



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

پروژه تحقیقاتی درس سمینار

روش‌های خلاصه‌سازی انتزاعی

نگارش

زهرا زنجانی

استاد درس

دکتر رضا صفابخش

شهریور ۱۴۰۲

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

صفحه فرم ارزیابی و تصویب پایان نامه- فرم تأیید اعضای کمیته دفاع

در این صفحه فرم دفاع یا تأیید و تصویب پایان نامه موسوم به فرم کمیته دفاع- موجود در پرونده آموزشی- را قرار دهید.

نکات مهم:

- نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو(دورو)** بلامانع است و انجام آن توصیه می شود.



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

به نام خدا

تعهدنامه اصالت اثر

تاریخ: شهریور ۱۴۰۲

اینجانب زهرا زنجانی متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است. در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

زهرا زنجانی

امضا

از استاد ارجمندم، خانم دکتر ممتازی، که با صبر و حوصله فراوان، راهنمایی‌های ارزشمندشان، حمایت‌های بی‌دینشان، نقش
بسنجایی در موفقیت این پژوهش داشتند، کمال تشکر و قدردانی را دارم.

این دستاورد کوچک را به معلم فریخته‌ام، آقای ساقی، تقدیم می‌کنم که با دانش گسترده و تخصص بی‌نظیرشان، الگوی من در عرصه علم آموزشی بودند و به من آموختند که چگونه با پشتکار و تلاش به اهدافم دست یابم.

فهرست مطالب

صفحه

عنوان

۱	۱ مقدمه
۱	۱-۱ مقدمه
۱	۲-۱ چالش‌های خلاصه‌سازی متون طولانی
۱	۳-۱ چشم‌انداز نوشتار
۲	۲ معرفی مفاهیم
۳	۱-۲ خلاصه‌سازی
۳	۲-۲ شبکه عصبی
۴	۱-۲-۲ ترنسفورمر
۴	۲-۲-۲ مدل‌های زبانی از پیش آموزش دیده و مدل‌های بزرگ زبانی
۵	۳-۲-۲ روش‌های فشرده‌سازی مدل‌های بزرگ زبانی
۵	۴-۲-۲ تقطیر دانش
۶	۵-۲-۲ کوانتایزاسیون
۶	۶-۲-۲ هرس کردن
۶	۷-۲-۲ تنظیم دقیق بهینه‌شده پارامترها برای مدل‌های زبانی بزرگ
۷	۳-۲ توهم
۹	۳ مرور تاریخچه
۱۰	۱-۳ روش‌های مبتنی بر ساختار
۱۰	۱-۱-۳ روش مبتنی بر درخت
۱۰	۲-۱-۳ روش مبتنی بر قالب
۱۱	۳-۱-۳ روش مبتنی بر هستان‌شناسی
۱۱	۴-۱-۳ روش عبارت مقدمه و بدنه
۱۱	۵-۱-۳ روش مبتنی بر گراف
۱۲	۶-۱-۳ روش مبتنی بر قانون
۱۲	۲-۳ روش‌های مبتنی بر مدل ترنسفورمر

۲۰	۱-۲-۳ ایده‌های ارائه شده بهبود خلاصه‌سازی متون طولانی
۲۵	۳-۳ روش‌های مبتنی بر یادگیری تقویتی
۲۵	۱-۳-۳ یادگیری تقویتی برای حل چالش‌های مدل دنباله به دنباله عمیق
۲۶	۲-۳-۳ یادگیری تقویتی برای ترکیب خلاصه‌های استخراجی و انتزاعی
۲۷	۳-۳-۳ یادگیری تقویتی برای ایجاد معیارها و پاداش‌های جدید
۲۸	۴-۳-۳ یادگیری تقویتی برای ایجاد خلاصه متناسب با نیاز کاربر
۳۱	۴-۳ روش‌های مبتنی بر مدل‌های زبانی بزرگ و چالش‌ها
۳۲	۱-۴-۳ هزینه‌گویی در مدل‌های زبانی بزرگ
۳۳	۲-۴-۳ هرس مدل‌های زبانی
۳۵	۳-۴-۳ هزینه‌گویی در خلاصه‌سازی
۳۷	۵-۳ معیارهای ارزیابی خلاصه‌سازی خودکار
۴۰	۱-۵-۳ محدودیت‌ها و پیشرفت‌ها
۴۱	۴ روش ارائه شده
۴۲	۵ نتایج
۴۳	کتاب‌نامه
۵۱	واژه‌نامه‌ی انگلیسی به فارسی
۵۴	واژه‌نامه‌ی انگلیسی به فارسی

شکل	فهرست تصاویر	صفحه
۱-۳	معماری مدل تی‌برت‌سام [۲۷]	۱۳
۲-۳	تعبیه مدل تی‌برت‌سام [۲۷]	۱۴
۳-۳	معماری ترنسفورمر تی‌برت‌سام [۲۷]	۱۵
۴-۳	چهارچوب ایجاد خلاصه با طول متغیر [۴۲]	۱۷
۵-۳	عمل‌های پیش‌آموزش بارت [۲۱]	۱۷
۶-۳	ساختار مدل پگاسوس [۴۹]	۱۹
۷-۳	الگوریتم امدات [۳۹]	۱۹
۸-۳	معماری مدل ترنسفورمر از بالا به پایین [۳۴]	۲۲
۹-۳	لایه خودتوجهی تقویت شده ادغام شده [۴۶]	۲۴
۱۰-۳	یک نمونه از نمودار منحنی بازیابی بر اساس طول [۳۸]	۳۰
۱۱-۳	معماری مدل ام‌سام [۳۷]	۳۱
۱۲-۳	رویکرد <i>SlimSum</i> برای حل تضادهای معنایی در خلاصه‌سازی	۳۷

فصل اول

مقدمه

۱-۱ مقدمه

با رشد روزافزون اینترنت، حجم محتوای متنی در اینترنت (به عنوان مثال وب سایت‌ها، اخبار، وبلاگ‌ها، شبکه‌های رسانه‌های اجتماعی و غیره) به صورت تصاعدی افزایش می‌یابد. در نتیجه، کاربران زمان زیادی را صرف یافتن اطلاعات مورد نظر خود می‌کنند و نمی‌توانند تمام محتوای متنی نتایج جستجو را بخوانند. خلاصه‌سازی خودکار اسناد می‌تواند به شناسایی مهم‌ترین اطلاعات و صرفه‌جویی در وقت خوانندگان کمک کند. خلاصه‌سازی خودکار متن فرآیند تولید یک متن کوتاه است که بخش‌های اصلی یک سند طولانی‌تر را پوشش می‌دهد. یک خلاصه خوب جنبه‌های مهمی مانند خوانایی، انسجام، نحو، غیر زائد بودن، ترتیب جملات، مختصر بودن، تنوع اطلاعات و پوشش اطلاعات را در نظر می‌گیرد [۵].

در سال‌های گذشته تلاش‌های زیادی برای تولید خلاصه‌سازی خودکار قابل قبول و خوانا صورت گرفته است. پژوهش‌های مرتبط با عمل خلاصه‌سازی خودکار متن در دهه ۵۰ میلادی شکل گرفتند. در یکی از این پژوهش‌ها لوهن و همکاران روشی برای خلاصه‌سازی اسناد علمی ارائه دادند که در آن تابعی بر اساس فرکانس تکرار کلمات یا عبارات به عنوان ویژگی تعریف می‌شود و وزن‌های مرتبط با این ویژگی‌ها خلاصه استخراج می‌شود [۲۶]. در کارهای تحقیقاتی اولیه، مدل‌های غیرعصبی مبتنی بر ساختار برای تولید خلاصه‌سازی خودکار مورد استفاده قرار گرفتند. با شروع دوره‌ی شبکه‌های عصبی عمیق پژوهش‌ها بر روی خلاصه‌سازی بیشتر شد. رویکردهای نوین خلاصه‌سازی شامل شبکه‌های عصبی عمیق دنباله به دنباله^۱، روش‌های بر پایه‌ی مدل ترنسفورمر^۲ و مدل‌های زبانی از پیش آموزش دیده^۳ می‌باشد.

۲-۱ چالش‌های خلاصه‌سازی متون طولانی

۳-۱ چشم‌انداز نوشتار

¹Deep neural sequence to sequence models²transformer³Pretrained language models (PTLMs)

فصل دوم

معرفی مفاهیم

۱-۲ خلاصه‌سازی

خلاصه‌سازی متن به‌عنوان یکی از وظایف کلیدی در حوزه پردازش زبان طبیعی، نقش مهمی در مدیریت و تحلیل داده‌های متنی ایفا می‌کند. این فرآیند شامل فشرده‌سازی اطلاعات موجود در اسناد طولانی به شکلی مختصر و هدفمند است که معنای اصلی متن حفظ شود. در این راستا، استفاده از مدل‌های از پیش آموزش‌دیده در خلاصه‌سازی انتزاعی به دلیل توانایی آن‌ها در تولید متون خلاصه روان و دقیق، به‌طور فزاینده‌ای مورد توجه قرار گرفته است. این مدل‌ها با ترکیب تکنیک‌های پیشرفته یادگیری عمیق و دانش قبلی، امکان استخراج اطلاعات کلیدی را از اسناد حجیم فراهم می‌کنند و ابزار مؤثری برای کاربردهای متنوعی مانند جستجوی اطلاعات و تحلیل محتوا ارائه می‌دهند. این فرآیند به دلیل تنوع در کاربردها و نیازهای مختلف، شامل رویکردها و طبقه‌بندی‌های متنوعی می‌شود. از جمله این طبقه‌بندی‌ها می‌توان به خلاصه‌سازی تک‌سندی و چندسندی اشاره کرد؛ در خلاصه‌سازی تک‌سندی بر یک متن واحد تمرکز می‌شود و در خلاصه‌سازی چندسندی اطلاعات مرتبط از چند سند ترکیب می‌شود. همچنین، خلاصه‌سازی می‌تواند به صورت استخراجی، انتزاعی یا ترکیبی انجام شود. رویکرد استخراجی جملاتی از متن اصلی انتخاب می‌کند، در حالی که رویکرد انتزاعی، با تولید جملات جدید، ایده‌های اصلی را منتقل می‌کند. علاوه بر این، خلاصه‌ها می‌توانند از نظر زبانی تک‌زبانه، چندزبانه یا بین‌زبانه باشند. روش‌های خلاصه‌سازی نیز به دو دسته نظارت‌شده و نظارت‌نشده تقسیم می‌شوند که هر یک با توجه به نوع داده‌ها و اهداف مختلف، عملکرد و محدودیت‌های خاص خود را دارند.

[۵، ۴]

۲-۲ شبکه عصبی

شبکه‌های عصبی یکی از ابزارهای کلیدی در یادگیری ماشین هستند که بر اساس ساختار و عملکرد مغز انسان طراحی شده‌اند. این شبکه‌ها از لایه‌هایی متشکل از نورون‌های مصنوعی تشکیل می‌شوند که با اتصال‌های وزنی به یکدیگر مرتبط شده‌اند و داده‌ها را به صورت سلسله‌مراتبی پردازش می‌کنند. هر نورون اطلاعاتی را از نورون‌های لایه قبلی دریافت کرده، آن را پردازش می‌کند و به نورون‌های لایه بعدی منتقل می‌سازد. این فرآیند باعث می‌شود شبکه‌های عصبی بتوانند ویژگی‌های پیچیده داده‌ها را استخراج کرده و روابط غیرخطی میان متغیرها را یاد بگیرند. با استفاده از الگوریتم‌های بهینه‌سازی مانند پس‌انتشار خطا، این شبکه‌ها به‌طور مداوم تنظیم شده و دقت خود را در انجام وظایف مختلف مانند طبقه‌بندی،

رگرسیون و یادگیری عمیق بهبود می‌دهند.

۱-۲-۲ ترنسفورمر

مدل ترنسفورمر یکی از معماری‌های پیشرفته شبکه‌های عصبی است که از مکانیزم خود-توجهی^۱ بهره می‌برد تا به‌طور مؤثر وابستگی‌های موجود در دنباله‌های ورودی را شناسایی و وزن‌دهی کند. این مدل به‌ویژه در پردازش زبان طبیعی و تسک‌های پیچیده‌ای مانند ترجمه ماشینی، خلاصه‌سازی و تحلیل احساسات توانسته است موفقیت‌های چشمگیری کسب کند. در حالی که ترنسفورمرها در درک وابستگی‌های بلندمدت و ارتباطات پیچیده میان کلمات و جملات بسیار کارآمد هستند، پردازش متون طولانی برای این مدل‌ها چالش‌های خاص خود را به همراه دارد. یکی از این چالش‌ها محدودیت‌های ذاتی در اندازه ورودی است که مدل‌های از پیش آموزش داده‌شده (□□□) را مجبور می‌کند تا متون طولانی را به تکه‌های کوچکتر تقسیم کنند، که این امر ممکن است منجر به از دست رفتن اطلاعات مهم شود. علاوه بر این، پیچیدگی محاسباتی در پردازش دنباله‌های طولانی می‌تواند در کاربردهای عملی باعث محدودیت‌هایی شود. برای مقابله با این مسائل، نیاز به تکنیک‌های نوآورانه است که بتوانند پردازش متون طولانی را بهینه کرده و ساختارهای پیچیده‌تر و الگوهای زبان‌شناختی موجود در این متون را به‌طور مؤثر مدل‌سازی کنند. [۴].

۲-۲-۲ مدل‌های زبانی از پیش آموزش دیده و مدل‌های بزرگ زبانی

در سال‌های اخیر، مدل‌های زبانی پیش‌آموزش‌دیده (□□□) و مدل‌های بزرگ زبانی (□□□) پیشرفت‌های چشمگیری در زمینه پردازش زبان طبیعی داشته‌اند. مدل‌های □□□ ابتدا بر روی مجموعه‌داده‌های عظیم آموزش می‌بینند و سپس برای انجام وظایف مختلف زبان طبیعی مانند تولید متن، تحلیل احساسات و ترجمه ماشینی تنظیم می‌شوند. این مدل‌ها توانایی استخراج الگوهای پیچیده زبانی را از داده‌ها دارند و به‌طور مؤثر برای بسیاری از وظایف کاربردی پردازش زبان طبیعی استفاده می‌شوند. با ظهور معماری ترانسفورمر، مدل‌های □□□ به‌طور قابل‌توجهی بهبود یافته و قادر به پردازش وابستگی‌های بلندمدت در متون شده‌اند و عملکرد بهتری در بسیاری از کاربردها از خود نشان می‌دهند.

از سوی دیگر، مدل‌های بزرگ زبانی (□□□) مشابه مدل‌های □□□ هستند، اما دارای ویژگی‌هایی متفاوت در پردازش اطلاعات هستند. یکی از ویژگی‌های اصلی این مدل‌ها، قابلیت پردازش دامنه

^۱ Self-Attention

وسیع‌تری از متن در یک بار اجرا است که به آن‌ها اجازه می‌دهد اسناد طولانی‌تر را به‌طور مؤثرتر پردازش کنند. این ویژگی به‌ویژه برای وظایفی مانند خلاصه‌سازی متون طولانی بسیار مفید است. به‌علاوه، ها[[[توانایی تعمیم‌پذیری بالایی دارند و می‌توانند حتی با تعداد محدودی نمونه، عملکرد خوبی در بسیاری از وظایف پردازش زبان طبیعی داشته باشند.

با این حال، مدل‌های [[[همچنان با محدودیت‌هایی مواجه هستند، از جمله محدودیت در حداکثر طول ورودی و تعداد توکن‌هایی که می‌توانند در یک بار پردازش شوند. به همین دلیل، تحقیق و توسعه روش‌های کارآمد برای پردازش و خلاصه‌سازی متون طولانی با استفاده از این مدل‌ها، همچنان یک چالش تحقیقاتی فعال است. [۱][۴] [۲۲، ۴].

۳-۲-۲ روش‌های فشرده‌سازی مدل‌های بزرگ زبانی

با توجه به پیشرفت‌های سریع در مدل‌های زبانی بزرگ، (LLM) شاهد رشد چشمگیر اندازه این مدل‌ها در سال‌های اخیر هستیم. این مدل‌ها که دارای میلیاردها و حتی تریلیون پارامتر هستند، قادر به شناسایی و تولید الگوهای پیچیده در زبان طبیعی می‌باشند. با این حال، این اندازه بزرگ موجب ایجاد چالش‌های عمده‌ای در آموزش و استقرار مدل‌ها می‌شود، زیرا برای آموزش و استفاده از این مدل‌ها به منابع محاسباتی عظیم، مانند پردازنده‌های گرافیکی متعدد، نیاز است. در این راستا، فشرده‌سازی مدل‌ها به‌عنوان یک راهکار مؤثر برای کاهش این نیازهای محاسباتی و بهبود کارایی مدل‌ها مورد توجه قرار گرفته است. تکنیک‌های فشرده‌سازی مدل‌های بزرگ به سه دسته اصلی تقسیم می‌شوند: تقطیر دانش، هرس کردن و کوانتایزاسیون.

۴-۲-۲ تقطیر دانش

تقطیر دانش روشی است که به‌منظور پر کردن شکاف عملکرد بین مدل‌های بزرگ و مدل‌های کوچک‌تر استفاده می‌شود. در این فرآیند، مدل بزرگ‌تر (مدل استاد) به‌عنوان مرجعی برای آموزش یک مدل کوچک‌تر (مدل دانشجو) عمل می‌کند. مدل دانشجو سعی می‌کند تا عملکرد مدل استاد را تقلید کند. این روش علاوه بر کاهش نیازهای محاسباتی، دسترسی به مدل‌های پیشرفته‌تر را برای محققان تسهیل می‌کند و در عین حال موجب می‌شود که مدل‌های کوچک‌تر بتوانند به‌طور مؤثر و با منابع محدودتر، وظایف مشابه مدل‌های بزرگ را انجام دهند. تقطیر دانش همچنین در بهینه‌سازی مدل‌های بزرگ‌تر نیز کاربرد دارد، به‌طوری که مدل‌های استاد می‌توانند بدون کاهش عملکرد قابل توجه، به‌طور کارآمدتر اجرا

شوند.

