توانایی شبکه در تشخیص الگوهای پیچیده‌تر و یادگیری ارتباطات غیرخطی بین ویژگی‌های ورودی می‌شود. وظیفه اصلی لایه Feed-Forward در ترانسفورمر، تبدیل خطی نقطه‌به‌نقطه (point-wise) از ویژگی‌های ورودی است. به عبارت دیگر، این لایه از یک تابع خطی تک‌لایه‌ای (single-layer linear function) به عنوان عملگر اصلی خود استفاده می‌کند و هیچ تابع غیرخطی اعمال نمی‌کند. فرمول عملکرد لایه Feed-Forward به صورت زیر است:

$$\mathrm{FFN}(x) = \mathrm{ReLU}(xW_1 + b_1)W_2 + b_2 \qquad (۷.۲)$$

در این فرمول، x نمایانگر ویژگی‌های ورودی است که از لایه قبل به دست می‌آید. W_1 و b_1 ماتریس و بایاس وزن‌های لایه اول (تابع خطی اولیه) هستند و W_2 و b_2 ماتریس و بایاس وزن‌های لایه دوم (خروجی نهایی) هستند. ReLU نمایانگر تابع غیرخطی است که نقاط مثبت را بدون تغییر باقی می‌گذارد و نقاط منفی را به صفر تبدیل می‌کند.
وظیفه لایه Feed-Forward این است که با اعمال تبدیل خطی و تابع غیرخطی ReLU بر روی ویژگی‌های ورودی، ویژگی‌های جدیدی ایجاد کند که باعث تفکیک الگوهای پیچیده‌تر و بهتر در ویژگی‌های ورودی شود. این تغییرات و افزایش بعد ویژگی‌ها به مدل کمک می‌کند که اطلاعات پیچیده‌تر و سطوح بالاتر از ویژگی‌های ورودی را یاد بگیرد.
مهم‌ترین نکته این است که لایه Feed-Forward به صورت جداگانه برای هر عنصر

(توکن) در دنباله اعمال می‌شود و به ازای هر عنصر، محاسبات این لایه به صورت مستقل از دیگر عناصر انجام می‌شود. این ویژگی معماری ترانسفورمر، عملکرد مدل را به صورت موازی امکان‌پذیر می‌کند که بهبود عملکرد و کارآیی آن را برای پردازش داده‌های بزرگ افزایش می‌دهد.

۴. اتصالات باقی‌مانده و نرمال سازی لایه‌ها (Residual Connections and Layer Normalization):
این دو مورد، ، دو تکنیک مهم هستند که در معماری ترانسفورمر و سایر معماری‌های مشابه استفاده می‌شوند و در کمک به آموزش مدل‌ها و بهبود عملکرد آن‌ها نقش دارند.
Residual Connections یا اتصالات باقی‌مانده از مهم‌ترین اجزاء معماری ترانسفورمر هستند که توسط He و همکارانش در مقاله "Deep Residual Learning for Image Recognition" ارائه شد. این تکنیک به معماری شبکه اجازه می‌دهد که در فرآیند آموزش، اطلاعاتی که در لایه‌های عمیق‌تر به دست می‌آید، به صورت مستقیم با داده‌های اصلی جمع شده و به عنوان خروجی این بخش در نظر گرفته میشود. این عملیات به صورت ریاضی به این شکل نمایش داده می‌شود:

$$\text{Output} = \text{Input} + \text{F}(\text{Input}) \qquad (۸.۲)$$

Layer Normalization یک تکنیک نرمال‌سازی مقدارهای خروجی یک لایه از شبکه می‌باشد. این تکنیک در لایه‌های مکرر ترانسفورمر (مانند لایه Feed-Forward و لایه Self-Attention) بکار می‌رود. هدف اصلی از نرمال‌سازی لایه، کاهش واریانس مقادیر خروجی لایه و ایجاد توزیعی با میانگین صفر و انحراف معیار یک برای خروجی‌ها است. این کار باعث استحکام‌بخشی به فرآیند آموزش شبکه و کمک به جلوگیری از مشکل گرفتار شدن مدل در نقاط ناحیه‌های خطی و ناحیه‌های بسیار خطی‌تر می‌شود. فرمول نرمال‌سازی لایه به صورت زیر است:

$$\text{Output} = \frac{\text{Input} - \text{Mean}(\text{Input})}{\sqrt{\text{Var}(\text{Input}) + \epsilon}} \times \gamma + \beta \qquad (۹.۲)$$

در این فرمول، Input نمایانگر ورودی لایه است، Mean(Input) میانگین مقادیر ورودی، Var(Input) واریانس مقادیر ورودی و ϵ یک عدد بسیار کوچک (معمولاً بسیار کوچکتر از یک) است که به جلوگیری از تقسیم بر صفر در صورتی که واریانس بسیار کوچک باشد، کمک می‌کند. همچنین،γ و β نشان‌دهنده پارامترهای قابل آموزش بهینه‌سازی نرمال‌سازی هستند.
نرمال‌سازی لایه به معماری ترانسفورمر کمک می‌کند که بهبودهای معناداری در عملکرد داشته باشد، به خصوص زمانی که مدل‌ها به صورت عمیق باشند و دارای تعداد لایه‌های بیشتری باشند.

۵. چسباندن لایه‌ها:
لایه‌های استکینگ (Stacking Layers) یکی از اجزای اصلی معماری ترانسفورمر هستند

و به معنای انتقال چندین لایه مشابه از شبکه به همدیگر است. در این معماری، چندین لایه از لایه‌های تکراری مشابه به صورت پشت سر هم قرار می‌گیرند، و خروجی لایه‌های قبلی به عنوان ورودی لایه‌های بعدی استفاده می‌شوند.
این روش از نظر مفهومی معادل اعمال یک تابع تکراری برای چندین بار بر روی ورودی است. به عنوان مثال، اگر یک لایه مکرر شامل لایه‌های Self-Attention و Feed-Forward باشد، استفاده از دو لایه استکینگ به معنای اعمال لایه‌های Self-Attention و Feed-Forward به ترتیب بر روی ورودی است.
با استفاده از لایه‌های استکینگ، مدل قادر به یادگیری ساختارها و ویژگی‌های پیچیده‌تری از داده‌ها می‌شود، زیرا هر لایه مکرر به اندازه‌ی کافی آزادی دارد تا اطلاعاتی منحصر به فرد را از ورودی به خروجی خود منتقل کند. همچنین، استفاده از لایه‌های استکینگ باعث می‌شود که مدل بتواند ارتباطات پیچیده‌تر و وابستگی‌های بین عناصر مختلف در داده‌ها را بیان کند و در نتیجه عملکرد بهتری در وظایف تشخیص الگو و ترجمه ماشینی و سایر وظایف متنی داشته باشد.
تعداد لایه‌های استکینگ معمولاً یکی از پارامترهای مدل است که می‌تواند به صورت مستقل از دیگر پارامترها انتخاب شود. معمولاً با افزایش تعداد لایه‌های استکینگ، مدل به اندازه‌ی کافی پیچیده‌تر می‌شود و از طرفی هم ممکن است نیاز به منابع محاسباتی بیشتر داشته باشد. بنابراین، انتخاب تعداد لایه‌های استکینگ باید با توجه به معیارهای عملکرد و منابع موجود در دسترس صورت گیرد.