۵-۲-۲ کوانتیزاسیون

کوانتیزاسیون یک تکنیک فشرده‌سازی است که در آن دقت داده‌های ورودی مدل کاهش می‌یابد. در این فرآیند، وزن‌ها و فعالیت‌های مدل‌های بزرگ زبانی از یک نوع داده با دقت بالا (مثلاً ۳۲ بیت) به نوع داده‌ای با دقت پایین‌تر (مانند ۸ بیت یا ۴ بیت) تبدیل می‌شوند. این کاهش دقت به مدل این امکان را می‌دهد که سریع‌تر و با نیاز به منابع محاسباتی کمتری اجرا شود، در حالی که تقریباً همان عملکرد را در پردازش داده‌ها حفظ می‌کند. این تکنیک به‌ویژه در استقرار مدل‌ها در دستگاه‌های با منابع محدود یا در پردازش‌های زمان واقعی بسیار مفید است.

۶-۲-۲ هرس کردن

هرس کردن به‌عنوان روشی برای کاهش پیچیدگی و اندازه مدل‌های یادگیری عمیق استفاده می‌شود. در این روش، وزن‌ها و پارامترهای کم‌اهمیت مدل از شبکه حذف می‌شوند، بدون آنکه دقت مدل به‌شدت کاهش یابد. این امر منجر به مدل‌های کوچک‌تر و سریع‌تر می‌شود که برای اجرا در سیستم‌های با منابع محدود یا در پردازش‌های آنالین بسیار مناسب است. هرس کردن به دو دسته اصلی تقسیم می‌شود:

هرس بدون ساختار: در این روش، وزن‌های کم‌اهمیت از شبکه حذف می‌شوند، در حالی که ساختار کلی شبکه حفظ می‌شود. این روش به‌طور معمول دقت مدل را به‌خوبی حفظ می‌کند، اما ممکن است منجر به ساختار پراکنده‌تری در شبکه شود. هرس ساختاریافته: در این روش، بخش‌های بزرگ‌تری از شبکه مانند لایه‌ها، فیلترها یا نرون‌ها از شبکه حذف می‌شوند. این روش می‌تواند به‌طور چشمگیری اندازه مدل را کاهش دهد، اما ممکن است به دقت مدل آسیب وارد کند.

این تکنیک‌ها به‌طور گسترده‌ای برای بهینه‌سازی مدل‌های بزرگ زبانی و کاهش نیازهای محاسباتی طراحی شده‌اند و می‌توانند بهبود عملکرد و کارایی این مدل‌ها را در زمینه‌های مختلف پردازش زبان طبیعی فراهم کنند.

۷-۲-۲ تنظیم دقیق بهینه‌شده پارامترها برای مدل‌های زبانی بزرگ

تنظیم دقیق کارآمد پارامترها (□□□□) روشی عملی است که به ما اجازه می‌دهد مدل‌های بزرگ زبان را به طور کارآمد برای انجام وظایف مختلف، بدون نیاز به آموزش مجدد کامل آن‌ها، آماده کنیم. این

روش با تنظیم دقیق بخشی از پارامترهای یک مدل از پیش آموزش دیده، آن را برای انجام وظایف خاص تطبیق می‌دهد. این کار به طور قابل توجهی هزینه محاسباتی و زمانی مورد نیاز برای آموزش مدل را کاهش می‌دهد. این روش به ویژه برای مدل‌های بسیار بزرگ زبانی که دارای میلیاردها پارامتر هستند بسیار مفید است، زیرا آموزش مجدد کامل این مدل‌ها نیازمند منابع محاسباتی عظیمی است [۸].

۲-۷-۲-۲ لورا

لورا^۲ روشی برای بهبود کارایی و کاهش هزینه‌های آموزش مدل‌های زبانی بزرگ (LLMs) است. این روش با حفظ ثابت بودن وزن‌های پیش‌آموزشی مدل و جایگزینی ماتریس‌های قابل آموزش با رتبه پایین در هر لایه از معماری ترانسفورمر کار می‌کند. این رویکرد هوشمندانه تعداد پارامترهای قابل آموزش را به طور چشمگیری کاهش می‌دهد، که منجر به نیاز کمتر به حافظه GPU و سرعت آموزش بالاتر می‌شود. LoRA با استفاده از جبر خطی، این بهبود کارایی را بدون افزایش زمان استفاده از مدل و حتی با حفظ کیفیت قابل مقایسه مدل به دست می‌آورد [۹].

۲-۷-۲-۲ تنظیم دقیق پرامپت

تنظیم دقیق پرامپت (Prompt Engineering) یک روش است که در آن پارامترهای اضافی به نام "پرامپت‌های نرم" به ورودی یک مدل زبانی اضافه می‌شود. این پرامپت‌های نرم نحوه تفسیر ورودی توسط مدل را تحت تأثیر قرار می‌دهند و امکان تنظیمات بدون تغییر وزن‌های مدل را فراهم می‌کنند. این روش تعادلی بین بهبود عملکرد و بهره‌وری منابع برقرار می‌کند و در مواردی که منابع محاسباتی محدود هستند یا نیاز به انعطاف‌پذیری در چندین کار وجود دارد، بسیار مفید است. از آنجایی که وزن‌های اصلی مدل بدون تغییر باقی می‌مانند، تنظیم دقیق پرامپت راهی مناسب برای تطبیق رفتار مدل بدون نیاز به بازآموزی گسترده است [۲۰].

۳-۲ توهم

توهم یا هالوسینیشن^۳ در مدل‌های زبانی بزرگ (LLMs) به پدیده‌ای اشاره دارد که در آن مدل محتوایی تولید می‌کند که با داده‌های ورودی نامرتبط، ساختگی یا ناسازگار است. این مشکل می‌تواند به ارائه

^۲LoRA

^۳Hallucination

اطلاعات نادرست منجر شود و اعتماد کاربران به این مدل‌ها را کاهش دهد. در خلاصه‌سازی متون طولانی، این مشکل بیشتر نمایان می‌شود، زیرا مدل‌ها معمولاً تمایل دارند اطلاعات بیشتری را از بخش‌های ابتدایی و انتهایی متن استخراج کنند و به جزئیات میانی متن کمتر توجه داشته باشند. این عدم توازن در پردازش متن می‌تواند باعث تولید اطلاعات نادرست یا بی‌ارتباط با منبع شود. برای مقابله با این چالش، روش‌هایی مانند استفاده از پنجره‌های همپوشانی و تکنیک‌های خودسازگاری (Self-supervised) پیشنهاد شده‌اند. این رویکردها با تقسیم متن به بخش‌های کوچک‌تر، تولید خلاصه‌های محلی، و سپس ترکیب آن‌ها، به کاهش تناقضات و افزایش دقت خلاصه‌ها کمک می‌کنند. این تکنیک‌ها نه تنها دقت و سازگاری خلاصه‌ها را بهبود می‌بخشند، بلکه موجب کاهش خطای هالوسینیشن در کاربردهای عملی می‌شوند.

فصل سوم

مرور تاریخچه

۱-۳ روش های مبتنی بر ساختار

روش های خلاصه سازی مبتنی بر ساختار از نخستین رویکردهای توسعه یافته در حوزه خلاصه سازی متون هستند. این روش ها با بهره گیری از ویژگی های ساختاری متن ورودی، به تولید خلاصه های مختصر و منسجم می پردازند. در این رویکرد، اطلاعات مهم متن به ساختاری از پیش تعریف شده تخصیص داده می شود و خلاصه بر اساس این ساختار ایجاد می گردد. با پیشرفت فناوری و ظهور روش های نوین، مانند مدل های زبانی پیشرفته و تکنیک های یادگیری عمیق، کارایی و دقت در خلاصه سازی متون بهبود یافته است. با این حال، روش های مبتنی بر ساختار همچنان به عنوان پایه و اساس درک و پردازش متون مورد استفاده قرار می گیرند. در این بخش روش های مبتنی بر درخت^۱، مبتنی بر قالب^۲، مبتنی بر هستان شناسی^۳، عبارت مقدمه و بدنه^۴، مبتنی بر گراف^۵ و مبتنی بر قانون^۶ مورد بررسی قرار می گیرد.

۱-۱-۳ روش مبتنی بر درخت

روش های مبتنی بر درخت در خلاصه سازی متن از درخت های وابستگی برای نمایش ساختار نحوی اسناد متنی استفاده می کنند. ابتدا، متن مبدأ به درخت های وابستگی تبدیل می شود و سپس این درخت ها در یک ساختار واحد ادغام می شوند. در نهایت، با خطی سازی درخت ادغام شده، جملات جدیدی تولید می شوند. این فرآیند، که به آن «خطی سازی درخت» گفته می شود، به انتخاب تجزیه کننده و حفظ وابستگی های نحوی بین کلمات وابسته است که می تواند بر کارایی تأثیر بگذارد [۳۳].

۲-۱-۳ روش مبتنی بر قالب

این روش ها از قالب های از پیش تعریف شده برای نمایش اسناد استفاده می کنند. این قالب ها برای انطباق با الگوها و قوانین خاص در محتوای متنی طراحی شده اند و امکان استخراج اطلاعات مرتبط را فراهم می کنند که می توان آن ها را در چارچوب این قالب ها ترسیم کرد. این فرآیند شامل تطبیق متن با الگوها و قوانین مذکور برای شناسایی محتوای متناسب با الگو است. این روش به دلیل تولید خلاصه هایی که به ساختار و قالب های تعیین شده پایبند هستند، از انسجام بالایی برخوردار است. [۳۳].

¹tree-based

²template-based

³ontology-based

⁴lead-and-body phrase

⁵graph-based

⁶rule-based

۳-۱-۳ روش مبتنی بر هستان‌شناسی

روش‌های مبتنی بر هستی‌شناسی در خلاصه‌سازی متن از پایگاه‌های دانش برای بهبود فرآیند خلاصه‌سازی استفاده می‌کنند. بسیاری از اسناد اینترنتی به حوزه‌های خاص با واژگان محدود مرتبط هستند که می‌توان آن‌ها را با هستی‌شناسی‌ها بهتر نمایش داد. هستی‌شناسی‌ها نام‌گذاری و تعریف رسمی انواع موجودیت‌های یک دامنه خاص را ارائه می‌دهند و به‌عنوان پایگاه دانش عمل می‌کنند. با استفاده از هستی‌شناسی، سیستم خلاصه‌سازی می‌تواند نمایش معنایی محتوای اطلاعات را بهبود بخشد. تکنیک‌های این روش شامل ساخت مدل معنایی با هستی‌شناسی، نگاشت جملات به گره‌های آن و محاسبه امتیاز هر موجودیت برای رتبه‌بندی جملات است. [۳۳]. لی و همکاران یک سیستم فازی را ارائه کرد که از هستی‌شناسی طراحی شده توسط متخصص حوزه اخبار استفاده می‌کند. جملات بر اساس طبقه‌بندی کننده اصطلاحی مبتنی بر هستی‌شناسی طبقه‌بندی می‌شوند. مکانیزم استنتاج فازی درجه عضویت برای هر جمله را با توجه به طبقه‌بندی کننده محاسبه می‌کند [۱۹].

۳-۱-۴ روش عبارت مقدمه و بدنه

روش «عبارت مقدمه و بدنه» در خلاصه‌سازی متن بر شناسایی و بازنگری جملات اصلی، معروف به جملات کلیدی، در یک سند تمرکز دارد. این جملات معمولاً حاوی اطلاعات مفید هستند و خلاصه‌ای جامع از محتوا ارائه می‌دهند. این روش شامل درج و جایگزینی عبارات در جملات اصلی برای ایجاد تکرار مناسب با بازبینی‌های معنایی است. از محدودیت‌های این روش می‌توان به نبود مدل تعمیم‌یافته برای خلاصه‌سازی و تأثیر منفی مدل‌های تجزیه دستوری اشاره کرد. [۳۳]. ایشیکاوا و همکاران روش خلاصه‌سازی ترکیبی مبتنی بر روش فرکانس تکرار عبارت^۷ و عبارت مقدمه و بدنه پیشنهاد کردند. تابع توزیع زاویه‌ای ضربدر بسامد عبارت میزان اهمیت هر جمله را مشخص می‌کند. دستورهای براساس اهمیت برای نوشتن خلاصه رتبه‌بندی می‌شوند [۱۰].

۳-۱-۵ روش مبتنی بر گراف

یکی دیگر از رویکردهای خلاصه‌سازی، روش مبتنی بر گراف است که در آن هر جمله‌ی سند به‌عنوان یک گره در گراف نمایش داده می‌شود. جملات بر اساس روابط معنایی با یکدیگر متصل می‌شوند و وزن یال‌ها نشان‌دهنده قدرت این روابط است. سپس، با استفاده از الگوریتم‌های رتبه‌بندی گراف، اهمیت

⁷Term frequency (TF)

هر جمله تعیین می‌شود و جملات با اهمیت بالاتر در خلاصه گنجانده می‌شوند. این روش بدون نیاز به دانش عمیق زبانی یا حوزه‌ای، می‌تواند با انتخاب جملات مهم، خلاصه‌های مختصر و منسجمی ایجاد کند. [۳۳]. مالیروس و اسکینیس از مرکزیت گره برای نشان دادن اهمیت یک اصطلاح در سند استفاده می‌کنند. مرکزیت‌های گره محلی و جهانی برای وزن‌دهی عبارت در نظر گرفته می‌شوند تا خلاصه را شکل دهند [۲۸].

۳-۱-۶ روش مبتنی بر قانون

در روش خلاصه‌سازی مبتنی بر قاعده، اسناد به دسته‌ها و جنبه‌های مختلف تقسیم می‌شوند. سپس، ماثول انتخاب محتوا بر اساس قوانین از پیش تعریف‌شده، اطلاعات بهینه را برای هر جنبه انتخاب می‌کند. در نهایت، با استفاده از الگوهای تولید، جملات خلاصه و مختصر ایجاد می‌شوند. به عبارت دیگر، این روش با بهره‌گیری از قوانین مشخص، مهم‌ترین اطلاعات مرتبط با هر جنبه را انتخاب کرده و سپس آن‌ها را در قالب یک خلاصه منسجم ارائه می‌دهد [۳۱].

روش‌های سنتی خلاصه‌سازی متن، هرچند در زمان خود مؤثر بوده‌اند، اما در مقایسه با شبکه‌های عصبی مدرن کارایی کمتری دارند. این روش‌ها غالباً به دانش زبانی عمیق و قوانین از پیش تعریف‌شده متکی هستند که ممکن است در مواجهه با متون متنوع و پیچیده ناکارآمد باشند.

۳-۲ روش‌های مبتنی بر مدل ترنسفورمر

با ظهور ترنسفورمرها^۸، بهبودهای قابل توجهی در کیفیت نتایج خلاصه‌سازی خودکار به وجود آمد. ترنسفورمرها با استفاده از مکانیزم توجه به خود^۹ شباهت بین ورودی‌ها را بدون توجه به موقعیت موازی آن‌ها با حضور مستقل هر توکن در توالی ورودی مدل می‌کنند و به طور مؤثر مشکلات شبکه‌های بازگشتی را حل می‌کنند [۴۳]. یکی از جهت‌گیری‌های رایج پژوهشی، اصلاح یا تطبیق ترنسفورمرها و مدل‌های زبانی از پیش آموزش دیده با وظایف مختلف مانند خلاصه‌سازی است. مدل‌های مبتنی بر مدل‌های زبانی از پیش آموزش دیده که با هدف خلاصه‌سازی انتزاعی طراحی شده‌اند از ویژگی‌های معنایی و متنی غنی بازنمایی‌های زبان برای بهبود کیفیت و دقت خلاصه‌ها استفاده می‌کنند.

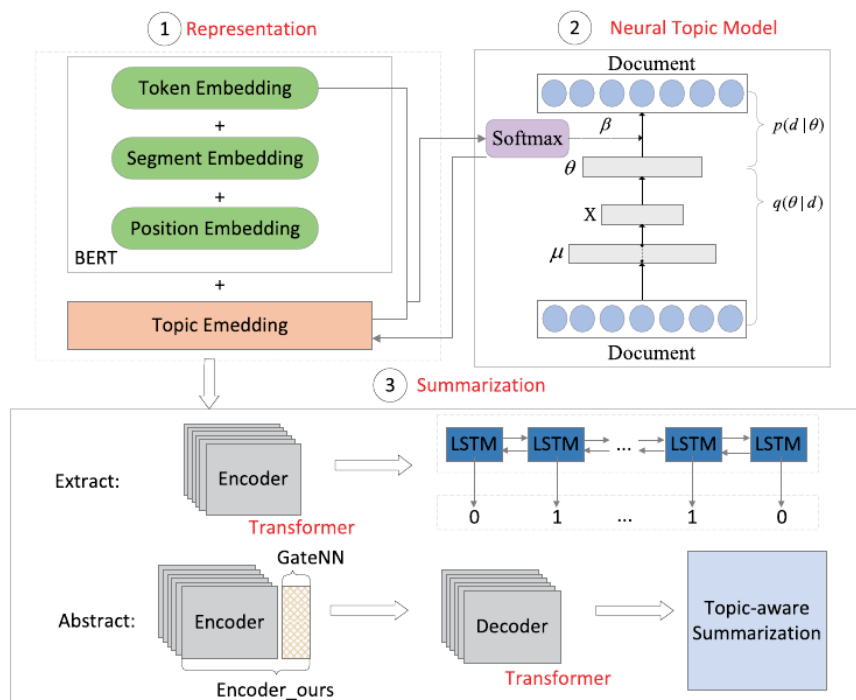
پان^{۱۰} و همکاران یک مدل خلاصه‌سازی بر اساس مدل برت را پیشنهاد کرده‌اند. نویسندگان استدلال

^۸transformers

^۹Self-attention

^{۱۰}Pan

می‌کنند که خلاصه‌های تولید شده توسط مدل‌های خلاصه‌سازی متن موجود که موضوع متن را در نظر نمی‌گیرند، مرتبط یا حاوی اطلاعات مفید نیستند. مدل ارائه شده که تی‌برت‌سام^{۱۱} نامیده می‌شود از سه بخش ایجاد بازنمایی، مدل موضوعی عصبی^{۱۲} و مدل خلاصه‌سازی تشکیل شده است. ساختار مدل را در شکل ۱-۳ نشان داده شده است.



شکل ۱-۳: معماری مدل تی‌برت‌سام [۲۷]

همانطور که در شکل ۲-۳ نشان داده شده است، بازنمایی ایجاد شده برای هر جمله ورودی، با استفاده از یک شبکه‌ی ترنسفورمر دوسویه‌ی^{۱۳} چند لایه و حاصل جمع چهار نوع تعبیه (تعبیه نشانه^{۱۴}، تعبیه قطعه^{۱۵}، تعبیه موقعیت و تعبیه موضوع) به دست می‌آید که تعبیه موضوع در این مقاله معرفی شده و سه تعبیه دیگر مشابه مدل بERT هستند. وجود تعبیه موضوع در تولید بازنمایی هر کلمه یا جمله موجب افزودن اطلاعات پیش زمینه‌ای به هر کلمه و حل مشکل چند معنایی می‌شود.

مدل موضوعی عصبی وظیفه‌ی ایجاد تعبیه موضوعی را دارد. این مدل دارای دو جزء است: یک

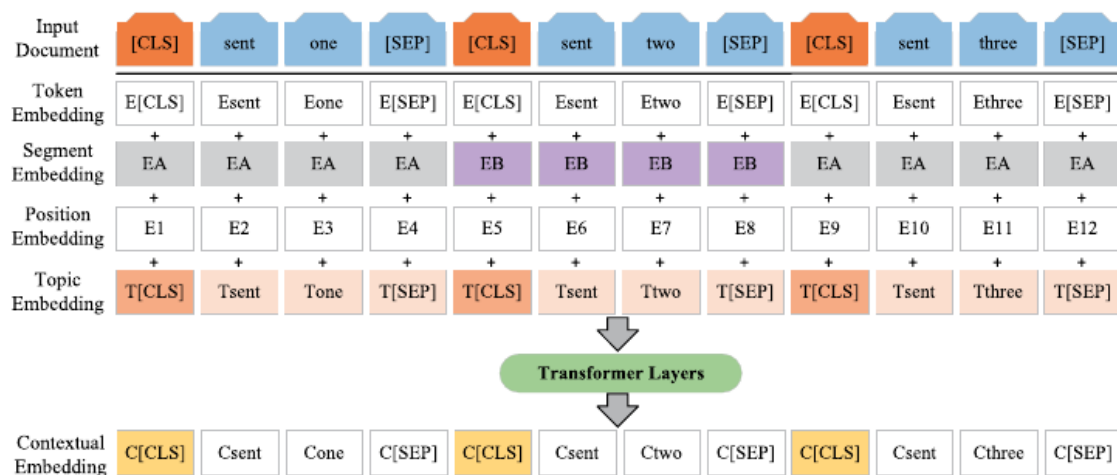
¹¹T-BERTSum

¹²Neural Topic Model (NTM)

¹³bidirectional

¹⁴token embedding

¹⁵segment embedding

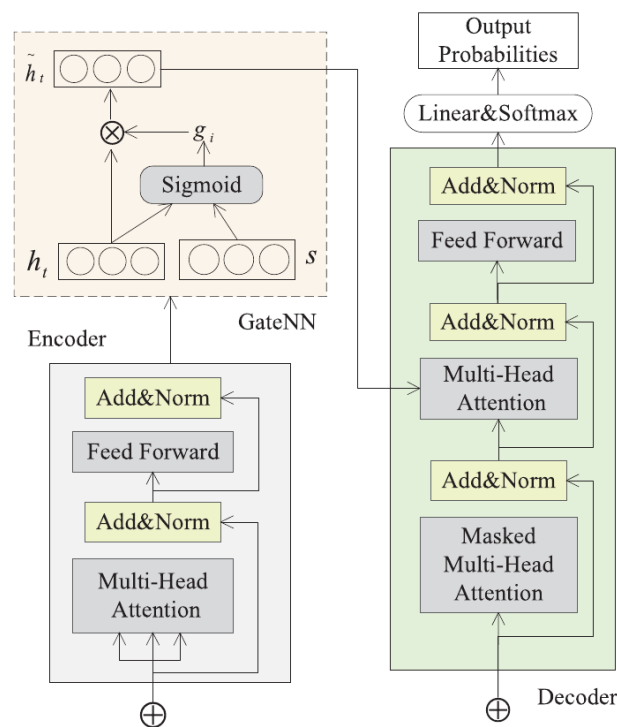


شکل ۳-۲: تعبیه مدل تی‌برت‌سام [۲۷]

شبکه مولد و یک شبکه استنتاج. شبکه مولد یک سند را به عنوان ورودی می‌گیرد و یک توزیع موضوعی را بر روی کلمات موجود در سند خروجی می‌دهد. شبکه استنتاج یک سند را به عنوان ورودی می‌گیرد و خروجی آن پارامترهای توزیع موضوع است. بخش خلاصه‌سازی مدل مبتنی بر معماری کدگذار - کدگشای ترنسفورمر است. کدگذار بازنمایی ایجاد شده را به عنوان ورودی می‌گیرد و دنباله‌ای از حالت‌های پنهان را تولید می‌کند. سپس کدگشا با استفاده از حالت‌های پنهان و متن خلاصه را تولید می‌کند. همانطور که در شکل ۳-۳ نشان داده شده است، به منظور فیلتر کردن اطلاعات کلیدی توالی ورودی، شبکه دروازه‌ای قبل از کدگشا اضافه می‌شود. این شبکه برای کنترل جریان اطلاعات از دنباله ورودی به دنباله خروجی افزوده شده است و باعث می‌شود کدگشا بر روی تولید خلاصه از اطلاعات کلیدی و حذف اطلاعات غیرضروری تمرکز کند. این مدل می‌تواند خلاصه‌هایی تولید کند که مرتبط با موضوع متن و حاوی اطلاعات مفید باشد و قابلیت تطبیق با حوزه‌های مختلف را دارد.

اکثر مدل‌های خلاصه‌سازی انتزاعی موجود برای تولید خلاصه‌های با طول ثابت طراحی شده‌اند، بنابراین سو^{۱۶} و همکاران یک مدل دو مرحله‌ای مبتنی بر ترنسفورمر ارائه دادند که خلاصه‌های انتزاعی با طول متغیر را با توجه به تقاضای کاربر تولید کند. مطابق شکل ۳-۴ مدل پیشنهادی با تقسیم متن ورودی به بخش‌ها، استخراج اطلاعات کلیدی و تولید خلاصه‌ی هر بخش، خلاصه‌ی انتزاعی با طول متغیر تولید می‌کند [۴۲]. بخش‌های مدل ارائه شده به شرح زیر است.

¹⁶Ming-Hsiang Su



شکل ۳-۳: معماری ترنسفورمر تی‌برت‌سام [۲۷]

این مدل شامل شبکه‌ی دروازه‌ای و کدگذار-کدگشا با توجه چند سر می‌باشد [۲۷]

- بخش‌بندی متن: این مرحله متن ورودی را به تعدادی قسمت از پیش تعیین شده تقسیم می‌کند. تعداد بخش‌ها را می‌تواند توسط کاربر مشخص شود یا با توجه به نسبت دلخواه طول ورودی تنظیم کرد. برای شناسایی مرزهای بین بخش‌ها از مدل *BERT - biLSTM* استفاده می‌شود. این مرحله تضمین می‌کند که مرحله خلاصه‌سازی انتزاعی بر روی بخش‌های منسجم متن انجام می‌شود. هدف این بخش یافتن نقاط تقسیم‌بندی است که نشان دهنده تغییر موضوع در متن است و به بهبود کیفیت خلاصه‌های تولید شده کمک می‌کند.
- خلاصه‌سازی استخراجی: پس از تقسیم‌بندی متن، با استفاده از یک مدل خلاصه‌سازی استخراجی مبتنی بر برت‌سام^{۱۷} مهم‌ترین جمله را از هر بخش استخراج می‌شود.
- خلاصه‌سازی اسناد: با استفاده از جملات استخراج شده این مازول خلاصه سرفصل سند ورودی را تولید می‌کند که این خلاصه به عنوان خروجی هدف در مرحله آموزش مدل دو مرحله‌ای استفاده می‌شود. مدل ترنسفورمر سند به حل مشکل تغییر طول ورودی و خروجی در کار خلاصه‌سازی

¹⁷BertSum

کمک می‌کند.

- خلاصه‌سازی بخش: این ماژول وظیفه‌ی تولید خلاصه برای بخش‌های به دست آمده از مرحله تقسیم‌بندی متن را دارد.

- آموزش مشارکتی: برای آموزش متناوب ماژول خلاصه‌سازی بخش و ماژول خلاصه‌سازی اسناد تا زمان همگرایی آموزش مشارکتی اعمال می‌شود. این فرآیند به بهینه سازی عملکرد هر دو ماژول کمک می‌کند.

- ایجاد خلاصه با طول متغیر: پس از اینکه متن ورودی به بخش‌های مختلف تقسیم شد، هر بخش از ماژول خلاصه‌سازی بخش عبور می‌کند تا یک خلاصه انتزاعی مبتنی بر جمله ایجاد کند. سپس این خلاصه‌های مبتنی بر جمله به هم متصل می‌شوند تا خلاصه انتزاعی با طول متغیر را تشکیل دهند. این فرآیند الحاق تضمین می‌کند که خلاصه تولید شده شامل اطلاعات تمام بخش‌های متن ورودی است.

با ترکیب روش‌های استخراجی و انتزاعی در مدل خلاصه‌سازی دو مرحله‌ای، رویکرد پیشنهادی می‌تواند خلاصه‌های انتزاعی روان و با طول متغیر را با توجه به خواسته‌های کاربر تولید کند [۴۲].

لوئیس و همکاران مدلی با نام بارت^{۱۸} ارائه دادند. این مدل مشابه با مدل اصلی ترنسفورمر، ساختاری کدگذار-کدگشا دارد. بر خلاف سادگی، به دلیل داشتن کدگذار دو طرفه و کدگشای چپ به راست این مدل را می‌توان نسخه عمومی‌تری از برت و جی‌پی‌تی^{۱۹} دانست. بارت در عملیات تولید متن، مانند ترجمه ماشینی یا خلاصه‌سازی انتزاعی متن، و همچنین در فهم متن کاربرد دارد و با استفاده از اهداف کدگذاری خودکار حذف نویز آموزش می‌بیند. برای پیش‌آموزش بارت نواقصی در سندهای ورودی ایجاد می‌شود و با بهینه کردن تابع زیان آنتروپی-مقاطع^{۲۰} بین خروجی‌های کدگشا و سند اولیه، متن بازسازی می‌شود. همانطور که در شکل ۳-۵ نشان داده شده است این مدل طیف گسترده‌ای از نویزها از جمله پوشاندن توکن، حذف توکن، پر کردن متن، چرخش سند، به هم ریختن جمله (به هم زدن تصادفی ترتیب کلمه یک جمله) را استفاده می‌کند [۲۱].

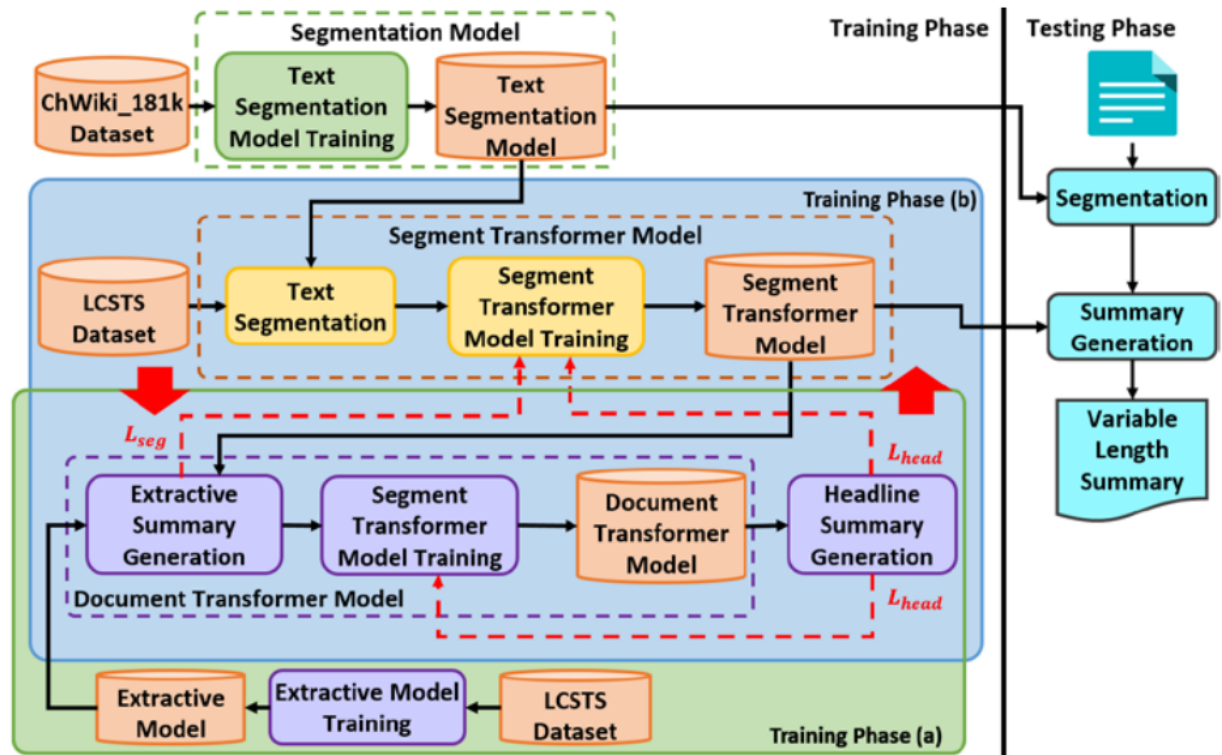
با این که بارت دقت خلاصه‌سازی انتزاعی متن را بهبود بخشید، ولی مراحل پیش‌آموزش آن، مختص خلاصه‌سازی انتزاعی متن نیستند، در نتیجه در سال ۲۰۲۰ مدلی تحت عنوان پگاسوس^{۲۱} توسط ژنگ

¹⁸BART

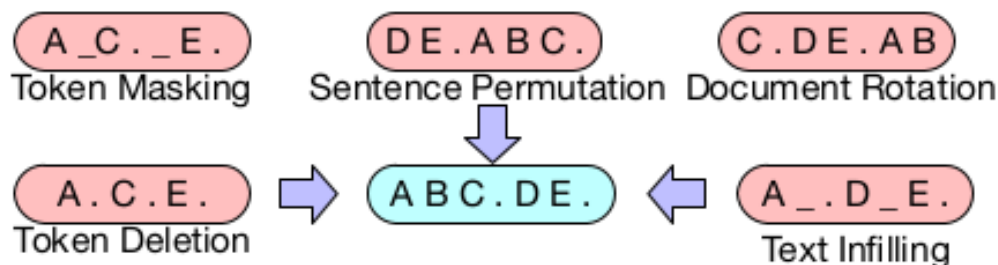
¹⁹GPT

²⁰cross-entropy

²¹PEGASUS



شکل ۳-۴: چهارچوب ایجاد خلاصه با طول متغیر [۴۲]



شکل ۳-۵: عمل‌های پیش‌آموزش بارت [۲۱]

و همکاران ارائه شد که معماری مشابه با بارت داشت ولی پیش‌آموزش آن مختص خلاصه‌سازی انتزاعی متن بود. مدل پگاسوس یک مدل دنباله به دنباله کدگذار کدگشا مبتنی بر ترنسفورمر است که بر روی مجموعه‌های متنی بدون نظارت با هدف تولید جملات فاصله‌افتاده^{۲۲} از قبل آموزش داده شده است [۴۹]. این مدل دو روش پیش‌آموزش را معرفی کرده است که در ادامه به شرح آنها می‌پردازیم:

۱. تولید جملات فاصله‌افتاده: فرضی مطرح شده است که اگر عمل پیش‌آموزش مدل به وظایف

^{۲۲}gap sentences generation

پایین دست^{۲۳} نزدیک تر باشد، نتیجه نهایی بهتر و همچنین تنظیم دقیق پارامترها^{۲۴} سریع تر خواهد بود. با توجه به این که این مدل قرار است فقط برای خلاصه سازی انتزاعی متن استفاده شود، عمل پیش آموزش مشابه تولید متن های خلاصه از یک سند ورودی تعریف شده است. تعدادی از جملات انتخاب شده و هر جمله به طور کامل با توکن $[MASK1]$ جایگزین میشود. برای انتخاب این جملات، سه راه پیشنهاد شده است.

- انتخاب تصادفی: m جمله به صورت تصادفی از متن انتخاب شده و پنهان می شوند.
- انتخاب جملات اول متن: m جمله اول متن پنهان می شوند زیرا اغلب جملات ابتدای متن نسبت به جملات بعدی مهم تر هستند.
- انتخاب جملات مهم متن: برای انتخاب m جمله مهم متن از تقریب معیار ارزیابی روژ-۱ استفاده می شود. به ازای هر جمله از متن، یک دوتایی از آن جمله و متن سند فاقد آن جمله ساخته شده و ارزیابی می شود که چقدر ممکن است این جمله، خلاصه سند فاقد آن جمله باشد. جملاتی که امتیاز بالاتر گرفته اند از نظر خلاصه بودن مهم تر هستند و پنهان می شوند.

۲. مدل زبانی پوشیده شده: مشابه مدل برت ۵۱ درصد از توکن های متن ورودی انتخاب می شوند و سپس ۸۰ درصد از این توکن ها، با توکن $[MASK2]$ و ۱۰ درصد توکن ها با یک توکن تصادفی جایگزین می شوند. ۱۰ درصد دیگر بدون تغییر باقی می ماند. شکل ۳-۶ اعمال همزمان این دو عمل، یعنی تولید جمالت فاصله افتاده و مدل زبانی پوشیده شده را بر روی یک ورودی نشان می دهد.