کد گشا

لایه کدگشا (Encoder Layer) نیز از اجزای کلیدی معماری ترنسفورمر است که برای تولید خروجی‌ها در مسائل مانند ترجمه ماشینی و تولید متنی مورد استفاده قرار می‌گیرد. این لایه مسئولیت تبدیل ویژگی‌ها که از لایه‌های کد گذار به دست می‌آیند را دارد.
اجزاء لایه کدگشا نیز همانند لایه کدگذار هستند و تنها تفاوت بین این دو بخش در لایه توجه متقاطع (Cross-Attention) است. کدگشا خروجی‌های لایه کدگذار را به عنوان Query و Key درنظر گرفته و Value ها را بر اساس حاصل جمع مقادیر وزن‌دهی شده، حاصل از خروجی بخش تعبیه کلمات، به دست می‌آورد.
تعداد لایه‌های کدگشا همچنین می‌تواند به عنوان یک پارامتر تعیین شود. افزایش تعداد لایه‌های آن می‌تواند بهبودی در کیفیت خروجی‌ها و دقت تولید به ارمغان بیاورد، اما همچنین ممکن است نیاز به منابع محاسباتی بیشتری داشته باشد. ترکیب مناسب بین تعداد لایه‌های کدگشا و سایر پارامترهای مدل نقش مهمی در بهبود عملکرد و یادگیری بهتر مدل دارد.

فصل ۳

بررسی مدل‌های زبانی فشرده

در حوزه پردازش زبان طبیعی ،(NLP) مدل‌های زبان مبتنی بر ترانسفورمر روشی را که ماشین‌ها زبان انسان را درک و پردازش می‌کنند متحول کرده است. مدل پیشگام BERT (بازنمودهای رمزگذار دوطرفه از ترانسفورمرها)، با جاسازی کلمات متنی و مکانیزم‌های توجه، دروازه‌ها را به روی طیف گسترده‌ای از برنامه‌های NLP باز کرد. با این حال، موفقیت فوق‌العاده BERT به قیمت اندازه عظیم آن تمام شد، که اغلب شامل صدها میلیون پارامتر است، که در هنگام استقرار در پلت‌فرم‌های با منابع محدود مانند دستگاه‌های تلفن همراه و سیستم‌های محاسباتی لبه‌ای، چالش‌های مهمی را ایجاد می‌کند.
برای غلبه بر این چالش‌ها و در دسترس‌تر و کارآمدتر کردن ،NLP نسل جدیدی از مدل‌های ترانسفورمری فشرده پدید آمده است. این مدل‌ها، از جمله DistilBERT، TinyBERT، MiniLM و MobileBERT برای فشرده‌سازی و سرعت بخشیدن به معماری اصلی ترانسفورمر و در عین حال حفظ عملکرد رقابتی در وظایف مختلف NLP طراحی شده‌اند. آنها نشان دهنده یک تغییر پارادایم در زمینه NLP هستند و راه حل قانع کننده ای را برای مبادله بین پیچیدگی مدل و کارایی محاسباتی ارائه می دهند.
همانطور که حوزه NLP به تکامل خود ادامه می دهد و تقاضا برای برنامه های کاربردی محاسباتی موبایل و لبه افزایش می یابد، مدل های ترانسفورمر فشرده بدون شک نقش اصلی را در شکل دادن به آینده پردازش زبان ایفا خواهند کرد. کارایی، مقیاس‌پذیری و عملکرد چشمگیر آن‌ها را به دارایی‌های ارزشمندی در حوزه‌های مختلف، از جمله دستیاران مجازی، ربات‌های گفتگو، تحلیل احساسات و موارد دیگر تبدیل می‌کند.

۱.۳ Tiny BERT

به منظور کاهش بار محاسباتی مدل‌های پیچیده پردازش زبان، برخی مدل‌های کاربردی و موثرتر تعریف شده اند که از جمله آن‌ها می‌توان مدل Bert Tiny را نام برد. اندیشه اصلی تشکیل دهنده

این مدل، فشرده سازی دانش (Knowledge Distillation) است، یک تکنیک که در آن یک مدل کوچک‌تر (شاگرد) به طوری آموزش می‌بیند که رفتار و پیش‌بینی‌های یک مدل بزرگ‌تر (معلم) را تقلید کند. در مورد Tiny Bert، مدل معلم BERT با اندازه کامل است، در حالی که مدل شاگرد یک نسخه کوچک‌تراز BERT است. با بهره‌گیری از فشرده سازی دانش ، مدل کوچک‌تر یعنی Tiny Bert از منابع غنی BERT یاد می‌گیرد و اطلاعات اساسی مورد نیاز برای مسائل مختلف NLP را دریافت کرده و به طور قابل توجهی اندازه و هزینه مدل را کاهش می‌دهد [۴].

۱.۱.۳ طراحی و معماری مورداستفاده

معماری تاینی‌برت به شرح زیر می‌تواند خلاصه شود:

۱. مدل معلم (BERT):
مدل معلم BERT تمام‌اندازه است که به طور معمول از چند لایه ترانسفورمر و تعداد زیادی پارامتر تشکیل شده است. این مدل به وسیله یادگیری بدون نظارت مانند مدل‌سازی زبان مخفی و پیش‌بینی جمله بعدی، روی یک مجموعه بزرگ از متون، پیش‌آموزش داده می‌شود. مدل معلم به عنوان منبع دانش برای انتقال به مدل دانش‌آموز عمل می‌کند.