کدیا^{۲۵} و همکاران الگوریتم حداکثر سازی نقطه-محصول فرایادگیری (امدات)^{۲۶} را برای بهبود پگاسوس پیشنهاد دادند. این الگوریتم بر اساس ایده به حداکثر رساندن حاصل ضرب نقطه ای بین گرادین های مدل در نقاط مختلف آموزش با استفاده از تکنیکی به نام تفاوت های محدود^{۲۷} است. این الگوریتم از نظر محاسباتی کارآمد است و می تواند برای مدل های بزرگ مانند برت اعمال شود و سربار

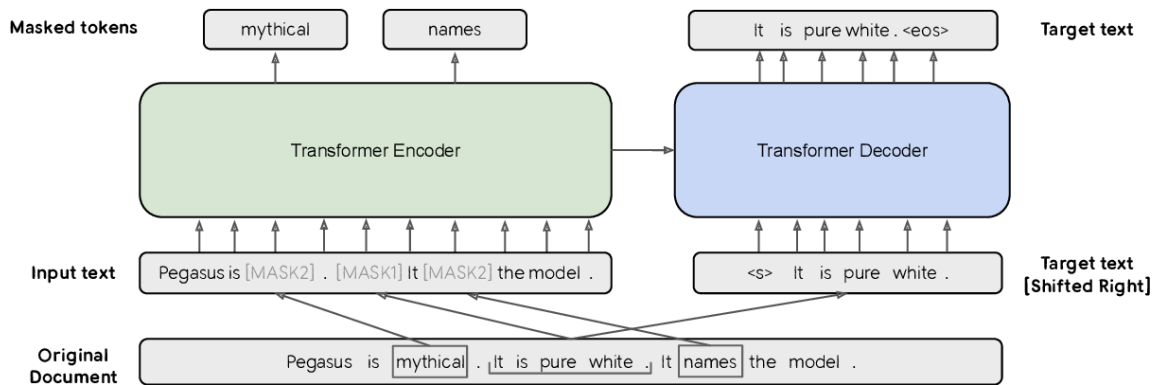
²³downstream task

²⁴fine-tuning

²⁵Kedia

²⁶Meta-Learned Dot-Product Maximization (MDot)

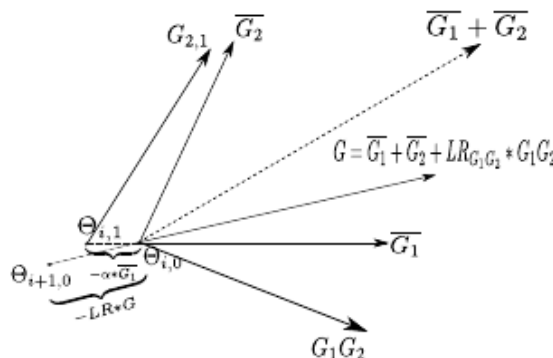
²⁷finite differences



شکل ۳-۶: ساختار مدل پگاسوس [۴۹]

معماری پایه پگاسوس یک کدگذار-کدگشا ترنسفورمر استاندارد است. جملات فاصله‌افتاده و مدل زبانی پوشیده شده به طور همزمان در این مثال به عنوان اهداف پیش آموزش اعمال می‌شوند. در اصل سه جمله وجود دارد. یک جمله با [MASK1] پوشانده شده و به عنوان متن تولید هدف جملات فاصله‌افتاده استفاده می‌شود. دو جمله دیگر در ورودی باقی می‌مانند و برخی از نشانه‌ها به طور تصادفی توسط [MASK2] پوشانده می‌شوند [۴۹].

محاسباتی را کاهش بدهد [۳۹]. عملکرد مناسب مدل پگاسوس در خلاصه‌سازی متون باعث شده بهترین مدل خلاصه‌سازی متون کوتاه مبتنی بر مدل پگاسوس و تکنیک تنظیم^{۲۸} امادات باشد.



شکل ۳-۷: الگوریتم امادات [۳۹]

محاسبه گرادیان برای به حداکثر رساندن محصول نقطه‌ای با استفاده از تقریب تفاضل محدود و استفاده از آن برای تنظیم گرادیان استاندارد [۳۹].

²⁸regularization

۱-۲-۳ ایده‌های ارائه شده بهبود خلاصه‌سازی متون طولانی

یکی از مشکلات مدل ترنسفورمر در خلاصه‌سازی متون طولانی، حافظه‌ی درجه دوم، پیچیدگی‌های محاسباتی و تعداد زیاد عملیات می‌باشد. برای حل این چالش‌ها ایده‌های مختلفی ارائه شده است. به عنوان مثال شبکه‌ی ریفورمر^{۲۹} برای حل چالش‌های محاسباتی مرتبط با پردازش دنباله‌های طولانی متن ارائه شده است. لایه‌های برگشت‌پذیر^{۳۰} معرفی شده در این مقاله امکان بازسازی ورودی از خروجی را در طول گذر به عقب را فراهم می‌کنند که موجب کاهش نیازهای حافظه و امکان پردازش کارآمد دنباله‌های طولانی می‌شود. علاوه بر این، ریفورمر از تکه تکه کردن برای پردازش بخش‌های کوچک‌تر ورودی به طور مستقل استفاده می‌کند که موازی‌سازی را ممکن می‌کند و مصرف حافظه را کاهش می‌دهد. همچنین استفاده از درهم‌سازی حساس به مکان^{۳۱} در مکانیسم توجه منجر به محاسبه توجه کارآمدتر می‌شود. درهم‌سازی حساس به مکان با توجه به زیرمجموعه‌ای از نشانه‌ها بر اساس مقادیر هش آنها، محاسبه توجه کامل را تقریب می‌زند. علاوه بر این، ریفورمر از کدگذاری‌های موقعیت محوری برای کدگذاری اطلاعات موقعیت توکن‌ها به صورت فشرده استفاده می‌کند. این تکنیک‌ها مجموعاً مدل ریفورمر را مقیاس‌پذیر می‌سازد، و آن را قادر می‌سازد تا دنباله‌های طولانی متن را مدیریت کند و در عین حال عملکرد رقابتی را در وظایف مختلف پردازش زبان طبیعی حفظ کند [۱۴].

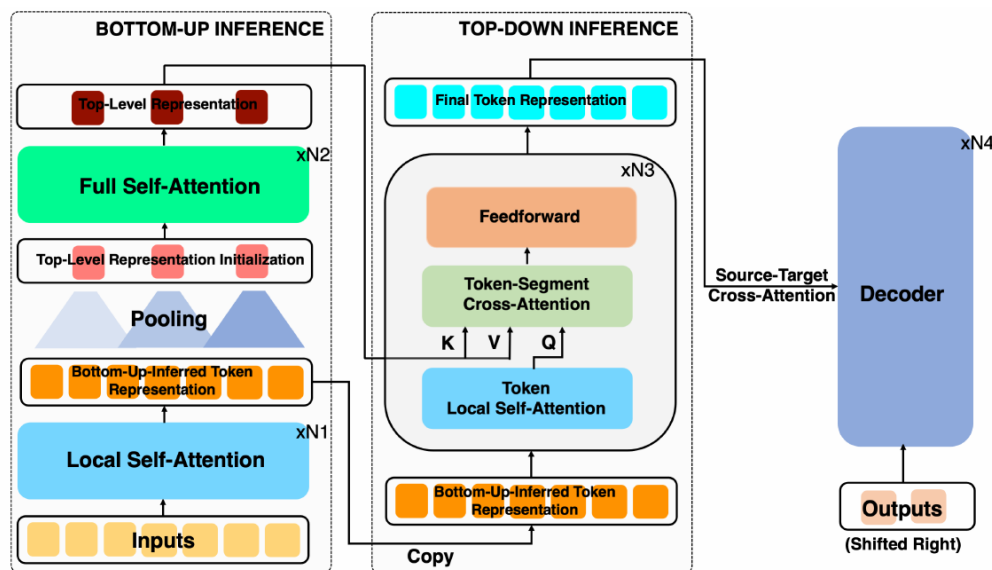
شبکه‌ی ترنسفورمر پراکنده^{۳۲} با معرفی فاکتورسازی ماتریس پراکنده‌ی توجه، زمان و حافظه مورد نیاز را به کاهش می‌دهد. با استفاده از پراکندگی، مدل می‌تواند تنها به زیرمجموعه‌ای از نشانه‌های ورودی توجه کند و روی مرتبط‌ترین اطلاعات تمرکز کند و بقیه را نادیده بگیرد. این رویکرد پیچیدگی محاسباتی را کاهش می‌دهد و مدل می‌تواند توالی‌های طولانی را مدیریت کند [۳]. مشابه شبکه‌ی ترنسفورمر پراکنده مدل بیگ‌برد که^{۳۳} با استفاده از مکانیزم توجه پراکنده^{۳۴} عملکرد ترنسفورمر را در مواجهه با دنباله‌ی کلمات طولانی بهبود می‌بخشد، نوآوری‌های دیگری مانند توجه جامع^{۳۵} را معرفی می‌کند. در این مدل توکن‌های خاص به تمام توکن‌های دیگر در دنباله توجه می‌کنند و وابستگی‌های دوربرد را بهتر از سایر روش‌ها به دست می‌آورند. همچنین فرآیند پالایش تکراری وزن‌های توجه را برای بهبود عملکرد مدل اصلاح می‌کند [۴۸].

²⁹Reformer³⁰reversible layers³¹locality-sensitive hashing (LSH)³²sparse³³Big Bird³⁴Sparse attention³⁵global attention

در سال‌های اخیر ایده‌های مختلفی برای بهبود کیفیت خروجی مدل خلاصه‌سازی خودکار اسناد بلند ارائه شده است. به عنوان مثال پایل^{۳۶} و همکاران که برای بهبود خلاصه انتزاعی نهایی متون طولانی از رویکرد ترکیبی استخراجی-انتزاعی با استفاده از مدل زبانی از پیش آموزش دیده جی‌پی‌تی-دو^{۳۷} استفاده می‌کنند. در این مدل مرحله استخراج ساده قبل از تولید خلاصه انجام می‌شود، سپس برای شرطی کردن مدل زبانی ترنسفورمر بر روی اطلاعات مربوط قبل از تولید خلاصه استفاده می‌شود. این رویکرد در مقایسه با کارهای قبلی که از مکانیزم کپی استفاده می‌کنند، خلاصه‌های انتزاعی بیشتری تولید می‌کند [۳۶]. پانگ^{۳۸} و همکاران یک ساختار سلسله مراتبی برای اسناد طولانی فرض کرده‌اند. در این ساختار سطح بالا بر وابستگی دوربرد تمرکز می‌کند و سطح پایین جزئیات را حفظ می‌کند. در استنتاج از پایین به بالا، تعبیه‌های متنی نشانه‌ها با استفاده از توجه محلی محاسبه می‌شوند و برای دریافت وابستگی‌های دوربرد و زمینه جامع، استنتاج از بالا به پایین برای نمایش‌های توکن اعمال می‌شود. یک ساختار پنهان چند مقیاسی دو سطحی استفاده می‌شود، که در آن سطح پایین شامل نمایش‌های نشانه‌ای است که توسط استنتاج پایین به بالا محاسبه می‌شود، سپس با اعمال مکانیزم توجه به سطوح بزرگ‌تر روابط بین بخش‌های مختلف سند را بدست می‌آورد. ساختار مدل را در شکل ۳-۸ نشان داده شده است. روش پیشنهادی یک رویکرد جدید امیدوارکننده برای خلاصه‌سازی اسناد طولانی است و نسبت به روش‌های قبلی کارآمدتر و موثرتر است [۳۴].

جیدیوتیس و همکاران شیوهی تقسیم و غلبه (دنسر)^{۳۹} را برای بهبود خلاصه‌سازی اسناد طولانی پیشنهاد کرده‌اند. این روش به طور خودکار خلاصه یک سند را به چند بخش تقسیم می‌کند و هر یک از این بخش‌ها را به بخش مناسب سند جفت می‌کند تا خلاصه‌های هدف متمایز ایجاد کند. شیوهی معرفی شده در نظر می‌گیرد که متون طولانی به صورت بخش‌های گسسته ساختار بندی شده‌اند. برای مطابقت هر قسمت از خلاصه با بخشی از سند در دنسر از معیار روژ^{۴۰} استفاده می‌شود. در این روش معیار روژ-ال بین هر یک از جملات خلاصه و تمام جملات سند محاسبه می‌شود و هر جمله‌ی خلاصه هدف به بخش حاوی جمله با بیشترین روژ-ال نسبت داده می‌شود. سپس تمام جملات خلاصه‌ی هدف مربوط به هر بخش را به هم الحاق می‌کنیم تا خلاصه‌ی هدف برای هر بخش ایجاد شود. در طول آموزش هر بخش از سند به همراه جمله‌ی خلاصه‌ی مربوط به آن به عنوان متن ورودی و خلاصه‌ی هدف

³⁶Pilault³⁷GPT-2³⁸Pang³⁹Divide-and-Conquer (DANCER)⁴⁰ROUGE



شکل ۳-۸: معماری مدل ترنسفورمر از بالا به پایین [۳۴]

استفاده می‌شود. مزایای این روش آموزش:

۱. تقسیم مساله به چند زیر مساله باعث کاهش پیچیدگی و ساده‌سازی مساله می‌شود.
۲. انتخاب خلاصه‌های هدف برای هر بخش بر اساس امتیازات روژ-ال هر جمله باعث تطابق بهتر و متمرکزتر بین دنباله‌های منبع و هدف ایجاد می‌شود.
۳. تقسیم هر سند آموزشی به چند جفت ورودی-هدف، نمونه‌های آموزشی بسیار بیشتری ایجاد می‌کند. این کار برای مدل‌های خلاصه‌سازی عصبی مفید است.
۴. این روش می‌تواند از مدل‌های خلاصه‌سازی مختلف از جمله شبکه‌ی عصبی بازگشتی و ترنسفورمرها استفاده کند.

هنگام کار با اسناد ساختاریافته طولانی، معمولاً همه بخش‌های سند کلیدی برای سند نیستند. اگر یک مقاله آکادمیک را به عنوان مثال در نظر بگیریم، بخش‌هایی مانند مرور ادبیات یا پیشینه در تلاش برای خلاصه کردن نکات اصلی مقاله ضروری نیستند و باعث افزودن نویز می‌شوند. بنابراین از بخش مرور ادبیات صرف نظر می‌شود و تمرکز سیستم خلاصه‌سازی فقط روی بخش‌های مقدمه، روش‌ها، نتایج و نتیجه‌گیری می‌باشد.

این مدل قابل ترکیب با پگاسوس یا مدل مولد نقطه‌ای^{۴۱} می‌باشد. بخش کدگشا مدل مولد نقطه‌ای با ایجاد جملات تکراری مقابله می‌کند. هرچند ممکن است به خاطر تکرار اطلاعات در بخش‌های مختلف بازهم خلاصه‌ی تکراری ایجاد شود.

شیونگ و همکاران با اصلاح هدف بهینه‌سازی، معماری مدل‌های از پیش آموزش دیده و مجموعه‌ی دادگان پیش‌آموزش^{۴۲} روشی را برای ساخت مدل‌های مناسب متون طولانی پیشنهاد می‌کنند. مدل‌های پیش‌آموزش دیده متن به متن، مانند برت و بارت، معمولاً بر روی دنباله‌های متن کوتاه، مانند جملات یا پاراگراف‌ها آموزش داده می‌شوند. در حالی که بسیاری از وظایف پردازش زبان طبیعی، مانند پاسخگویی به سؤال و خلاصه کردن، به توانایی پردازش توالی متن طولانی نیاز دارند این مقاله تعدادی از تکنیک‌ها را برای تطبیق مدل‌های متن به متن از پیش آموزش دیده برای دنباله‌های متن طولانی پیشنهاد می‌کند. این تکنیک‌ها عبارتند از:

- ارائه‌ی مدل براساس یک ترنسفورمر با مکانیزم توجه به خود پراکنده‌ی بلوکی^{۴۳} در قسمت کدگذار است. این مکانیزم امکان استفاده‌ی مجدد از وزن‌های مدل‌های از پیش آموزش دیده را فراهم می‌کند.
- مکانیزم توکن سراسری^{۴۴}: در این مکانیزم یک مجموعه‌ی کوچک از توکن‌های سراسری به کل توالی توجه می‌کنند و امکان تعاملات دوربرد در کدگذار فراهم می‌شود.
- هم‌پوشانی بلوک‌های توجه^{۴۵}: توجه لغزشی با هم‌پوشانی یک راه ساده برای معرفی اتصالات دوربرد در مدل‌های توجه محلی است. در این رویکرد، توکن‌های درون هر بلوک به تمام توکن‌های درون خود بلوک و همچنین نیمی از توکن‌های بلوک‌های چپ و راست مجاور نزدیک می‌شوند. این نسخه بلوکی از پنجره‌های توجه هم‌پوشانی، راه ساده‌تر و کارآمدتری را برای معرفی اتصالات دوربرد ارائه می‌کند و در عین حال موازی‌سازی را در پیاده‌سازی مدل تسهیل می‌کند.
- لایه‌ی خود توجه مبتنی بر ادغام بلوکی تقویت شده^{۴۶}: این لایه به عنوان جایگزین لایه خود توجهی برای اتصالات دوربرد معرفی شده است. این رویکرد به واحدهای توجه درون بلوک‌ها اجازه می‌دهد

⁴¹ Pointer-Generator model

⁴² pretraining corpus

⁴³ Block-sparse self-attention

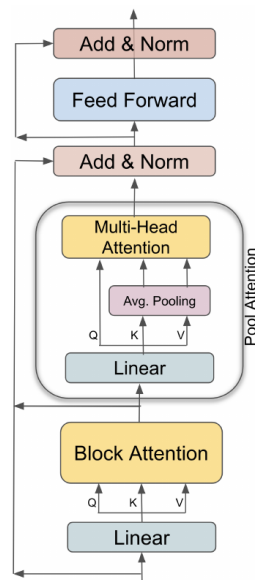
⁴⁴ Global-token mechanism

⁴⁵ Overlapping attention windows

⁴⁶ Pooling-augmented blockwise attention

تا به جای توجه به همسایگان بلافصل خود، بر خلاصه‌ای از اطلاعات کلی در بلوک‌ها تمرکز کنند. این لایه در تصویر ۳-۹ نشان داده شده است. این مدل را قادر می‌سازد تا از اطلاعات گسترده تری در سراسر سند برای تصمیم‌گیری استفاده کند و وابستگی‌های دوربرد را در نظر بگیرد. با بکارگیری عملیات ادغام، ابعاد و نمایش بردارهای توجه کاهش می‌یابد که منجر به افزایش سرعت و کارایی مدل می‌شود.