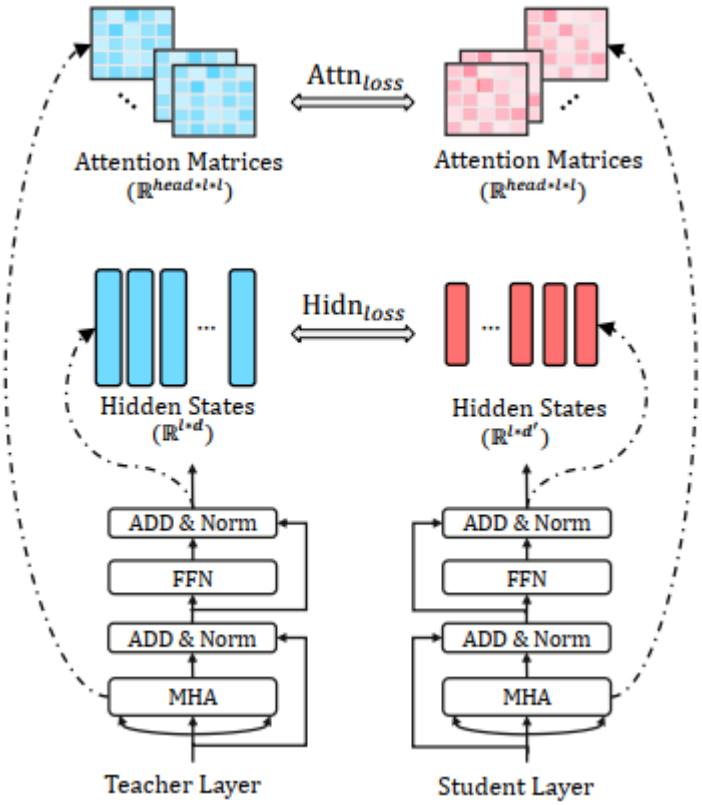
۲. مدل دانش‌اموز (Tiny BERT):
مدل دانش‌آموز نسخه‌ای کوچک‌تر و کارآمدتر از BERT از لحاظ محاسباتی است. این مدل نیز از چند لایه ترانسفورمر تشکیل شده است، اما تعداد پارامترها در مقایسه با مدل معلم کاهش می‌یابد. مدل دانش‌آموز برای استفاده در محیط‌هایی با منابع محاسباتی محدود مانند دستگاه‌های تلفن همراه طراحی شده است.

۳. فشرده سازی دانش (Knowledge Distillation):
فرآیند آموزش تاینی‌برت شامل فشرده سازی دانش است، جایی که مدل دانش‌آموز به گونه‌ای آموزش می‌بیند که رفتار مدل معلم را تقلید کند. در طول آموزش، پیش‌بینی‌های نرم مدل معلم (احتمالات) برای داده‌های آموزش به عنوان نظارت اضافی برای مدل دانش‌آموز استفاده می‌شوند. مدل دانش‌آموز آموزش داده می‌شود تا علاوه بر هدف یادگیری نظارت‌شده استاندارد (مثلاً خطای آنتروپی متقابل)، پیش‌بینی‌های نرم مشابه مدل معلم را تولید کند. لایه فشرده سازی دانش به صورت رابطه ۱.۳ مدل می‌شود:

$$L_{KD} = \sum_{x \in X} L(f^S(x), f^T(x)) \qquad (۱.۳)$$

که در این رابطه، L_{KD} نمایانگر فشرده سازی دانش است و X نمایانگر مجموعه‌ای از داده‌ها است که از آن‌ها برای این عمل استفاده می‌شود. x یک نمونه از داده‌ها در مجموعه

X است. همچنین $f^S(x)$ و $f^T(x)$ نمایانگر پیش‌بینی‌های مدل دانش‌آموز و مدل معلم به ترتیب برای نمونه x هستند. این معادله به عنوان یک روش انتقال دانش، معیاری برای انطباق پیش‌بینی‌های مدل دانش‌آموز با پیش‌بینی‌های مدل معلم ارائه می‌دهد.

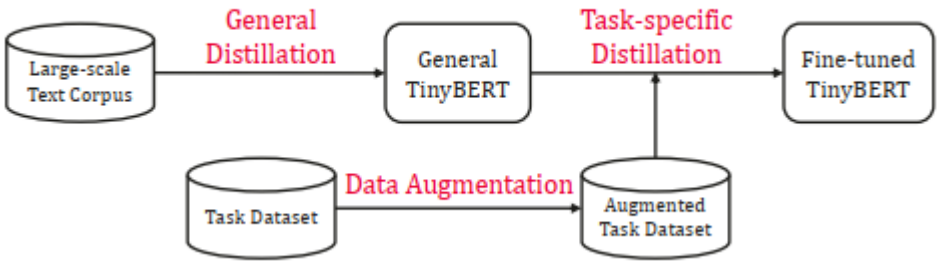


شکل ۱.۳: ساختار کلی مدل Tiny BERT[۴]

۴. خطای فشرده سازی:
تابع خطای مضاعف برای محاسبه خطای فشرده سازی. این تابع میزان شباهت در تخمین‌های مدل دانش‌آموز و مدل معلم را بررسی می‌کند. محاسبه این خطا به مدل دانش‌آموز کمک می‌کند تا دانش، انتقال مؤثر بازنمایی های غنی و قابلیت های تعمیم مدل بزرگتر را فراگیرد.
با بهره‌گیری از فشرده سازی دانش، تاینی‌برت می‌تواند الگوها و اطلاعات محتوایی مهم از مدل معلم را دریافت کند و در همین حین به صورت کارآمد و قابل دسترس در محیط‌های با منابع محاسباتی محدود استفاده شود.

۲.۱.۳ روش یادگیری

در کاربردهای مدل BERT دو مرحله یادگیری وجود دارد: پیش‌آموزش و تنظیم نهایی. دانستن مقدار زیادی از دانش کسب‌شده توسط BERT در مرحله پیش‌آموزش از اهمیت بسیاری برخوردار است و باید به مدل فشرده‌شده منتقل شود. به همین دلیل، یک چارچوب یادگیری دو مرحله‌ای نوآورانه شامل فشرده سازی عمومی (general) و فشرده سازی ویژه وظیفه (task-specific) ارائه شده است، که در شکل ۲.۳ نشان داده شده است.