نویسندگان تکنیک‌های پیشنهادی را در تعدادی از وظایف توالی متن طولانی، از جمله پاسخ‌گویی به سؤال و خلاصه‌نویسی، ارزیابی کرده‌اند. نتایج نشان می‌دهد که مدل‌های اقتباس‌شده در تمامی وظایف از مدل‌های پایه بهتر عمل می‌کنند. این تکنیک‌ها استفاده از مدل‌های متنی از پیش آموزش دیده را برای طیف وسیعی از وظایف پردازش زبان طبیعی ممکن می‌سازد [۴۶]. کارهای تحقیقاتی در زمینه



شکل ۳-۹: لایه خودتوجهی تقویت شده ادغام شده [۴۶]

ی یادگیری تقویتی^{۴۷} و پردازش زبان طبیعی در سال‌های اخیر رشد کرده است. در یادگیری تقویتی یک عامل با محیط تعامل می‌کند و با آزمون و خطا، خط مشی بهینه را برای تصمیم‌گیری متوالی برای به حداکثر رساندن پاداش تجمعی آینده می‌آموزد. این پاداش می‌تواند یک معیار تعریف شده توسط توسعه دهنده بر اساس کار در حال حل باشد. در خلاصه‌سازی خودکار انتزاعی متن، نمونه‌هایی از چنین پاداش‌هایی ممکن است شامل حفظ برجستگی، مستلزم منطقی هدایت‌شده، و غیر افزونگی باشد. به

⁴⁷reinforcement learning

طور کلی، یادگیری تقویتی در چهار حوزه مختلف برای بهبود خلاصه‌سازی خودکار استفاده می‌شود:

۳-۳ روش‌های مبتنی بر یادگیری تقویتی

۱-۳-۳ یادگیری تقویتی برای حل چالش‌های مدل دنباله به دنباله عمیق

استفاده از یادگیری تقویتی به منظور حل مسائل گوناگونی که مدل‌های دنباله به دنباله عمیق قادر به حل آن‌ها نیستند، امکانات بیشتری را فراهم می‌کند. به عنوان مثال، مشکلاتی مانند کمبود نوآوری در ایجاد خلاصه‌های خلاقانه و آموزنده و کاهش کیفیت خلاصه‌ها در صورت افزایش طول مقالات منبع، با استفاده از سیستم‌های یادگیری تقویتی و یادگیری خط‌مشی^{۴۸} بهبود یافته است. علاوه بر این مدل‌های دنباله به دنباله عمیق را نمی‌توان برای خلاصه کردن طیف گسترده‌ای از اسناد استفاده کرد، زیرا مدلی که بر روی یک مجموعه داده آموزش داده می‌شود، در یک مجموعه داده دیگر به خوبی عمل نمی‌کند و قابلیت تعمیم ندارد. رویکردهای مبتنی بر یادگیری تقویتی می‌تواند این مشکل را با استفاده از گرادینان خط‌مشی انتقادی^{۴۹} و ترکیب آن با یادگیری انتقالی^{۵۰} برای انتقال دانش از یک مجموعه داده به مجموعه دیگر برطرف کنند [۱۲].

فریم‌ورک پوبرل^{۵۱} (ترکیب سیاست‌ها با حداکثر ارتباط حاشیه‌ای و یادگیری تقویتی) اهمیت، ارتباط و طول خلاصه را در زمینه خلاصه‌سازی چند سندی با جدا کردن مسئله بهینه‌سازی چند هدفه به مسائل فرعی کوچک‌تر که با استفاده از یادگیری تقویتی قابل حل هستند، بهینه می‌کند. اهمیت، ارتباط و طول خلاصه را در زمینه خلاصه‌سازی چند سندی با جدا کردن مسئله بهینه‌سازی چند هدفه به مسائل فرعی کوچک‌تر که قابل حل هستند، بهینه می‌کند. این فریم‌ورک از الگوریتم حداکثر ارتباط حاشیه‌ای^{۵۲} برای استخراج اطلاعات مهم از اسناد استفاده می‌کند. استفاده از این الگوریتم باعث افزایش ارتباط بین جملات و کاهش افزونگی می‌شود. در ادامه با از یادگیری تقویتی رای بهینه‌سازی هر هدف به صورت جداگانه استفاده می‌کند و خط‌مشی‌های جداگانه‌ای را برای اهمیت، ارتباط و طول می‌آموزد [۴۱]. خلاصه‌سازی چند سندی شامل سر و کار داشتن با اطلاعات پیچیده و همپوشانی از منابع متعدد است. الگوریتم‌های یادگیری تقویتی می‌توانند با مدل‌سازی خلاصه‌سازی به عنوان یک فرآیند تصمیم‌گیری

⁴⁸ policy learning

⁴⁹ self-critic policy gradient

⁵⁰ Transfer Learning (TL)

⁵¹ PoBRL

⁵² Maximal Marginal Relevance (MMR)

متوالی پیچیدگی را مدیریت کنند و یاد بگیرند جملات مرتبط حاوی اطلاعات را برای خلاصه انتخاب کنند. علاوه بر این یادگیری تقویتی امکان بهینه‌سازی همزمان اهداف متعدد مانند اهمیت، افزونگی و طول را فراهم می‌کند و موجب برقراری تعادل بین اهداف و تولید خلاصه‌های مختصر، مرتبط و غیر تکراری شوند.

سلیکيلمز^{۵۳} و همکاران مدل کدگذار-کدگشای چندعامله را برای بهبود خلاصه‌سازی اسناد طولانی با استفاده از عامل تعامل کننده^{۵۴} ارائه کرده‌اند. این مدل وظیفه کدگذاری یک متن طولانی را بین چندین عامل همکاری تقسیم می‌کند، که هر کدام مسئول یک زیربخش از ورودی هستند. این عوامل برای به اشتراک گذاشتن اطلاعات پایه‌ی جامع و ایجاد یک خلاصه متمرکز و منسجم با یکدیگر ارتباط برقرار می‌کنند. مدل ارائه شده در مقایسه با سایر مدل‌ها عملکرد بهتری دارد و خلاصه‌سازی اسناد طولانی با مدل‌های دنباله به دنباله را بهبود می‌بخشد.

۳-۲-۳ یادگیری تقویتی برای ترکیب خلاصه‌های استخراجی و انتزاعی

از یادگیری تقویتی برای ترکیب ویژگی‌های استخراجی با خلاصه انتزاعی برای استفاده از هر دو نوع خلاصه‌ی خودکار با الهام از رفتار انسان استفاده می‌شود. این مدل‌ها ابتدا برجسته‌ترین جملات را از سند ورودی استخراج می‌کنند، سپس با استفاده از دو شبکه: شبکه‌های استخراج کننده و انتزاعی، آنها را انتزاع می‌کنند. به عنوان مثال لیو^{۵۵} و همکاران یک چارچوب متخاصم را پیشنهاد می‌کنند که مدل‌های انتزاعی و استخراجی را همزمان با استفاده از گرادیان خط مشی برای بهینه‌سازی مدل انتزاعی برای خلاصه‌ای با پاداش بالا، آموزش می‌دهد که منجر به خلاصه‌ای منسجم‌تر می‌شود [۲۵]. همچنین چن و بانسال^{۵۶} یک مدل خلاصه‌سازی سریع پیشنهاد کردند که جملات برجسته را استخراج می‌کرد و سپس با استفاده از گرادیان خط مشی سطح جمله مبتنی بر یادگیری تقویتی بازنویسی می‌کرد [۲]. کریسینسکی^{۵۷} و همکاران دو روش برای افزایش سطح انتزاع در خلاصه سازی پیشنهاد می‌کنند: تجزیه رمزگشا به یک شبکه متنی و یک مدل زبانی از پیش آموزش دیده، و بهبود معیار جدید از طریق یادگیری خط مشی. تکنیک اول شامل یک شبکه‌ی محتوایی^{۵۸} و یک مدل زبانی از پیش آموزش دیده است. شبکه‌ی محتوایی

⁵³Celikyilmaz

⁵⁴communicating agent

⁵⁵Liu

⁵⁶Chen and Bansal

⁵⁷Kryscinski

⁵⁸contextual network

بخش‌های مرتبط از سند منبع را بازیابی کرده و آنها را فشرده می‌کند. مدل زبان از پیش آموزش حاوی دانش قبلی در مورد تولید زبان است. این تفکیک مسئولیت‌ها امکان استخراج بهتر و تولید جملات مختصر را فراهم می‌کند. تکنیک دوم شامل معرفی یک معیار جدید است که از طریق یادگیری خط مشی بهینه می‌شود. این معیار مدل را به تولید عبارات بدیع که در سند منبع وجود نداشته‌اند تشویق می‌کند. با ترکیب این معیار جدید با معیار روژ که همپوشانی کلمات را با خلاصه حقیقت پایه اندازه گیری می‌کند، مدل قادر به تولید خلاصه‌های انتزاعی با عملکرد بالا در همپوشانی کلمات می‌شود [۱۶].

۳-۳-۳ یادگیری تقویتی برای ایجاد معیارها و پاداش‌های جدید

خلاصه‌سازی اسناد، مانند سایر کارهای مولد زبان، اغلب به دلیل استفاده از اهداف آموزشی مبتنی بر درست‌نمایی بیشینه^{۵۹} مورد انتقاد قرار گرفته است. درست‌نمایی بیشینه کیفیت خلاصه‌ی تولید شده را در نظر نمی‌گیرد و ممکن است خلاصه‌هایی تولید کند که فقط یک کپی از اسناد ورودی هستند یا پر از کلمات بی‌معنی هستند. به همین دلیل، یادگیری تقویتی به عنوان جایگزینی برای بهینه‌سازی مستقیم مدل‌ها بر روی معیارهای ارزیابی و پاداش صریح به کیفیت پیش‌بینی‌های مدل استفاده شده است [۲۵]. معیارهای ارزیابی خلاصه‌سازی مانند روژ-۱^{۶۰}، روژ-۲^{۶۱}، روژ-ال^{۶۲} و امتیازبرت^{۶۳} به عنوان پاداش در رویکردهای یادگیری تقویتی استفاده شده است. با این حال، پارنل و همکاران استدلال می‌کنند که استفاده از امتیازات روژ به عنوان پاداش، جنبه‌های مهم خلاصه‌سازی، مانند خوانایی، روان بودن و اشتراک اطلاعات بین اسنادی در خلاصه‌سازی چند سندی را نادیده می‌گیرد و یک پاداش پوشش اصلاح شده همراه با یک برآوردگر گرادیان سیاست مبتنی بر اصول (ریلکس)^{۶۴} را پیشنهاد می‌دهند [۳۵، ۹]. ریلکس یک برآوردگر گرادیان خط مشی^{۶۵} با واریانس کم و بدون سوگیری^{۶۶} است که برای مسائل یادگیری تقویتی با فضاهاى کنش مداوم، مانند خلاصه‌سازی متن، مناسب است [۷].

در عبارت ۱-۳ تابع زیان ارائه شده برحسب ریلکس نمایش داده شده است. بخش اول این عبارت سیاست را به تولید خروجی‌هایی با پاداش مورد انتظار بالا و بخش دوم به تولید خروجی‌های مشابه

⁵⁹maximum likelihood

⁶⁰ROUGE-1

⁶¹ROUGE-2

⁶²ROUGE-L

⁶³BERTScore

⁶⁴modified coverage reward along with a principled policy gradient estimator (RELAX)

⁶⁵policy

⁶⁶bias

خروجی‌های قبلی تشویق می‌کند. در این عبارت r نشان دهنده‌ی پاداش $c_\phi(\tilde{z})$ یک متغیر کنترلی از پارامترهای است که انتظار می‌رود با پاداش کاهش واریانس همبستگی داشته باشد. $p(y_s)$ احتمال دنباله مشاهده شده خروجی y_s است. z دنباله نمونه‌های $Gumbel - Softmax$ است. \tilde{z} دنباله ای از نمونه‌ها از یک توزیع $Gumbel - Softmax$ مشروط بر y_s است.

$$L_{RELAX} = -[r - c_\phi(\tilde{z})] \log p(y^s) + c_\phi(z) - c_\phi(\tilde{z}) \quad (۱-۳)$$

۴-۳-۳ یادگیری تقویتی برای ایجاد خلاصه متناسب با نیاز کاربر

در خلاصه‌سازی متن، یادگیری تقویتی می‌تواند نقش مهمی به عنوان یک رویکرد پیشرفته برای ارائه خلاصه‌های متناسب با نیاز کاربر ایفا کند. با استفاده از یادگیری تقویتی، سیستم‌ها قادر به تحلیل و فهم متن‌ها و درک نیازهای کاربران می‌شوند، سپس با اعمال تصمیمات متناسب، خلاصه‌هایی ایجاد می‌کنند که بیان کننده اصلی‌ترین اطلاعات و مفاهیم موجود در متن اصلی هستند. این رویکرد توانایی ارائه خلاصه‌های متناسب با نیازهای کاربر را بهبود می‌بخشد و تجربه خواندن و درک محتوای متن را بهبود می‌بخشد. همچنین، با استفاده از یادگیری تقویتی، سیستم‌ها قادر به بهبود خودکار خلاصه‌سازی و افزایش کیفیت خلاصه‌های تولید شده هستند. سایر روش‌های خلاصه‌سازی به کاربران اجازه نمی‌دهند، سلیقه‌ی خود را برای کنترل جنبه‌های مختلف خلاصه‌های تولید شده نشان بدهند.

مدل کنترل سام^{۶۷} با افزودن توکن‌های کنترلی به ابتدای متن ورودی و استفاده از یک مدل کدگذار-کدگشا به کاربران اجازه‌ی اعمال ویژگی‌های مورد نیازهای خود بر خلاصه را می‌دهند. به عنوان مثال برای کنترل طول خلاصه خروجی ده طول مجزا تعریف می‌شود و هریک توکن‌های کنترلی نشانگر یکی از این طول‌ها هستند. هدف آموزش این مدل از طریق تابع زیان درست‌نمایی بیشینه^{۶۸} است [۶]. این هدف آموزش هیچ سیگنال نظارتی صریحی ندارد. برای حل این مشکل چان^{۶۹} و همکاران با اعمال محدودیت بر روی هدف آموزشی با استفاده از فرآیند تصمیم‌گیری مارکوف محدود^{۷۰} یک چهارچوب خلاصه سازی پیشنهاد کرده‌اند که شامل یک تابع پاداش همراه با مجموعه‌ای از محدودیت‌ها است و

^{۶۷}controlSum

^{۶۸}maximum likelihood loss

^{۶۹}Chan

^{۷۰}Constrained Markov Decision Process (CMDP)

کنترل خلاصه سازی را تسهیل می‌کند. هدف عامل بیشینه کردن پاداش مورد انتظار در عین اعمال محدودیت بر هزینه‌ها است. با داشتن این هدف، تصمیم‌گیرنده سعی می‌کند سیاستی را انتخاب کند که منجر به بیشینه کردن پاداش کلی تجمعی در طول زمان شود، در حالی که محدودیت‌ها بر هزینه‌ها رعایت شوند. این هدف مدل را تشویق می‌کند که خلاصه‌ای شبیه خلاصه‌ی تولید شده توسط انسان تولید کند. با استفاده از این مدل کاربران می‌توانند طول، میزان فشردگی و محتوای خلاصه را کنترل کنند به عنوان مثال توضیحات یک محصول را به گونه‌ای خلاصه کند که در یک محدودیت کلمه در تبلیغات آنلاین قرار گیرد. برای تبدیل مسئله محدود به مسئله بدون محدودیت از ساده‌سازی لاگرانژ^{۷۱} و برای بهینه سازی از الگوریتم بهینه سازی مبتنی بر گرادیان، مانند ادامه استفاده می‌شود. برای اندازه‌گیری شباهت بین خلاصه خروجی و مرجع بر اساس تعبیه‌های متنی برت به عنوان تابع پاداش از امتیازبرت استفاده می‌شود. برای کنترل تمرکز خلاصه بر روی یک موجودیت نامدار^{۷۲} ابتدا ارجاع موجودیت نامدار به سند اضافه می‌شود سپس یک محدودیت سوال و جواب اعمال می‌شود. این محدودیت بر روی امتیاز اف-۱ خروجی یک مدل سوال جواب که ورودی آن شامل یک سوال راجع به موجودیت نامدار و خلاصه‌ی تولید شده است اعمال می‌شود. علاوه بر این دو محدودیت عدم تکرار برای گرم^{۷۳} و موجودیت‌های درخواستی برای افزایش خوانایی و کاهش تکرار در متن اعمال می‌شود. مدل اینت‌سام^{۷۴} یک مدل خلاصه‌سازی تعاملی با هدف خلاصه کردن اطلاعات مهم بر اساس کوئری‌های^{۷۵} کاربر و ارائه کوئری پیشنهادی برای کمک به کاربران است. در ابتدا این مدل یک خلاصه‌ی اولیه تولید می‌کند و به کاربر نمایش می‌دهد سپس یک کوئری از کاربر دریافت می‌کند و خلاصه‌ی اولیه به همراه پاسخ کوئری را به کاربر نمایش می‌دهد. برای ارزیابی مدل ارائه شده مساحت منحنی بازیابی^{۷۶} بر اساس طول خلاصه معرفی شده است که ستون عمودی آن امتیاز بازیابی روژ و ستون افقی آن طول خلاصه مرجع می‌باشد و مساحت بیشتر زیر منحنی نشان دهنده‌ی مدل بهتر است. یک نمونه از این نمودار در شکل ۳-۱۰ نمایش داده شده است [۳۸].