شکل ۲.۳: شماتیک آموزش مدل Tiny BERT[۴]

در مرحله اول، فشرده سازی عمومی، مدل TinyBERT دانش غنی تعبیه‌شده در BERT پیش‌آموزش‌شده را یاد می‌گیرد. این فشرده سازی دانش به عنوان عامل بهبود قابلیت تعمیم مدل TinyBERT بسیار مهم است و اجازه می‌دهد که در وظایف مختلف عملکرد خوبی داشته باشد.
در مرحله دوم، فشرده سازی ویژه وظیفه، مدل TinyBERT با دانش به دست آمده از Fine-tuned BERT آموزش می‌بیند. در فرآیند تنظیم نهایی، BERT به داده‌های وظایف خاص پرداخته و به وظایف خاص NLP تطبیق می‌یابد. با استفاده از فشرده سازی دانش از Fine-tuned BERT، TinyBERT می‌تواند اطلاعات ویژه را به دست آورده و در وظایف خاص NLP عملکرد برتری داشته باشد.
با استفاده از این رویکرد انتقال دو مرحله‌ای، فاصله بین مدل‌های معلم (BERT پیش‌آموزش‌شده و Fine-tuned BERT) و دانش‌آموز (TinyBERT) به طرز چشم‌گیری کاهش می‌یابد. این به این معناست که TinyBERT، علی‌رغم اندازه کوچک‌تر آن، با فشرده سازی دانش از مدل‌های بزرگتر و قدرتمندتر، می‌تواند عملکرد رقابتی در وظایف مختلف NLP را داشته باشد.

۲.۳ مدل DistilBERT

DistilBERT، یک نسخه کوچک‌تر و سریع‌تر از BERT است که می‌تواند با عملکرد خوب در

تعداد زیادی از وظایف درک زبان طبیعی، آموزش یابد. در این مدل تلاش برای فشرده سازی در طول فرآیند پیش‌آموزش است که باعث کاهش ۴۰ درصدی اندازه مدل نسبت به مدل Bert می‌شود، همچنین ٪۹۷ از توانایی‌های درک زبانی آن را حفظ کنند و ٪۶۰ سریع‌تر عمل کنند. تابع خطا در این روش به صورت تابعی سه‌گانه از خطای مدل‌های پردازش زبانی، فشرده سازی و خطاهای cosine-distance تشکیل شده است تا مدل DitilBert عملکرد نزیک‌تری به مدل اصلی داشته باشد. در نتیجه با کاهش اندازه و حفظ عملکر اصلی، مدل حاصل قابلیت اجرا روی دستگاه‌های کوچک‌تر را خواهد داشت [۵].

۱.۲.۳ خطای سه‌گانه

در این مدل، تابع سه‌گانه خطا معرفی شده، مجموعه‌ای از سه تابع است: خطای مدل‌های پردازش زبانی، فشرده سازی و خطاهای cosine-distance. تابع خطای مدل‌های پردازش زبانی برای پیش‌بینی کلمه بعدی در یک دنباله با توجه به کلمات قبلی استفاده می‌شود. تابع خطای فشرده سازی از یک مدل معلم بزرگتر به یک مدل دانش‌آموز کوچک‌تر با تطابق حالت‌های مخفی و توزیع‌های توجه آنها استفاده می‌شود. تابع کاهش فاصله گوشی برای تشویق مدل دانش‌آموز به تولید نمایش‌های مشابه به مدل استاد برای ورودی داده شده استفاده می‌شود. با ترکیب این سه تابع، نویسندگان توانستند یک مدل نمایش زبان عمومی کوچک‌تر به نام DistilBERT را پیش‌آموزش دهند که سپس می‌تواند با عملکرد خوب در تعداد زیادی از وظایف مانند مدل‌های بزرگتر خود، به بهترین حالت تنظیم شود.

۲.۲.۳ طراحی و معماری

معماری مدل دانش‌آموز(DistilBERT) مشابه BERT است. با این حال، تعداد لایه‌ها به نسبت BERT به دو برابر کاهش می‌یابد و همچنین توکن های تعبیه سازی کلمه و pooler حذف می‌شوند. بیشتر عملیات استفاده شده در معماری ترانسفورمر (لایه خطی و نرمال‌سازی لایه) در چارچوب‌های جبر خطی مدرن به طور قابل‌توجهی بهینه‌سازی شده‌اند و بررسی‌ها نشان می‌دهد که تغییرات بر روی بعد آخر tensor (بعد مخفی) تأثیر کمتری نسبت به تغییرات بر روی عوامل دیگر، مانند تعداد لایه‌ها، بر کارایی محاسبه (با تعداد ثابت پارامترها) دارد. بنابراین، تمرکز بر روی کاهش تعداد لایه‌ها است. یک المال مهم در مدل ،DistilBERT تعیین صحیح تعداد لایه برای همگرا شدن مدل است. با استفاده از امتیاز اشتراک‌های بین دو مدل دانش‌آموز و معلم، می‌توان از هر دو لایه مدل آموزگار، یکی را برای مدل دانش‌آموز برگزید.

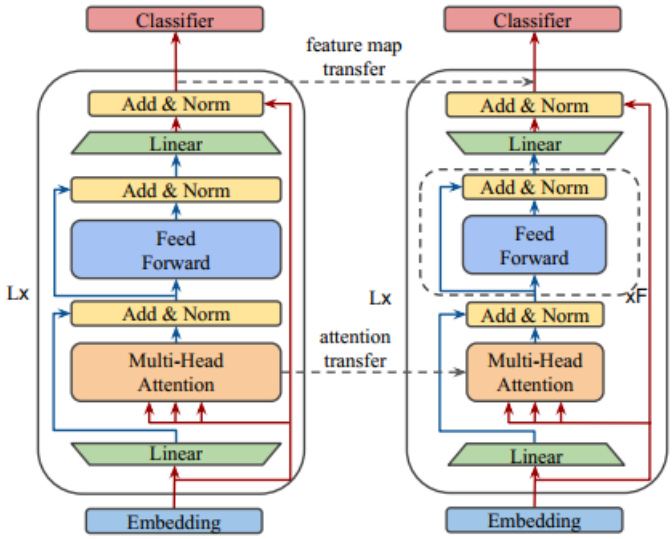
۳.۳ مدل زبانی MobileBERT

ضرورت MobileBERT از افزایش تقاضا برای قابلیت‌های پردازش زبان طبیعی (NLP) در دستگاه‌های تلفن همراه ناشی می‌شود که با منابع محاسباتی و ظرفیت‌های ذخیره‌سازی محدود مشخص می‌شوند. مدل‌های NLP از پیش آموزش‌دیده‌شده سنتی مانند BERT قابلیت‌های قابل‌توجهی در درک زبان را نشان داده‌اند، اما با یک اشکال قابل توجه همراه هستند: اندازه عظیم آنها، که اغلب شامل صدها میلیون پارامتر است. در نتیجه، استقرار این مدل‌های پرمصرف بر روی دستگاه‌های تلفن همراه به دلیل نیازهای حافظه قابل توجه و سربار محاسباتی غیرعملی می‌شود، که منجر به عملکرد کند و مصرف بیش از حد انرژی می‌شود.

MobileBERT با معرفی تکنیک‌های فشرده‌سازی و شتاب مدل نوآورانه که هدف آن ایجاد تعادل بین کارایی و عملکرد است، به این چالش حیاتی می‌پردازد. هدف اصلی MobileBERT متراکم ساختن معماری BERT به شکل فشرده‌تر و در عین حال حفظ قابلیت‌های درک زبان و اطمینان از عملکرد رقابتی در معیارهای استاندارد NLP است. MobileBERT با کاهش اندازه مدل و پیچیدگی محاسباتی، استقرار و استفاده یکپارچه از برنامه های کاربردی NLP پیچیده را در دستگاه های تلفن همراه با محدودیت منابع امکان پذیر می کند.

۱.۳.۳ معماری مورد استفاده در MobileBERT

MobileBERT یک نسخه باریک از مدل BERT LARGE است که به ساختارهای گلوگاهی و تعادلی با دقت طراحی شده بین لایه‌های خود توجه و شبکه های عصبی داخلی مجهز شده است. معماری MobileBERT در شکل ۳.۳ نشان داده شده است، هر بلوک ساختمانی در MobileBERT بسیار کوچکتر از BERT LARGE و تنها با ۱۲۸ بعد مخفی ساخته شده است. برای تنظیم ابعاد ورودی و خروجی هر بلوک به ۵۱۲، دو تبدیل خطی برای هر بلوک معرفی شده است. از این معماری به عنوان گلوگاه یاد می شود. برای غلبه بر مسئله آموزش چنین شبکه عمیق و نازکی، ابتدا یک شبکه معلم ساخته و تا زمان همگرایی آموزش داده می شود و سپس انتقال دانش از این شبکه معلم به MobileBERT انجام می شود. استراتژی های آموزشی مختلف در بخش بعدی PDF مورد بحث قرار می گیرد.



شکل ۳.۳: معماری مدل Mobile BERT[۶]

۲.۳.۳ روش یادگیری به کار گرفته‌شده

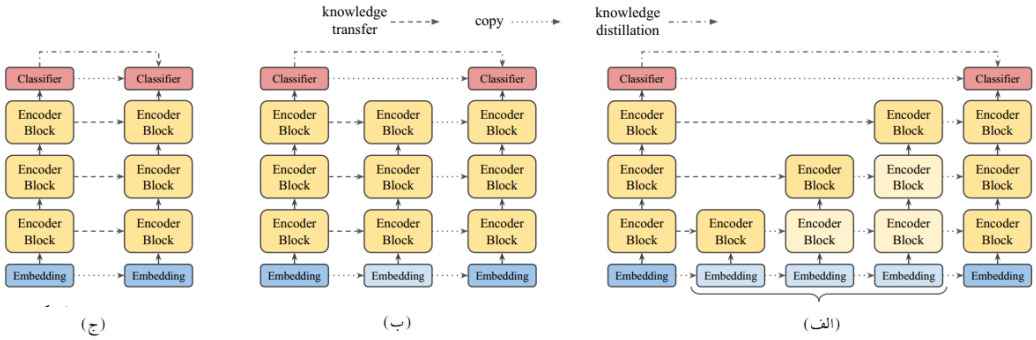
انتقال دانش در زمینه یادگیری ماشینی به فرآیند انتقال دانش یا اطلاعات به دست آمده از یک مدل به مدل دیگر اشاره دارد. هدف از انتقال دانش استفاده از تخصص و بینش های آموخته شده توسط یک مدل به خوبی آموزش دیده (معمولاً یک مدل بزرگتر یا پیچیده تر) و استفاده از آن دانش برای بهبود عملکرد مدل دیگر (اغلب یک مدل کوچکتر یا کمتر پیچیده) در مورد مشابه است. یا کارهای مرتبط این فرآیند به ویژه هنگام کار با محیط‌های محدود به منابع، مانند دستگاه‌های تلفن همراه، که در آن استقرار مدل‌های بزرگ با تعداد زیادی پارامتر ممکن است به دلیل محدودیت‌های حافظه و محاسباتی عملی نباشد، ارزشمند است.
مفهوم انتقال دانش ارتباط نزدیکی با حوزه وسیع تر یادگیری انتقالی دارد. یادگیری انتقالی شامل استفاده از دانش به دست آمده از یک کار برای بهبود عملکرد یک کار مرتبط دیگر است. به جای شروع فرآیند یادگیری از صفر برای هر کار جدید، یادگیری انتقالی به مدل‌ها اجازه می‌دهد تا بر روی بازنمایی‌ها و دانشی که قبلاً آموخته‌اند ساخته شوند، در نتیجه یادگیری را تسریع کرده و عملکرد را در کار هدف بهبود می‌بخشند [۶]. در این مقاله چندین روش برای انتقال دانش بررسی شده که عبارت‌اند از:

۱. انتقال دانش کمکی:
این نوع انتقال دانش شامل انتقال دانش از مدل معلم به مدل دانش آموز در لایه های میانی مدل است. به طور خاص، خروجی هر لایه میانی از مدل معلم برای آموزش لایه میانی مربوط

به مدل دانش‌آموز استفاده می‌شود. این به مدل دانش آموز اجازه می دهد تا از مدل معلم در سطوح مختلف انتزاعی بیاموزد که می تواند عملکرد آن را بهبود بخشد.