شاپیرا و همکاران برای بهبود مدل اینت‌سام و بهبود سرعت عمل در پاسخ‌گویی، توانایی پردازش کامل متون طولانی و رعایت تعادل میان اطلاعات کلی مقاله و اطلاعات مورد نیاز کاربر یک مدل جدید ارائه داده‌اند. ورودی این مدل مجموعه‌ی اسناد، کوئری و تاریخچه‌ی تعاملات با کاربر به همراه خروجی

⁷¹ Lagrangian relaxation

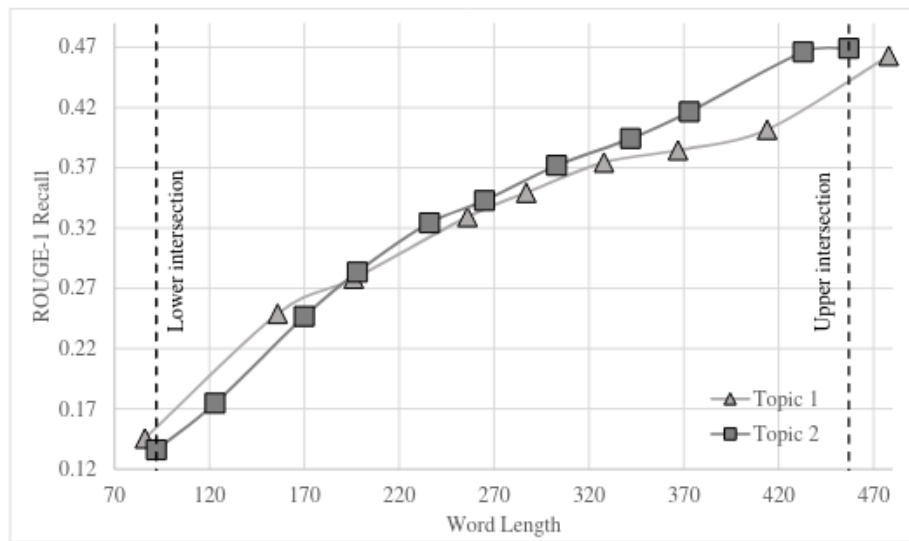
⁷² named entity

⁷³ trigram

⁷⁴ IntSumm

⁷⁵ query

⁷⁶ recall



شکل ۳-۱۰: یک نمونه از نمودار منحنی بازیابی بر اساس طول [۳۸]
این نمودار دو تعامل متفاوت با سیستم خلاصه‌سازی را مقایسه می‌کند. هر نقطه نمایانگر خروجی هر مرحله تعامل با کاربر است.

قبلی است. در ابتدا تعبیه کوئری به تعبیه اسناد ورودی الحاق شده سپس امتیاز $qMMR$ با استفاده از مدل $RL - MMR$ محاسبه می‌شود. هدف این امتیاز ایجاد خلاصه‌ای شبیه به اسناد ورودی و کوئری و متفاوت از تاریخچه است. سپس با استفاده از مکانیزم توجه با مرکزیت دوگانه^{۷۷} بر اساس کدگذاری به دست آمده از تاریخچه و مدل $RL - MMR$

توزیع احتمال هر جمله را به دست می‌آید. مدل ام‌سام^{۷۸} یک مدل خودرگرسیون^{۷۹} است که برای آموزش آن از یادگیری تقویتی به همراه مکانیزم پاداش دوگانه استفاده می‌شود. معیار دلتا-روژ^{۸۰} برای سنجش میزان اطلاعات اضافه‌ی خروجی نسبت به خروجی‌های قبلی و شباهت واژگانی و معنایی برای سنجش میزان شباهت خروجی به کوئری به عنوان پاداش استفاده شده‌اند. مدل $RL - MMR$ موجب افزایش سرعت پردازش اطلاعات در مدل و پردازش کامل مجموعه‌ی اسناد و مکانیزم پاداش دوگانه تعادل موجب ایجاد تعادل اطلاعات می‌شود. ساختار مدل در شکل ۳-۱۱ نشان داده شده است [۳۷].

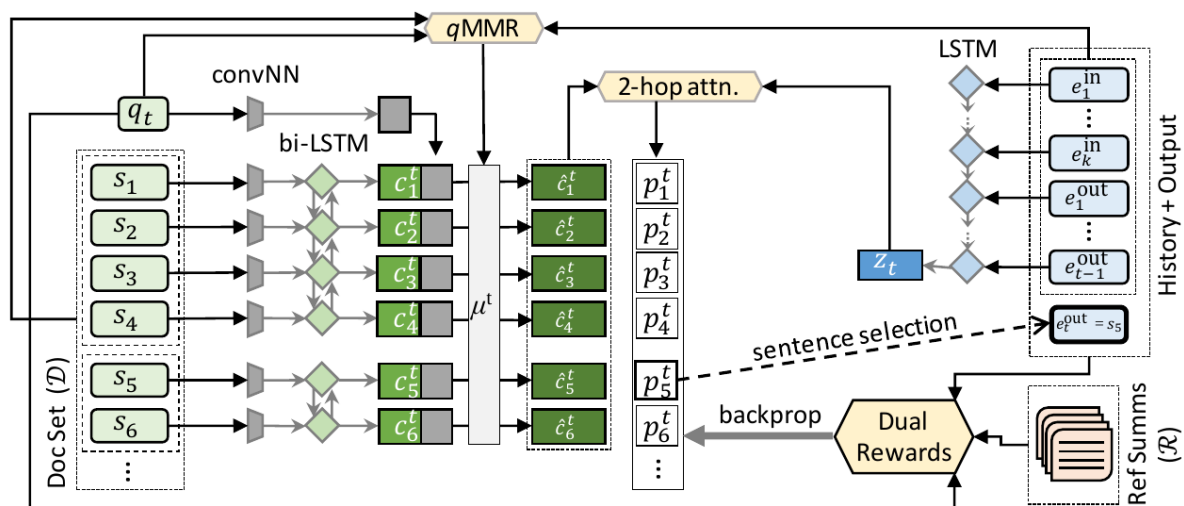
برای متون طولانی یک روش بهینه ارائه می‌دهد. مدل *AWESOME* از یک روش جدید دو مرحله‌ای برای بهبود خلاصه‌سازی متون طولانی استفاده می‌کند: استفاده از حافظه خارجی و شناسایی

⁷⁷two hub attention

⁷⁸MSumm

⁷⁹Autoregressive

⁸⁰Delta-ROUGE



شکل ۳-۱۱: معماری مدل امسام [۳۷]

مفاهیم برجسته در کل سند^{۸۱}. حافظه‌های خارجی در طول فرآیند خلاصه سازی قابل دسترسی هستند و بخش‌های کدگذاری شده سند و خلاصه‌های مربوط به آن‌ها را ردیابی می‌کنند تا درک جامع عمیق‌تر و انسجام خلاصه را تقویت کنند. علاوه بر این، محتوای برجسته‌ی جامع از بخش‌های گذشته و آینده استخراج می‌شود تا هر بخش را در حین کدگذاری تقویت کند و اطمینان حاصل شود که موضوعات مهم در خلاصه مورد توجه قرار می‌گیرند. با بهره‌گیری از این مکانیزم‌ها و یک معماری مبتنی بر حافظه کارآمد، این روش در زمینه‌های اطلاعاتی، انسجام و وفاداری نسبت به روش‌های قبلی عملکرد بهتری دارد؟؟

۳-۴ روش‌های مبتنی بر مدل‌های زبانی بزرگ و چالش‌ها

با استفاده از روش تنظیم پرامپت^{۸۲} توانستند عملکرد مدل‌های زبانی بزرگ را بهبود بخشند. آن‌ها به جای تنظیم دقیق پارامترهای مدل‌های فشرده‌شده که با روش‌هایی مانند کوانتیزشن و هرس مدل انجام می‌شود، تمرکز خود را بر سازگاری ورودی‌ها قرار دادند. در این روش، پرامپت‌های نرم^{۸۳} به عنوان توکن‌های یادگیرنده به ابتدای توالی ورودی اضافه شده و از طریق بهینه‌سازی احتمال خروجی مدل تنظیم شدند، بدون آن‌که پارامترهای مدل فشرده تغییر کنند. این اعلان‌ها توانستند افت دقت ناشی از

⁸¹global salient content identification

⁸²Prompt Tuning)

⁸³soft prompts

فشرده‌سازی را جبران کرده و حتی عملکرد مدل را در برخی موارد به سطح مدل‌های اصلی برسانند. ویژگی مهم این اعلان‌ها انتقال‌پذیری بالا بود، به‌طوری که در محیط‌ها، روش‌های فشرده‌سازی و وظایف مختلف قابل استفاده بودند. نتایج نشان داد که این روش سبک و کم‌هزینه می‌تواند تعادل میان کارایی و دقت مدل‌های فشرده را به شکلی مؤثر برقرار کند [۴۷].

۳-۴-۱ هزینه‌گویی در مدل‌های زبانی بزرگ

زیوی جی و همکارانش به بررسی مسئله هزینه‌گویی در مدل‌های زبان بزرگ پرداخته‌اند، پدیده‌ای که در آن مدل‌ها اطلاعاتی نامعتبر یا غیرواقعی تولید می‌کنند. این پژوهش با تمرکز بر کاربردهای پزشکی، روشی تعاملی مبتنی بر خودبازتابی ارائه می‌دهد که از طریق یک چرخه تکراری تولید، امتیازدهی و اصلاح، اعتبار پاسخ‌ها را افزایش می‌دهد. آزمایش‌های انجام شده نشان‌دهنده کاهش قابل توجه هزینه‌گویی و افزایش قابلیت اطمینان سیستم‌ها در پاسخگویی به سوالات پزشکی است [۱۱].

• ییجون شیائو و همکارانش رابطه بین هزینه‌گویی و عدم قطعیت پیش‌بینی در تولید زبان شرطی را بررسی کرده‌اند. آن‌ها با تأکید بر نقش مهم عدم قطعیت اپیستمیک، روشی برای بهبود الگوریتم جستجوی پرتو پیشنهاد می‌دهند که با کاهش عدم قطعیت، میزان هزینه‌گویی تولید شده توسط مدل را کاهش می‌دهد. این پژوهش در وظایفی مانند توصیف تصویر و تولید داده به متن انجام شده و نتایج مثبت قابل توجهی در کاهش محتوای غیرواقعی نشان داده است [۴۵].

دنیل کینگ و همکارانش روشی به نام PINOCCHIO برای بهبود انسجام سیستم‌های خلاصه‌سازی انتزاعی ارائه داده‌اند. این روش با اعمال محدودیت‌هایی در جستجوی پرتو، تولید خروجی‌هایی را که به منبع متن مرتبط نیستند کاهش می‌دهد. نتایج آزمایش‌ها نشان داده که این رویکرد می‌تواند انسجام متون تولید شده را به طور قابل توجهی افزایش داده و هزینه‌گویی را کاهش دهد، بدون آنکه بر روانی متن تأثیر زیادی بگذارد [۱۳].

، جاشوا ماینز و همکارانش به بررسی مسئله هزینه‌گویی در خلاصه‌سازی انتزاعی اسناد پرداخته‌اند. آن‌ها نشان می‌دهند که مدل‌های تولید متن شرطی اغلب محتوایی تولید می‌کنند که با متن منبع سازگار نیست و این پدیده را به عنوان "هزینه‌گویی درونی" و "هزینه‌گویی بیرونی" طبقه‌بندی می‌کنند. این مقاله از طریق ارزیابی انسانی نشان می‌دهد که بیش از ۷۰٪ خلاصه‌ها شامل محتوای هزینه‌گویی آمیز هستند که اکثریت آن‌ها به‌ویژه در هزینه‌گویی بیرونی، نادرست می‌باشند. نویسندگان همچنین پیشنهاد می‌کنند که مدل‌های از پیش آموزش‌دیده مانند BERTS2S نسبت به مدل‌های دیگر، خلاصه‌های دقیق‌تر و

با هزینه گویی کمتر تولید می کنند. این پژوهش با ارائه معیارهایی جدید برای ارزیابی دقت و اعتبار خلاصه ها، مسیر بهتری برای بهبود ارزیابی های خودکار و روش های تولید خلاصه سازی باز می کند [۲۹]. جورج کریسوستومو و همکارانش تأثیر هرس مدل های زبان بزرگ بر کاهش هزینه گویی در خلاصه سازی متون را بررسی کرده اند. آن ها با استفاده از روش های پیشرفته هرس، مانند *Wanda* و *SparseGPT*، نشان داده اند که مدل های هرس شده نسبت به مدل های اصلی هزینه گویی کمتری دارند. این مدل ها بیشتر بر متن منبع تکیه کرده و خلاصه هایی با همپوشانی واژگانی بالاتر و محتوای واقع گرایانه تر تولید می کنند. آزمایش ها روی پنج مجموعه داده مختلف و چندین مدل، کاهش قابل توجه ریسک هزینه گویی را با افزایش میزان هرس نشان داده است. این پژوهش، استفاده از مدل های هرس شده را به عنوان راهکاری مؤثر برای کاهش هزینه گویی در خلاصه سازی پیشنهاد می کند.

مسئله هزینه گویی در مدل های زبان بزرگ یکی از چالش های اساسی در تولید زبان طبیعی است که می تواند اعتبار و اطمینان به این مدل ها را تحت تأثیر قرار دهد. استفاده از روش های مبتکرانه و دقیق می تواند به طور قابل توجهی میزان هزینه گویی را کاهش داده و انسجام و دقت خروجی های مدل را بهبود بخشد. این پیشرفت ها راه را برای استفاده ایمن تر و موثرتر از مدل های زبان در کاربردهای حساس، از جمله پزشکی و خلاصه سازی متون، هموار می کند.

۳-۴-۲ هرس مدل های زبانی

در حوزه بهینه سازی مدل های زبانی بزرگ، روش های مختلفی برای کاهش پیچیدگی محاسباتی و منابع مورد نیاز ارائه شده اند. فانگ و همکاران الگوریتمی برای هرس ساختاریافته مدل های زبانی بزرگ ارائه داده اند که با وابستگی کمتر به داده، نیازی به مجموعه داده کامل ندارد و در مدت کوتاهی قابل اجرا است. الگوریتم *LLM – Pruner* شامل سه مرحله کلیدی است: ابتدا وابستگی بین اجزای مدل شناسایی می شود، سپس اهمیت هر بخش وابسته به صورت مستقل از وظایف^{۸۴} ارزیابی می شود و در نهایت، با استفاده از روش لورا، عملکرد مدل با حداقل داده بازیابی می شود. این روش توانسته با هرس ۲۰ درصد از مدل، ۹۴ درصد از کارایی اولیه را حفظ کند. نتایج این تحقیق نشان می دهد که رویکرد پیشنهادی با شناسایی و هرس بخش های وابسته مدل به صورت داده محور و بدون نیاز به به روزرسانی وزن ها، روشی کارآمد و سریع برای کاهش پیچیدگی مدل های زبانی بزرگ ارائه می دهد [۳۰].

ژانگ و همکاران الگوریتم *D – PRUNER* را معرفی کرده اند. این الگوریتم با هدف ارائه یک روش

⁸⁴task-agnostic

هرس داده‌محور و غیرساختاری طراحی شده که ضمن حفظ دانش عمومی مدل، توانایی آن را در فهم دانش دامنه خاص نیز حفظ کند. رویکرد $D - PRUNER$ در سه مرحله کلیدی شامل شناسایی اهمیت وزن‌های عمومی، بهینه‌سازی تابع ضرر با اضافه کردن عبارت منظم‌سازی^{۸۵} برای جلوگیری از تغییر وزن‌های مهم، و در نهایت هرس وزن‌های کم‌اهمیت با استفاده از "فیشر تجربی" پیاده‌سازی می‌شود. نتایج این تحقیق نشان می‌دهد که این روش نه تنها پیچیدگی محاسباتی مدل را کاهش می‌دهد، بلکه با حفظ توازن میان دانش عمومی و خاص، عملکرد مدل را در دامنه‌های تخصصی بهبود می‌بخشد و نرخ سردرگمی^{۸۶} کمتری نسبت به روش‌های دیگر ارائه می‌دهد [۵۰].

یکی از پژوهش‌های کلیدی در زمینه کاهش پیچیدگی مدل‌های زبانی بزرگ، مقاله‌ای از *XinMen* است. این تحقیق بر شناسایی اضافه‌بودگی^{۸۷} در لایه‌های شبکه عصبی مدل‌های زبانی بزرگ تمرکز دارد. یافته اصلی نشان می‌دهد که بسیاری از لایه‌های میانی و انتهایی مدل تغییرات محدودی در حالات پنهان ایجاد می‌کنند و می‌توان آن‌ها را با حداقل تأثیر بر عملکرد مدل حذف کرد.

برای ارزیابی اهمیت هر لایه، متریک $BlockInfluence(BI)$ معرفی شده است. این متریک میزان تغییر حالات پنهان^{۸۸} پس از عبور از هر لایه را می‌سنجد. BI بر اساس شباهت کسینوسی بین ورودی و خروجی لایه تعریف شده است؛ به این صورت که BI پایین‌تر نشان‌دهنده تغییرات کمتر و اهمیت کمتر لایه است.

با استفاده از این متریک، لایه‌های با BI پایین شناسایی و حذف می‌شوند. این روش توانسته است با حذف ۲۵٪ از لایه‌های مدل، حدود ۹۰٪ از عملکرد اولیه را حفظ کند و در عین حال، از روش‌های پیشرفته دیگر در این زمینه پیشی بگیرد. همچنین، این روش با تکنیک‌های کوانتایزه‌سازی سازگار است و امکان کاهش بیشتر محاسبات و پارامترها را فراهم می‌کند.

$$BI_i = 1 - \mathbb{E}_{X,t} \left[\frac{X_{i,t}^\top X_{i+1,t}}{\|X_{i,t}\|_2 \|X_{i+1,t}\|_2} \right] \quad (2-3)$$

در این رابطه‌ی ۲-۳ $X_{i,t}$ نشان‌دهنده t -امین ردیف از حالات پنهان لایه i است. $X_{i+1,t}$ نشان‌دهنده t -امین ردیف از حالات پنهان لایه $i+1$ است. $\|\cdot\|_2$ نرم اقلیدسی را نشان می‌دهد و $\mathbb{E}_{X,t}$ بیانگر امید

⁸⁵regularization term

⁸⁶perplexity

⁸⁷redundancy

⁸⁸hidden states

ریاضی روی مقادیر X و t است. این متریک بر اساس شباهت کسینوسی بین ورودی و خروجی لایه عمل می‌کند. هرچه این شباهت بیشتر باشد (مقدار BI کمتر)، لایه تغییرات کمتری ایجاد کرده و اهمیت آن کاهش می‌یابد.

یکی از پژوهش‌های برجسته در زمینه کاهش پیچیدگی مدل‌های زبانی بزرگ، مقاله‌ای از یانگ ژانگ است که روش برش دقیق‌تر^{۸۹} را برای هرس لایه‌های مدل‌های زبانی بزرگ ارائه می‌دهد. این روش برخلاف روش‌های پیشین، لایه‌های خودتوجهی^{۹۰} و شبکه عصبی پیشخور^{۹۱} را به‌صورت جداگانه و مستقل به‌عنوان کاندیداهای هرس در نظر می‌گیرد. الگوریتم به‌صورت تکراری لایه‌هایی را که حذف آن‌ها کمترین تغییر را در خروجی مدل ایجاد می‌کنند، انتخاب و حذف می‌کند. برای اندازه‌گیری تغییرات خروجی، از معیارهای مختلفی مانند فاصله اقلیدسی، فاصله زاویه‌ای و واگرایی جنسن-شانون (JSD) استفاده می‌شود. معیار JSD ، که برای سنجش شباهت بین توزیع‌های احتمالاتی طراحی شده، با در نظر گرفتن توزیع خروجی‌های مدل قبل و بعد از حذف یک لایه، به شناسایی لایه‌های کم‌اهمیت کمک می‌کند.