۲. انتقال دانش مشترک:
این نوع انتقال دانش شامل انتقال دانش از مدل معلم به مدل دانش آموز در تمام لایه های مدل به طور همزمان است. به طور خاص، خروجی هر لایه از مدل معلم برای آموزش لایه مربوط به مدل دانش آموز استفاده می شود. این به مدل دانش آموز اجازه می دهد تا از مدل معلم در تمام سطوح انتزاعی بیاموزد که می تواند عملکرد آن را بهبود بخشد.

۳. انتقال دانش پیشرو:
این نوع انتقال دانش شامل آموزش تدریجی هر لایه از مدل دانش آموز با استفاده از دانش آموخته شده از مدل معلم است. به طور خاص، الگوی دانش آموز ابتدا تنها با استفاده از دانش آموخته شده از لایه اول مدل معلم و سپس لایه دوم و به همین ترتیب تا زمانی که همه لایه‌ها آموزش داده شوند، آموزش داده می شود. این به مدل دانش آموز اجازه می دهد تا به تدریج از مدل معلم یاد بگیرد، که می تواند عملکرد آن را بهبود بخشد و در عین حال از اثرات منفی انتشار خطا جلوگیری کند.



شکل ۴.۳: شکل‌های (الف). انتقال دانش پیشرو (ب). انتقال دانش مشترک (ج). انتقال دانش کمکی [۶]

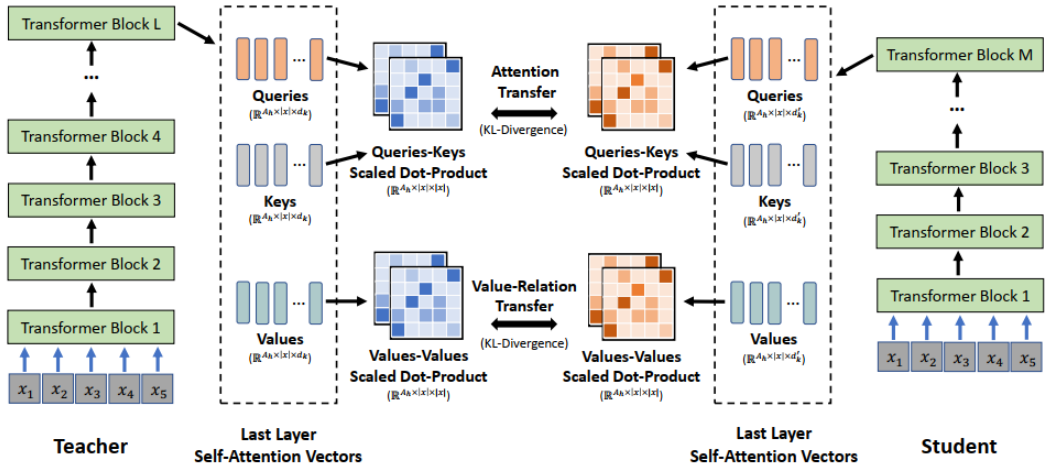
۴.۳ مدل MiniLM

این مدل چارچوب فشرده سازی خودتوجه عمیق (deep self-attention distillation) را برای انتقال مدل‌های زبانی مبتنی بر ترانسفورمر برای وظایف چندگانه ارائه می‌دهد. ایده اصلی این است که به طور عمیق ماژول‌های خودتوجه را که بخش‌های بنیادی و مهم در مدل‌های معلم و دانش‌آموز

مبتنی بر ترانسفورمر هستند، شبیه‌سازی کنیم. به خصوص، پیشنهاد می‌شود که از ماژول خودتوجه آخرین لایه ترانسفورمر مدل معلم استفاده شود. مقایسه‌ای با روش‌های پیشین نشان می‌دهد که استفاده از دانش لایه آخر ترانسفورمر به جای انجام انتقال دانش لایه به لایه، مشکلاتی مربوط به انتقال داده لایه بین مدل‌های معلم و دانش‌آموز را کاهش می‌دهد و تعداد لایه مدل دانش‌آموز می‌تواند بیشتر انعطاف‌پذیر باشد. علاوه بر این، ضرب نقطه‌ای مقیاس‌پذیر در لایه خودتوجه، عنوان دانش جدید ارائه شده است. استفاده از ضرب نقطه‌ای مقیاس‌پذیر بین مقادیر موجود در ماژول خودتوجه نیز قابلیت تبدیل ابعاد مختلف را به ماتریس‌های ارتباطی با همان ابعاد می‌دهد، بدون آن که پارامترهای اضافی معرفی شوند [۷].

۱.۴.۳ طراحی و معماری

در این روش، مدل دانش‌آموز توسط تقلید عمیق از رفتار توجه خودی لایهٔ آخر ترانسفورمر معلم، آموزش داده می‌شود. علاوه بر توزیع‌های توجه خودی، انتقال ارتباط‌های لایه توجه خودی را معرفی شده است تا به دانش‌آموز کمک کند تا تطابق عمیق‌تری را انجام دهد. شکل ۵.۳ نشان دهنده ساختار کلی این مدل است.



شکل ۵.۳: ساختار معماری مدل MiniLM[۷]

۵.۳ بررسی مدل DistilRoBERTa

در مدل DistilRoBERTa (Distilled RoBERTa)، ترکیبی از دو تکنیک کلیدی، یعنی RoBERTa و فشرده سازی دانش، به کار گرفته شده است. این مدل تلاش دارد تا از اندازه

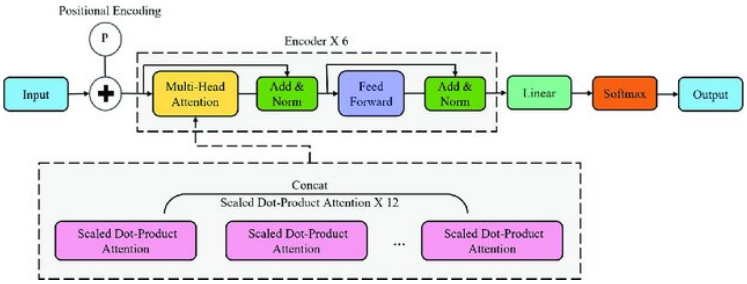
کوچکتری نسبت به RoBERTa بهره‌برداری کند، اما همچنان با کارایی و دقت قابل مقایسه با آن به وظایف پردازش زبان طبیعی بپردازد.
اصلی‌ترین بخش مدل DistilRoBERTa از مدل اصلی RoBERTa تشکیل شده است. RoBERTa یک نسخه بهبود یافته از مدل BERT است که با تمرکز بر آموزش از پیش بهینه‌شده و تغییرات مشخص در فرآیند آموزش، نتایج بهتری را در وظایف NLP ارائه می‌دهد.
در این مدل، انتقال دانش به عنوان یک تکنیک کلیدی استفاده می‌شود. مدل DistilRoBERTa از یک مدل بزرگ‌تر (به عنوان مدل استاد) به نام RoBERTa استفاده می‌کند و با استفاده از این مدل برای آموزش مدل کوچک‌تر خود (به عنوان مدل دانش‌آموز)، دانش اساسی حاصل از مدل استاد را به مدل کوچک‌تر منتقل می‌کند. این رویکرد به مدل کمک می‌کند که از دانش موجود در مدل استاد بهره‌برداری کند و در عین حال، با کاهش تعداد پارامترها، مدل کوچک‌تر را به وظایف مختلف NLP تطبیق دهد.