نتایج این روش نشان می‌دهد که با حذف 25% از لایه‌های $3-8B$ ، 90% از عملکرد مدل حفظ می‌شود و با حذف 30% از لایه‌های $3-70B$ ، مدل توانسته است 95% عملکرد اولیه را حفظ کند، بدون نیاز به تنظیم دوباره. همچنین، در لایه‌های انتهایی مدل، ترکیبی از حذف لایه‌های توجه و استفاده از لایه‌های متوالی شبکه تغذیه پیش‌رو، به ساختارهای کارآمدتری منجر شده است. این روش نه تنها عملکرد بهتری نسبت به روش‌های پیش‌رو ارائه می‌دهد، بلکه طراحی جدیدی برای معماری مدل‌های آینده پیشنهاد می‌کند که می‌تواند بازدهی بیشتری در استفاده از منابع محاسباتی داشته باشد [۵۲].

۳-۴-۳ هزینه گویی در خلاصه‌سازی

یکی از مطالعات برجسته در زمینه کاهش هزینه گویی در مدل‌های زبانی بزرگ و بهبود کیفیت خلاصه‌سازی انتزاعی متون طولانی، مقاله‌ای از یو شیا است. این تحقیق اولین چارچوب یادگیری فعال برای کاهش هزینه گویی در مدل‌های زبانی ارائه می‌دهد و بر روی تولید خلاصه‌های متنی با تأکید بر حفظ دقت معنایی متمرکز است. روش پیشنهادی، سه نوع خطای رایج شامل خطاهای چارچوب معنایی، خطاهای گفتمان، و خطاهای قابلیت تأیید محتوا را شناسایی کرده و از متریک‌های پیشرفته‌ای مانند $FactKB$

⁸⁹FINERCUT

⁹⁰self-attention

⁹¹FFN

BERT – P و UniEval برای ارزیابی این خطاها استفاده می‌کند.

الگوریتم *HADAS*، با رویکردی داده‌محور و متنوع‌محور، نمونه‌های داده‌ای را که شامل انواع مختلفی از هزین‌گوئیات هستند برای بهبود مدل انتخاب می‌کند. این انتخاب با استفاده از واگرایی جنسن-شانون (*JSD*) به منظور سنجش تنوع توزیع انواع هزین‌گوئیات انجام می‌شود. این روش نه تنها به طور مؤثری هزین‌گوئیات در خلاصه‌سازی‌های انتزاعی را کاهش داده است، بلکه کیفیت و دقت خلاصه‌های تولیدشده را نیز بهبود می‌بخشد. نتایج نشان می‌دهد که این چارچوب، با کاهش نیاز به حاشیه‌نویسی انسانی پرهزینه و ارائه روشی کارآمد برای هرس مدل، به‌طور خاص در زمینه خلاصه‌سازی متون طولانی تأثیرگذار است [۴۴].

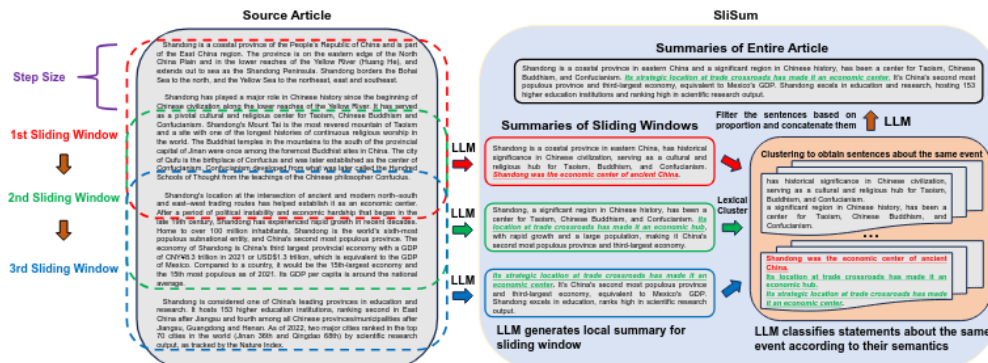
تایجی لی و همکارانش روش *SliSum* را برای کاهش هزین‌گوئی و افزایش دقت در خلاصه‌سازی مدل‌های زبان بزرگ ارائه داده‌اند. این روش شامل سه مرحله کلیدی است:

- تولید خلاصه‌های محلی با پنجره‌های لغزنده: متن منبع به بخش‌های همپوشانی‌شده تقسیم می‌شود (پنجره‌های لغزنده)، و مدل زبان برای هر پنجره یک خلاصه محلی تولید می‌کند. این همپوشانی‌ها کمک می‌کنند که متن به طور عادلانه در سراسر مقاله پردازش شود و مشکل سوگیری موقعیتی کاهش یابد.

- خوشه‌بندی و فیلتر کردن بر اساس انسجام درونی: جملات تولیدشده در خلاصه‌های محلی با استفاده از الگوریتم خوشه‌بندی *Lexical* (مانند *DBSCAN*) تجزیه و تحلیل می‌شوند. جملات مرتبط با یک رویداد مشخص در یک خوشه قرار می‌گیرند و جملات کم‌اهمیت یا ناسازگار حذف می‌شوند تا انسجام و دقت حفظ شود.

- تجمیع و تولید خلاصه جامع: جملات منتخب از خوشه‌ها، با استفاده از رأی‌گیری اکثریتی و به ترتیب معنایی ترکیب می‌شوند تا یک خلاصه جامع و دقیق برای کل مقاله تولید شود. این مرحله همچنین از مدل زبان برای اطمینان از روانی و ساختارمند بودن متن استفاده می‌کند.

این رویکرد، بدون نیاز به تنظیم مجدد مدل یا منابع اضافی، باعث کاهش هزین‌گوئی و افزایش دقت و انسجام خلاصه‌های تولیدشده می‌شود و در متون کوتاه و بلند عملکردی مؤثر نشان می‌دهد. تصویر ۱۲-۳ نشان‌دهنده مراحل مختلف این فرآیند است که شامل پردازش هر پنجره و ترکیب نتایج به روش ساختاریافته است [۲۳].



شکل ۳-۱۲: رویکرد *SlimSum* برای حل تضادهای معنایی در خلاصه‌سازی
 توضیح: تصویری از فرآیند *SlimSum* که با رأی‌گیری اکثریت میان جملات هر خوشه بر اساس معنای آن‌ها، به حل مشکل تضاد معنایی در خلاصه‌سازی می‌پردازد. به عنوان مثال، جملات سبز دارای معنای مشابه هستند و دو بار ظاهر می‌شوند، در حالی که جمله قرمز با معنای متفاوت فقط یک بار ظاهر می‌شود. بنابراین، جمله دوم سبز برای خلاصه نهایی انتخاب می‌شود. *SlimSum* مقالات منبع را در سطح جملات پردازش می‌کند و برای ساده‌تر کردن نمایش، پنجره‌های موجود در تصویر به صورت خطوط متنی نمایش داده شده‌اند [۲۳].

۳-۵ معیارهای ارزیابی خلاصه‌سازی خودکار

در ارزیابی خلاصه‌سازی خودکار، هدف اصلی سنجش کیفیت و دقت خلاصه‌های تولید شده است. برای این منظور، معیارهای مختلفی طراحی شده‌اند که توانایی مدل‌ها در تولید خلاصه‌های دقیق، مفهومی و وفادار به متن اصلی را ارزیابی می‌کنند. این معیارها می‌توانند به صورت کمی و بر اساس مقایسه خلاصه‌های تولید شده با متن‌های مرجع، یا به‌طور کیفی با استفاده از تحلیل‌های معنایی و ساختاری عمل کنند. در این بخش، به معرفی و بررسی مهم‌ترین معیارهای ارزیابی در این حوزه مانند روژ، امتیاز برت، فکت‌سی‌سی پرداخته می‌شود. این معیارها هرکدام از جنبه‌های مختلف کیفیت خلاصه‌ها را ارزیابی کرده و نقش مهمی در توسعه و بهبود الگوریتم‌های خلاصه‌سازی خودکار ایفا می‌کنند. همچنین، محدودیت‌های این معیارها در شناسایی هالوسینیشن و چالش‌های آن‌ها در سنجش وفاداری خلاصه‌ها نیز مورد بررسی قرار خواهد گرفت.

- روژ یکی از پرکاربردترین معیارهای ارزیابی در خلاصه‌سازی خودکار است که بر اساس تطابق کلمات یا عبارات $n - gram$ بین خلاصه تولیدشده و متن مرجع عمل می‌کند. روژ شاخص‌هایی مانند دقت ^{۹۲}، فراخوان ^{۹۳} و $F1$ را برای شباهت زبانی محاسبه می‌کند. اگرچه این معیار در اندازه‌گیری شباهت‌های سطح کلمه مؤثر است، اما از درک معنایی عمیق و سنجش وفاداری

^{۹۲}Precision

^{۹۳}Recall

محتوا ناتوان است. به عنوان مثال، Zhou و همکاران نشان داده‌اند که اگر یک خلاصه شامل مقدار زیادی محتوای هالوسینیشن باشد، ممکن است همچنان روژ بالایی کسب کند. [۵۳، ۲۴].

- امتیاز برت برخلاف معیارهای سطح کلمه مانند روژ، از مدل‌های زبانی پیش‌آموزش‌دیده (مانند برت) برای اندازه‌گیری شباهت معنایی میان خلاصه تولیدشده و متن مرجع استفاده می‌کند. این معیار توانایی بیشتری در درک روابط زبانی پیچیده و شباهت معنایی عمیق دارد، اما همچنان در شناسایی دقیق هالوسینیشن‌ها محدودیت‌هایی دارد [۵۱].

- فکت‌سی‌سی مبتنی بر مدل‌های استنتاج متنی^{۹۴} طراحی شده و تمرکز آن بر بررسی میزان درستی و وفاداری اطلاعات موجود در خلاصه به متن اصلی است. فکت‌سی‌سی تلاش می‌کند تا محتواهای نادرست یا ناسازگار را شناسایی کند و از این طریق بهبودهایی در سنجش وفاداری ایجاد کند. این معیار نسبت به روژ و امتیاز برت توانایی بهتری در ارزیابی هالوسینیشن دارد [۱۵].

- سطح موجودیت^{۹۵} بر شناسایی ناسازگاری‌های موجودیتی تمرکز دارند. برای مثال، شناسایی موجودیت‌هایی که در خلاصه ظاهر می‌شوند اما در متن اصلی نیستند (هذیان موجودیتی). این روش از ابزارهای شناسایی موجودیت نام‌دار^{۹۶} استفاده می‌کند [۳۲].

$$(3-3) \quad \text{موجودیت‌ها در مرجع} \cap \text{موجودیت‌ها در فرضیه} = \frac{\text{موجودیت‌ها در مرجع} \cap \text{موجودیت‌ها در فرضیه}}{\text{موجودیت‌ها در فرضیه}} = \text{دقت موجودیت}$$

- ریسک هزیان^{۹۷} به منظور ارزیابی ریسک هذیان در متن‌های تولیدشده توسط مدل‌های خلاصه‌سازی انتزاعی طراحی شده است. این متریک از احتمالات سطح توکن که توسط دو مدل پیش‌بینی می‌شوند، استفاده می‌کند: مدل دنباله به دنباله که به ورودی متن وابسته است و مدل زبان کمکی که مستقل از متن ورودی عمل می‌کند. فرمول متریک به گونه‌ای است که توکن‌هایی با احتمال پایین از مدل دنباله به دنباله و احتمال بالاتر از مدل زبان کمکی را که احتمالاً ناشی از توجه بیش‌از حد به محتوای قبلی هستند، شناسایی و جریمه می‌کند. ریسک هزیان نیازی به خلاصه‌های مرجع ندارد و به صورت مستقیم از مدل‌های موجود بدون نیاز به آموزش اضافه قابل استفاده است.

^{۹۴}Natural Language Inference(NLI)

^{۹۵}Entity-level Metrics

^{۹۶}NER

^{۹۷}HaRiM+

این متریک بر اساس ارزیابی‌های انجام‌شده، ارتباط بالایی با قضاوت انسانی در معیارهایی همچون صحت و کیفیت خلاصه‌ها دارد و ابزاری مؤثر برای ارزیابی کیفیت و کاهش محتوای هذیانی در مدل‌ها ارائه می‌دهد. متریک ریسک هزیان به صورت ریاضی به شکل زیر تعریف می‌شود:

(۴-۳)

$$HaRiM = \frac{1}{L} \sum_{i=0}^L (1 - p_{s2s}(y_i|y_{<i}, X)) (1 - (p_{s2s}(y_i|y_{<i}, X) - p_{lm}(y_i|y_{<i})))$$

در اینجا:

□ L : طول کل دنباله تولید شده است.□ $p_{s2s}(y_i|y_{<i}, X)$: احتمال توکن y_i که توسط مدل دنباله به دنباله پیش‌بینی شده است، باتوجه به ورودی X و توکن‌های قبلی $y_{<i}$.□ $p_{lm}(y_i|y_{<i})$: احتمال توکن y_i که توسط مدل زبان کمکی پیش‌بینی شده است، بدون توجهبه ورودی X .

این متریک زمانی افزایش می‌یابد که:

□ مدل دنباله به دنباله اعتماد کمتری به یک توکن داشته باشد (احتمال p_{s2s} کوچک باشد).

□ مدل زبان کمکی اعتماد بیشتری نسبت به مدل دنباله به دنباله به یک توکن داشته باشد

(احتمال p_{lm} غالب بر p_{s2s} باشد).

این رفتار مواردی را شناسایی می‌کند که در آن مدل محتوا را بیشتر تحت تأثیر توکن‌های قبلی تولید کرده است و ارتباط کمتری با متن ورودی دارد، که نشان‌دهنده خطر تولید محتوای هذیانی است. [۴۰].

این مدل بر اساس استفاده از مدل‌های استنتاج زبانی (LLMs) برای شناسایی ناسازگاری‌های متنی طراحی شده است. اسناد را به جملات تقسیم کرده و با ترکیب امتیازهای جملات، امتیاز کلی را محاسبه می‌کند. این روش دقتی برابر با ۴.۷۴ درصد در شناسایی ناسازگاری‌ها ارائه می‌دهد

• این مدل بر اساس تقویت مدل‌های زبانی با دانش مبتنی بر پایگاه‌های دانش (DB) عمل می‌کند. سه استراتژی برای پیش‌آموزش مدل‌های زبانی شامل BERT، GPT-3 و GPT-4، به کار گرفته می‌شود. این مدل برای ارزیابی دقت در حوزه‌های علمی و خبری بسیار مؤثر است [۱۸].

این متریک بر شناسایی ناسازگاری‌ها در خلاصه‌ها با ایجاد مجموعه‌های داده مبتنی بر ویرایش استوار است. SUMMEDITS با توجه به دامنه‌های مختلف، امکان ارزیابی مدل‌ها را با چالش بیشتری فراهم می‌کند و معیار جدیدی برای توانایی استدلال واقعی مدل‌ها ارائه می‌دهد. SUMMEDITS با تمرکز بر شناسایی خطاهای معنایی، به عنوان یک ابزار مؤثر برای ارزیابی توانایی استدلال مدل‌ها در پردازش متن عمل می‌کند. این متریک نه تنها خطاها را شناسایی می‌کند بلکه برای توسعه مدل‌هایی که توانایی استنتاج دقیق‌تری دارند، کاربردی است [۱۷].

۳-۵-۱ محدودیت‌ها و پیشرفت‌ها

اگرچه معیارهایی مانند روژ^{۹۸} و امتیازبرت^{۹۹} در اندازه‌گیری شباهت زبانی میان خلاصه‌ها و متن مرجع مؤثر هستند، اما توانایی لازم برای شناسایی هالوسینیشن را ندارند. هالوسینیشن به تولید محتوای نامرتبط، ساختگی یا ناسازگار با متن اصلی اشاره دارد که می‌تواند وفاداری خلاصه‌ها را کاهش دهد و اعتماد کاربران به خروجی مدل را به چالش بکشد.

پژوهش‌های اخیر معیارهای پیشرفته‌تری مانند ریسک هزیان و SUMMEDITS را معرفی کرده‌اند که به طور خاص بر شناسایی هالوسینیشن و ناسازگاری‌های معنایی تمرکز دارند. این متریک‌ها با ارائه ارزیابی‌های جزئی‌تر و دقیق‌تر، امکان بهبود قابل توجه در کیفیت و دقت خلاصه‌سازی انتزاعی را فراهم کرده‌اند و نقاط ضعف معیارهای سنتی مانند روژ را پوشش می‌دهند.

^{۹۸}rouge

^{۹۹}BERTScore

فصل چهارم

روش ارائه شده

فصل پنجم

نتایج

کتاب نامه

- [1] Better language models and their implications.
- [2] Chen, Yen-Chun and Bansal, Mohit. Fast abstractive summarization with reinforcement selected sentence rewriting. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 675–686, 2018.
- [3] Child, Rewon, Gray, Scott, Radford, Alec, and Sutskever, Ilya. Generating long sequences with sparse transformers, 2019.
- [4] Dong, Zican, Tang, Tianyi, Li, Lunyi, and Zhao, Wayne Xin. A survey on long text modeling with transformers, 2023.
- [5] El-Kassas, Wafaa S., Salama, Cherif R., Rafea, Ahmed A., and Mohamed, Hoda K. Automatic text summarization: A comprehensive survey. Expert Systems with Applications, 165:113679, 2021.
- [6] Fan, Angela, Grangier, David, and Auli, Michael. Controllable abstractive summarization. In Proceedings of the 2nd Workshop on Neural Machine Translation and Generation, pages 45–54, Melbourne, Australia, July 2018. Association for Computational Linguistics.
- [7] Grathwohl, Will, Choi, Dami, Wu, Yuhuai, Roeder, Geoffrey, and Duvenaud, David Kristjanson. Backpropagation through the void: Optimizing control variates for black-box gradient estimation. ArXiv, abs/1711.00123, 2017.