۱.۵.۳ طراحی و معماری مدل

اجزای اصلی معماری DistilRoBERTa به شرح زیر است:

۱. مدل RoBERTa:
هسته مدل DistilRoBERTa بر اساس مدل RoBERTa استوار است. RoBERTa نسخه بهبودیافته‌ای از مدل BERT است. RoBERTa به صورت از پیش آموزش دیده بر روی یک کرپوس بزرگ با استفاده از وظایف یادگیری بدون نظارت مختلف، مانند مدل‌سازی زبان ماسک شده و پیش‌بینی جمله بعدی، عمل می‌کند. این مدل دارای یک معماری عمیق با چندین لایهٔ ترانسفورمر است که به او امکان می‌دهد اطلاعات زبان‌شناختی غنی را از متن ورودی برداشت کند [۸].

۲. فشرده سازی دانش:
انتقال دانش تکنیک کلیدی اعمال شده در DistilRoBERTa است. در این فرآیند، DistilRoBERTa از یک مدل پیش‌آموزش دیدهٔ بزرگ‌تر، مانند RoBERTa (مدل معلم)، یاد می‌گیرد. هدف، انتقال دانش اساسی از مدل معلم به مدل کوچک‌تراست. این شامل آموزش مدل دانش‌آموز برای تقلید رفتار و پیش‌بینی‌های مدل معلم می‌شود. کار کرد این مدل نیز اساسا شبیه مدل‌های فشرده شده پیشین است. این مدل با استفاده از کنار هم قرار دادن چندین ترانسفورمر پایه جملات اصلی و ماسک آن‌ها را به عنوان ورودی در نظر گرفته و در هسته اصلی که همان مدل BERT است، پردازش می‌کند. در خروجی مدل نیز بنا به وظیفه تعریف شده برای آن ماسک‌های مورد نظر خارج میشوند. در نهایت با روش‌های سرهم سازی کلمات جملات یا عبارات هدف از مدل خارج می‌شود.



شکل ۶.۳: معماری مدل DistilRoBERTa[۸]

فصل ۴

نتیجه گیری

در این فصل، ما به نتایج کلیدی و بررسی‌ها از مقایسهٔ مدل‌های BERT، DistilBERT MINILM و TinyBERT می‌پردازیم. این مدل‌ها نقش‌های قابل توجهی در پیشرفت حوزه پردازش زبان طبیعی ایفا کرده‌اند و برای پردازش‌های مرتبط با اندازهٔ مدل، کارایی محاسباتی و محدودیت‌های منابع طراحی شده‌اند.
BERT همچنان پرقدرت‌ترین مدلی است که عملکرد برتری را در بسیاری از وظایف NLP ارائه می‌دهد. با این حال، اندازهٔ بزرگ آن باعث عدم اجرای آن در برخی از برنامه‌ها با منابع محاسباتی محدود می‌شود.
DistilBERT، MINILM و TinyBERT همه با هدف حل مشکل اندازهٔ مدل و کارایی طراحی شده‌اند. DistilBERT توازنی بین اندازه و عملکرد دارد، در حالی که MINILM و TinyBERT مدل را برای محیط‌هایی با منابع محاسباتی محدود، بهینه‌سازی می‌کنند.
MINILM و TinyBERT، کوچک‌ترین مدل‌ها، به‌خصوص برای وظایف NLP برروی دستگاه‌های خود بسیار مناسب هستند و در صورتی که از سناریوهای محاسباتی حافظه محدود استفاده می‌کنند. همهٔ چهار مدل از دانش دست‌آموز برای کسب اطلاعات اساسی از مدل BERT بزرگ‌تر بهره می‌برند، که اثربخشی این تکنیک در ایجاد مدل‌های کارآمدتر را نشان می‌دهد.

۱.۴ نتیجه‌گیری

به عنوان نتیجه‌گیری، مدل‌های زبانی نام برده شده، پیشرفتی در جهت مدل‌های NLP کارآمدتر و قابل اجرا به‌حساب می‌آیند. با توجه به نیازها و محدودیت‌های خاص یک برنامه NLP، می‌توان مدل متناسب را انتخاب کرد تا تعادلی بین اندازه مدل، عملکرد و کارآیی محاسباتی برقرار کند. این پیشرفت‌ها راه را برای راه‌حل‌های NLP قابل دسترس‌تر می‌کنند و علمی در گستره‌ی وسیعی از برنامه‌های عملیاتی می‌دهد.

جدول ۱.۴: مقایسه مدل‌های فشرده به همراه روش‌های آموزش

روش	مدل معلم	دانش فشرده‌شده
DistillBERT	Base Bert	احتمالات اهداف ساده / تعبیه‌های خروجی
TinyBERT	Base Bert	تعبیه‌های خروجی / حالات مخفی / توزیع لایه خود توجه
MobileBERT	Large IB-BERT	حالات مخفی / توزیع لایه خود توجه / احتمالات اهداف ساده
MiniLM	Base Bert	توزیع لایه خود توجه / نسبت ارزش مقادیر توجه

واژه‌نامه فارسی به انگلیسی

الف)
انتقال دانش ..... Knowledge Transfer
انفجار گرادیان ..... Exploding Gradient

پ)
پردازش زبان طبیعی ..... Natural Language Processing
پیش پردازش ..... Pre-Processing

ت)
تابع خطی ..... Linear Function
تحلیل احساسات ..... Sentiment Analysis
ترجمه ماشین ..... Machine Translation
توابع فعال‌ساز ..... Activasion Function

د)
دقت ..... Accuracy

ر)
روش‌های آماری ..... Statistical Methods

ز)
زبان‌شناسی محاسباتی ..... Computational Linguistics
زیرواژگان ..... Sub-words

ش)
شبکه‌های عصبی ..... Neural Networks

ع)

علائم نگارشی ..... Punctuation Marks

ک)

کد گذار ..... Encoder

کد گشا ..... Decoder

ل)

لایه توجه ..... Attention Layer

م)

ماتریس ..... Matrix

مبتنی بر قوانین ..... Rule-Based

محاسبات لبه ..... Edge Computing

محو شدن گرادیان ..... Vanishing Gradient

مدل دانش‌آموز ..... Student Model

مدل معلم ..... Teacher Model

مدل‌های احتمالی ..... Probabilistic Models

مدل‌های زبانی فشرده ..... Distilled Language Models

وابسته به زمینه ..... Contextualized

موجودیت نامدار ..... Named-Entity

میان رشته‌ای ..... Interdisciplinary

ن)

نرون ..... Neuron

ه)

همگرا شدن ..... Convergence

هوش مصنوعی ..... Artificial Intelligence

ی)

یادگیری بدون نظارت ..... Unsupervised-Learning

یادگیری ماشین ..... Machine Learning

واژه‌نامه انگلیسی به فارسی

A)
Artificial Neural Networks ..... شبکه‌های عصبی مصنوعی
Attention Mechanism ..... سازوکار توجه

C)
Cell State ..... حالت سلولی
Conditional Random Fields ..... میدان‌های تصادفی مشروط
Convolutional Neural Network ..... شبکه‌های عصبی پیچشی
Cosine-Distance ..... فاصله کسینوسی

D)
Deep Learning ..... یادگیری عمیق
Dictionary-Based ..... مبتنی بر واژگان

G)
Generative Pre-trained Transformers ..... ترانسفورمرهای از پیش آموزش‌دیده مولد
Graphic Processing Unit ..... واحد پردازشی گرافیک

H)
Hidden State ..... حالت مخفی
Hybrid ..... ترکیبی

K)
Key ..... کلید
Knowledge Distillation ..... فشرده‌سازی دانش

L)
Layer Normalization ..... نرمال‌سازی لایه

Long Short-Term Memory ..... حافظه کوتاه بلند مدت

M)
Markov Hidden Models ..... مدل‌های مخفی مارکوف
Morphological Analysis ..... تحلیل مورفولوجیکال

N)
Named Entity ..... موجودیت نامدار
Natural Language Processing ..... پردازش زبان طبیعی

P)
Part-of-Speech Tagging ..... برچسب گذاری بخشی از سخن
Perceptron ..... پرسپترون

Q)
Query ..... جستار

R)
Rectified Linear Unit ..... واحد خطی یکسوشده
Recurrent Neural Networks ..... شبکه عصبی بازگشتی
Rule-Based ..... مبتنی بر قوانین

S)
Semantic ..... معنا
Statistical ..... آماری
Support Vector Machines ..... ماشین‌های بردار پشتیبان
Syntax ..... نحو

T)
Tokenization ..... توکن بندی

V)
Value ..... ارزش

کتاب‌نامه

[1] P. K. Jain, V. Saravanan, and R. Pamula, *"A hybrid CNN-LSTM: A deep learning approach for consumer sentiment analysis using qualitative user-generated contents,"* Transactions on Asian and Low-Resource Language Information Processing, vol. 20, no. 5, pp. 1-15, 2021.

[2] A. Adak, B. Pradhan, and N. Shukla, *"Sentiment analysis of customer reviews of food delivery services using deep learning and explainable artificial intelligence: Systematic review,"* Foods, vol. 11, no. 10, p. 1500, 2022.

[3] L. Zhang, S. Wang, and B. Liu, *"Deep learning for sentiment analysis: A survey,"* Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 8, no. 4, p. e1253, 2018.

[4] Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., ... & Liu, *"Tinybert: Distilling bert for natural language understanding,"* arXiv preprint arXiv:1909.10351, 2019.

[5] Sanh, V., Debut, L., Chaumond, J., & Wolf, DistilBERT,*"a distilled version of BERT: smaller, faster, cheaper and lighter,"* arXiv preprint arXiv:1910.01108, 2019.

[6] Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y., & Zhou ,*"Mobilebert: a compact task-agnostic bert for resource-limited devices,"* arXiv preprint arXiv:2004.02984, 2020.

[7] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou,*"Minilm: Deep self-attention distillation for task-agnostic compression of pretrained transformers,"* Advances in Neural Information Processing Systems, 33, 5776-5788, 2020.

[8] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov,*" Roberta: A robustly optimized bert pretraining approach,"*, arXiv preprint arXiv:1907.11692, 2019.

Abstract

The future of natural language processing (NLP) hinges on the adoption and advancement of compact language models. As the demand for intelligent language processing expands to diverse domains and applications, the limitations posed by resource-constrained environments become increasingly evident. Compact language models, such as MobileBERT, TinyBERT, MiniLM, and DistilBERT, offer a transformative solution by striking a delicate balance between model complexity and efficiency. By compressing and distilling knowledge from their larger counterparts, these models enable seamless deployment on mobile devices, edge computing systems, and other constrained platforms, paving the way for on-the-go language processing. This accessibility and portability of NLP capabilities have far-reaching implications, ranging from enhancing user experiences with voice assistants to facilitating real-time language translation and sentiment analysis, all while reducing power consumption and response times.

Furthermore, the proliferation of compact language models contributes to democratizing NLP research and application development. By reducing the computational burden and memory footprint, these models empower a broader range of researchers and developers to participate in cutting-edge NLP innovations. The accessibility of compact language models allows for faster experimentation and iterative improvements, accelerating the pace of NLP advancements and fostering a vibrant and collaborative research community. As these models continue to evolve and mature, they have the potential to revolutionize human-computer interactions, making sophisticated language understanding and processing capabilities accessible to a global audience across diverse devices and applications. The significance of compact language models in the future of NLP lies not only in their technological prowess but also in their potential to reshape the landscape of human-machine communication and usher in a new era of intelligent and ubiquitous language processing.

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

A review on Natural Language Processing and Distilled Language Models

Yazdan Zandiyevakili

Supervisor: Dr.Hedie Sajedi

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

2023