- [8] Han, Zeyu, Gao, Chao, Liu, Jinyang, Zhang, Jeff, and Zhang, Sai Qian. Parameter-efficient fine-tuning for large models: A comprehensive survey. ArXiv, abs/2403.14608, 2024.
- [9] Hu, Edward J., Shen, Yelong, Wallis, Phillip, Allen-Zhu, Zeyuan, Li, Yuanzhi, Wang, Shean, Wang, Lu, and Chen, Weizhu. Lora: Low-rank adaptation of large language models, 2021.
- [10] Ishikawa, Kai, Ando, Shinichi, and Okumura, Akitoshi. Hybrid text summarization method based on the tf method and the lead method. In NTCIR Conference on Evaluation of Information Access Technologies, 2001.
- [11] Ji, Ziwei, Yu, Tiezheng, Xu, Yan, Lee, Nayeon, Ishii, Etsuko, and Fung, Pascale. Towards mitigating LLM hallucination via self reflection. In Bouamor, Houda, Pino, Juan, and Bali, Kalika, editors, Findings of the Association for Computational Linguistics: EMNLP 2023, pages 1827–1843, Singapore, December 2023. Association for Computational Linguistics.
- [12] Keneshloo, Yaser, Ramakrishnan, Naren, and Reddy, Chandan K. Deep transfer reinforcement learning for text summarization. ArXiv, abs/1810.06667, 2018.
- [13] King, Daniel, Shen, Zejiang, Subramani, Nishant, Weld, Daniel S., Beltagy, Iz, and Downey, Doug. Don’t say what you don’t know: Improving the consistency of abstractive summarization by constraining beam search. In Bosselut, Antoine, Chandu, Khyathi, Dhole, Kaustubh, Gangal, Varun, Gehrmann, Sebastian, Jernite, Yacine, Novikova, Jekaterina, and Perez-Beltrachini, Laura, editors, Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM), pages 555–571, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics.
- [14] Kitaev, Nikita, Kaiser, Lukasz, and Levskaya, Anselm. Reformer: The efficient transformer. In International Conference on Learning Representations, 2019.

- [15] Kryscinski, Wojciech, McCann, Bryan, Xiong, Caiming, and Socher, Richard. Evaluating the factual consistency of abstractive text summarization. In Webber, Bonnie, Cohn, Trevor, He, Yulan, and Liu, Yang, editors, Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 9332–9346, Online, November 2020. Association for Computational Linguistics.
- [16] Kryscinski, Wojciech, Paulus, Romain, Xiong, Caiming, and Socher, Richard. Improving abstraction in text summarization. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1808–1817, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.
- [17] Laban, Philippe, Kryscinski, Wojciech, Agarwal, Divyansh, Fabbri, Alexander, Xiong, Caiming, Joty, Shafiq, and Wu, Chien-Sheng. SummEdits: Measuring LLM ability at factual reasoning through the lens of summarization. In Bouamor, Houda, Pino, Juan, and Bali, Kalika, editors, Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 9662–9676. Association for Computational Linguistics.
- [18] Laban, Philippe, Schnabel, Tobias, Bennett, Paul N., and Hearst, Marti A. SummaC : Re-visiting NLI-based models for inconsistency detection in summarization. 10:163–177.
- [19] Lee, Chang-Shing, Jian, Zhi-Wei, and Huang, Lin-Kai. A fuzzy ontology and its application to news summarization. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 35(5):859–880, 2005.
- [20] Lester, Brian, Al-Rfou, Rami, and Constant, Noah. The power of scale for parameter-efficient prompt tuning, 2021.
- [21] Lewis, Mike, Liu, Yinhan, Goyal, Naman, Ghazvininejad, Marjan, Mohamed, Abdelrahman, Levy, Omer, Stoyanov, Veselin, and Zettlemoyer, Luke. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and

- comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7871–7880, Online, July 2020. Association for Computational Linguistics.
- [22] Li, Junyi, Tang, Tianyi, Zhao, Wayne Xin, and rong Wen, Ji. Pretrained language models for text generation: A survey. ArXiv, abs/2105.10311, 2021.
- [23] Li, Taiji, Li, Zhi, and Zhang, Yin. Improving faithfulness of large language models in summarization via sliding generation and self-consistency. In Calzolari, Nicoletta, Kan, Min-Yen, Hoste, Veronique, Lenci, Alessandro, Sakti, Sakriani, and Xue, Nianwen, editors, Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pages 8804–8817, Torino, Italia, May 2024. ELRA and ICCL.
- [24] Lin, Chin-Yew. ROUGE: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.
- [25] Liu, Linqing, Lu, Yao, Yang, Min, Qu, Qiang, Zhu, Jia, and Li, Hongyan. Generative adversarial network for abstractive text summarization. In Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.
- [26] Luhn, Hans Peter. The automatic creation of literature abstracts. IBM Journal of research and development, 2(2):159–165, 1958.
- [27] Ma, Tinghuai, Pan, Qian, Rong, Huan, Qian, Yurong, Tian, Yuan, and Al-Nabhan, Najla Abdulrahman. T-bertsum: Topic-aware text summarization based on bert. IEEE Transactions on Computational Social Systems, 9:879–890, 2022.
- [28] Malliaros, Fragkiskos D. and Skianis, Konstantinos. Graph-based term weighting for text categorization. In Proceedings of the 2015 IEEE/ACM International Conference on

- Advances in Social Networks Analysis and Mining 2015, ASONAM '15, page 1473–1479, New York, NY, USA, 2015. Association for Computing Machinery.
- [29] Maynez, Joshua, Narayan, Shashi, Bohnet, Bernd, and McDonald, Ryan. On faithfulness and factuality in abstractive summarization. In Jurafsky, Dan, Chai, Joyce, Schluter, Natalie, and Tetreault, Joel, editors, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1906–1919, Online, July 2020. Association for Computational Linguistics.
- [30] Men, Xin, Xu, Mingyu, Zhang, Qingyu, Wang, Bingning, Lin, Hongyu, Lu, Yaojie, Han, Xianpei, and Chen, Weipeng. Shortgpt: Layers in large language models are more redundant than you expect. arXiv e-prints, pages arXiv–2403, 2024.
- [31] Moratanch, N. and Chitrakala, S. A survey on abstractive text summarization. In 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), pages 1–7, 2016.
- [32] Nan, Feng, Nallapati, Ramesh, Wang, Zhiguo, Nogueira dos Santos, Cicero, Zhu, Henghui, Zhang, Dejiao, McKeown, Kathleen, and Xiang, Bing. Entity-level factual consistency of abstractive text summarization. In Merlo, Paola, Tiedemann, Jorg, and Tsarfaty, Reut, editors, Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 2727–2733. Association for Computational Linguistics.
- [33] Narendra, Andhale and Bewoor, Laxmi A. An overview of text summarization techniques. In 2016 international conference on computing communication control and automation (ICCUBEA), pages 1–7. IEEE, 2016.
- [34] Pang, Bo, Nijkamp, Erik, Kryscinski, Wojciech, Savarese, Silvio, Zhou, Yingbo, and Xiong, Caiming. Long document summarization with top-down and bottom-up inference. In Findings of the Association for Computational Linguistics: EACL 2023, pages 1237–1254, 2023.

- [35] Parnell, Jacob, Unanue, Inigo Jauregi, and Piccardi, Massimo. A multi-document coverage reward for relaxed multi-document summarization. In Annual Meeting of the Association for Computational Linguistics, 2022.
- [36] Pilault, Jonathan, Li, Raymond, Subramanian, Sandeep, and Pal, Christopher. On extractive and abstractive neural document summarization with transformer language models. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 9308–9319, 2020.
- [37] Shapira, Ori, Pasunuru, Ramakanth, Bansal, Mohit, Dagan, Ido, and Amsterdamer, Yael. Interactive query-assisted summarization via deep reinforcement learning. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2551–2568, Seattle, United States, July 2022. Association for Computational Linguistics.
- [38] Shapira, Ori, Pasunuru, Ramakanth, Ronen, Hadar, Bansal, Mohit, Amsterdamer, Yael, and Dagan, Ido. Extending multi-document summarization evaluation to the interactive setting. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 657–677, Online, June 2021. Association for Computational Linguistics.
- [39] Sherborne, Tom and Lapata, Mirella. Meta-learning a cross-lingual manifold for semantic parsing. Transactions of the Association for Computational Linguistics, 11:49–67, 2023.
- [40] Son, Seonil (Simon), Park, Junsoo, Hwang, Jeong-in, Lee, Junghwa, Noh, Hyungjong, and Lee, Yeonsoo. HaRiM \square : Evaluating summary quality with hallucination risk. In He, Yulan, Ji, Heng, Li, Sujian, Liu, Yang, and Chang, Chua-Hui, editors, Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language

- Processing (Volume 1: Long Papers), pages 895–924. Association for Computational Linguistics.
- [41] Su, DiJia, Su, Difei, Mulvey, John M., and Poor, H.Vincent. Optimizing multidocument summarization by blending reinforcement learning policies. *IEEE Transactions on Artificial Intelligence*, 4(3):416–427, 2023.
- [42] Su, Ming-Hsiang, Wu, Chung-Hsien, and Cheng, Hao-Tse. A two-stage transformer-based approach for variable-length abstractive summarization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2061–2072, 2020.
- [43] Vaswani, Ashish, Shazeer, Noam, Parmar, Niki, Uszkoreit, Jakob, Jones, Llion, Gomez, Aidan N, Kaiser, Lukasz, and Polosukhin, Illia. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [44] Xia, Yu, Liu, Xu, Yu, Tong, Kim, Sungchul, Rossi, Ryan A., Rao, Anup, Mai, Tung, and Li, Shuai. Hallucination diversity-aware active learning for text summarization, 2024.
- [45] Xiao, Yijun and Wang, William Yang. On hallucination and predictive uncertainty in conditional language generation. In Merlo, Paola, Tiedemann, Jorg, and Tsarfaty, Reut, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2734–2744, Online, April 2021. Association for Computational Linguistics.
- [46] Xiong, Wenhan, Gupta, Anchit, Toshniwal, Shubham, Mehdad, Yashar, and tau Yih, Wen. Adapting pretrained text-to-text models for long text sequences. *ArXiv*, abs/2209.10052, 2022.
- [47] Xu, Zhaozhuo, Liu, Zirui, Chen, Beidi, Zhong, Shaochen, Tang, Yuxin, Wang, Jue, Zhou, Kaixiong, Hu, Xia, and Shrivastava, Anshumali. Soft prompt recovers compressed llms, transferably. In *International Conference on Machine Learning*, 2024.

- [48] Zaheer, Manzil, Guruganesh, Guru, Dubey, Kumar Avinava, Ainslie, Joshua, Alberti, Chris, Ontanon, Santiago, Pham, Philip, Ravula, Anirudh, Wang, Qifan, Yang, Li, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020.
- [49] Zhang, Jingqing, Zhao, Yao, Saleh, Mohammad, and Liu, Peter. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020.
- [50] Zhang, Nan, Liu, Yanchi, Zhao, Xujiang, Cheng, Wei, Bao, Runxue, Zhang, Rui, Mitra, Prasenjit, and Chen, Haifeng. Pruning as a domain-specific llm extractor. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1417–1428, 2024.
- [51] Zhang, Tianyi, Ladhak, Faisal, Durmus, Esin, Liang, Percy, McKeown, Kathleen, and Hashimoto, Tatsunori B. Benchmarking large language models for news summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57, 2024.
- [52] Zhang, Yang, Li, Yawei, Wang, Xinpeng, Shen, Qianli, Plank, Barbara, Bischl, Bernd, Rezaei, Mina, and Kawaguchi, Kenji. Finercut: Finer-grained interpretable layer pruning for large language models. *arXiv e-prints*, pages arXiv–2405, 2024.
- [53] Zhou, Chunting, Neubig, Graham, Gu, Jiatao, Diab, Mona, Guzmán, Francisco, Zettlemoyer, Luke, and Ghazvininejad, Marjan. Detecting hallucinated content in conditional neural sequence generation. In Zong, Chengqing, Xia, Fei, Li, Wenjie, and Navigli, Roberto, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1393–1404, Online, August 2021. Association for Computational Linguistics.

واژه‌نامه‌ی انگلیسی به فارسی

ا	تابع زیان درست‌نمایی بیشینه maximum
	likelihood loss
الگوریتم حداکثر سازی نقطه-محصول فرا	
یادگیری Meta-Learned Dot-Product	ترای گرم trigram
Maximization	ترنسفورمر transformer
امتیازبرت BERTScore	تعبیه embedding
ب	تعبیه نشانه token embedding
بارت BART	تنظیم regularization
بازنمایی Representation	تنظیم دقیق پارامترها fine-tuning
بازیابی recall	توالی sequence
بیگ‌برد Big Bird	توجه با مرکزیت دوگانه two hub
پ	attention
پراکنده sparse	توجه به خود Self-attention
پیش‌آموزش pretraining	توجه جامع global attention
پگاسوس PEGASUS	توسعه deploy
ت	جامع global
	ج

gap sentences جملات فاصله‌افتاده	bidirectional دوسویه
GPT جی‌پی‌تی	ر
چ	ROUGE-L روژ-ال
polysemy چند معنایی	ROUGE-1 روژ-۱
ح	ROUGE-2 روژ-۲
long حافظه‌ی بلند مدت طولانی	modified cover- ریلکس
short-term memory networks(LSTM)	age reward along with a principled policy
informative حاوی اطلاعات مفید	gradient estimator (RELAX)
Maximal حداکثر ارتباط حاشیه‌ای	س
Marginal Relevance (MMR)	Lagrangian ساده‌سازی لاگرانژ
خ	relaxation
policy خط مشی	bias سوگیری
Autoregressive خود رگرسیون	ش
د	recurrent . . . شبکه‌های عصبی بازگشتی
maximum درست‌نمایی بیشینه	neural network (RNN)
likelihood	ع
locality- . . . درهم‌سازی حساس به مکان	agent عامل
sensitive hashing	communicating عامل تعامل کننده
Delta-ROUGE دلتا-روژ	agent
Divide-and-Conquer دنسر	lead-and-body . . . عبارت مقدمه و بدنه
(DANCER)	phrase
	task عمل

عمل پایین‌دست downstream task	نمایش نهفته latent representation
ف	و
فرا یادگیری Meta-Learning	واحد بازگشتی دروازه‌ای gated recurrent unit (GRU)
ک	
کوئری query	وظایف tasks
ل	
لایه‌های برگشت‌پذیر . . reversible layers	وظایف پایین‌دست . . downstream tasks
م	ه
مقاطع زبانی cross-lingual	هدف آموزش training objective
مجموعه‌ی دادگان corpus	هستان‌شناسی ontology
مدل موضوعی عصبی Neural Topic Model(NTM)	هم‌پوشانی بلوک‌های توجه . Overlapping attention windows
مدل مولد نقطه‌ای . . Pointer-Generator model	ی
مکانیزم توکن سراسری . . . Global-token mechanism	یادگیری تقویتی reinforcement learning
ن	یادگیری خط مشی . . . policy learning

واژه‌نامه‌ی انگلیسی به فارسی

A	فرآیند تصمیم‌گیری مارکوف محدود
Agent عامل	Constrained markov decision process (cmdp)
Autoregressive خودرگرسیون	Contextual network . شبکه‌ی محتوایی
B	Corpus مجموعه‌ی دادگان
Bart بارت	Cross-lingual متقاطع زبانی
Bertscore امتیازبرت	D
Bias سوگیری	Decoder کدگشا
Bidirectional دوسویه	Delta-rouge دلتا-روژ
Big bird بیگ‌برد	Deploy توسعه
Block- توجه به خود پراکنده‌ی بلوکی	Divide-and-conquer (dancer) . . . دنسر
sparse self-attention	Dot-product نقطه-محصول
C	Downstream tasks . . . وظایف پایین‌دست
Communicating عامل تعامل‌کننده	E
agent	Embedding تعبیه

Encoder کدگذار	Locality- . . مکان به حساسی درهم‌سازی
F	sensitive hashing (lsh)
Fine-tuning تنظیم دقیق پارامترها	Long حافظه‌ی بلند مدت طولانی
G	short-term memory networks(lstm)
Gap sentences جملات فاصله‌افتاده	M
Gated recurrent واحد بازگشتی دروازه‌ای	Maximum درست‌نمایی بیشینه
unit (gru)	likelihood
Global جامع	تابع زیان درست‌نمایی بیشینه
Global attention توجه جامع	likelihood loss
Global-token . . . مکانیزم توکن سراسری	فرا یادگیری
mechanism	الگوریتم حداکثر سازی نقطه-محصول فرا
Gpt جی‌پی‌تی	Meta-learned dot-product یادگیری
I	maximization
Informative حاوی اطلاعات مفید	Modified cover- ریلکس
L	age reward along with a principled policy
Lagrangian ساده‌سازی لاگرانژ	gradient estimator (relax)
relaxation	N
Latent representation . . . نمایش نهفته	Named entity موجودیت نامدار
Lead-and-body . . . عبارت مقدمه و بدنه	Neural topic مدل موضوعی عصبی
phrase	model(ntm)
	O
	Ontology هستان‌شناسی

هم‌پوشانی بلوک‌های توجه . Overlapping attention windows	لایه‌های برگشت‌پذیر . Reversible layers
P	روژ-۱ Rouge-1
پگاسوس Pegasus	روژ-۲ Rouge-2
مدل مولد نقطه‌ای . . . Pointer-generator model	روژ-ال Rouge-l
خط مشی Policy	S
یادگیری خط مشی . . Policy learning	تعبیه قطعه Segment embedding
چند معنایی Polysemy	توجه به خود Self-attention
خود توجهی مبتنی بر ادغام بلوکی تقویت شده Pooling-augmented blockwise attention	گرایان خط مشی انتقادی . . Self-critic policy gradient
پیش‌آموزش Pretraining	توالی Sequence
Q	پراکنده Sparse
کوئری Query	T
R	تی‌برت‌سام T-bertsum
بازیابی Recall	وظایف Tasks
شبکه‌های عصبی بازگشتی . . . Recurrent neural network (rnn)	فرکانس تکرار عبارت . . . Term frequency
تنظیم Regularization	تعبیه نشانه Token embedding
یادگیری تقویتی Reinforcement learning	هدف آموزش Training objective
بازنمایی Representation	ترنسفورمر Transformer
	ترای‌گرم Trigram
	توجه با مرکزیت دوگانه Two hub attention