دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده مهندسی کامپیوتر

پروژه تحقیقاتی درس سمینار

روش‌های خلاصه‌سازی انتزاعی

نگارش

زهرا زنجانی

استاد درس

دکتر رضا صفابخش

شهریور ۱۴۰۲


بسم الله الرحمن الرحيم

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

در این صفحه فرم دفاع یا تایید 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.

به نام خدا

دانشگاه صنعتی امیرکبیر
(پلی‌تکنیک تهران)

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 پایان نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

زهرا زنجانی

امضا

از استادار جمندم، خانم دکتر ممتازی، که با صبر و حوصله فراوان، راهنمایی های ارزشمندشان، و حمایت های بی دریغشان، نقش بسزایی در موفقیت این پژوهش داشتند، کمال تشکر و قدردانی را دارم.

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

فهرست مطالب

عنوان صفحه

فهرست تصاویر

شکل صفحه

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 ویژگی ها خلاصه استخراج می‌شود [۲۸]. 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.

۱-۲ چالش‌های خلاصه‌سازی‌متون طولانی

۱-۳ چشم‌انداز نوشتار

[1] Deep neural sequence to sequence models

[2] transformer

[3] Pretrained language models (PTLMs)

فصل دوم

معرفی مفاهیم

۲-۱ 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، ؟]

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.

۲-۲-۱ 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 آموزش‌داده‌شده (□□□) 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

[1] 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] [۲۳، ؟].

۳-۲-۲ روش های فشرده‌سازی مدل‌های بزرگ زبانی

با توجه به پیشرفت‌های سریع در مدل‌های زبانی بزرگ ،(□□□) شاهد رشد چشمگیر اندازه این مدل‌ها در سال‌های اخیر هستیم. این مدل‌ها که دارای میلیاردها و حتی تریلیون‌ها 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 قابل‌توجه، به‌طور کارآمدتر اجرا

شوند.

۲-۲-۵ کوانتایزاسیون

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.

۲-۲-۶ 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 آنلاین 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.

۲-۲-۷ 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 [۹].

۲-۲-۷-۱ لورا

لورا [۲] روشی برای بهبود کارایی و کاهش هزینه‌های آموزش مدل‌های زبانی بزرگ (□□□□) 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 [۱۱].

۲-۲-۷-۲ تنظیم دقیق پرامپت

تنظیم دقیق پرامپت □□□□□□ (□□□□□□ 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 [۲۱].

۲-۳ توهم

توهم یا هالوسینیشن [۳] در مدل‌های زبانی بزرگ (□□□) 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

[2] LoRA
[3] Hallucination

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 (□□□□□□□□□□□□□-□□□□) 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 نه‌تنها دقت و سازگاری خلاصه‌ها را بهبود می‌بخشند، بلکه موجب کاهش خطای هالوسینیشن در کاربردهای عملی می‌شوند.

فصل سوم

مرور تاریخچه

۳-۱ روش های مبتنی برساختار

روش‌های خلاصه‌سازی مبتنی بر ساختار از نخستین رویکردهای توسعه‌یافته در حوزه خلاصه‌سازی متون هستند. این روش‌ها با بهره‌گیری از ویژگی‌های ساختاری متن ورودی، به تولید خلاصه‌های مختصر و منسجم می‌پردازند. در این رویکرد، اطلاعات مهم متن به ساختاری از پیش تعریف‌شده تخصیص داده می‌شود و خلاصه بر اساس این ساختار ایجاد می‌گردد. با پیشرفت فناوری و ظهور روش‌های نوین، مانند مدل‌های زبانی پیشرفته و تکنیک‌های یادگیری عمیق، کارایی و دقت در خلاصه‌سازی متون بهبود یافته است. با این حال، روش‌های مبتنی بر ساختار همچنان به‌عنوان پایه و اساس درک و پردازش متون مورد استفاده قرار می‌گیرند. در این بخش روش‌های مبتنی بر درخت [۱] ، مبتنی بر قالب[۲]، مبتنی بر هستان‌شناسی[۳] ، عبارت مقدمه و بدنه[۴] ، مبتنی بر گراف[۵] و مبتنی بر قانون[۶] مورد بررسی قرار می‌گیرد.

۳-۱-۱ روش مبتنی بر درخت

روش‌های مبتنی بر درخت در خلاصه‌سازی متن از درخت‌های وابستگی برای نمایش ساختار نحوی اسناد متنی استفاده می‌کنند. ابتدا، متن مبدأ به درخت‌های وابستگی تبدیل می‌شود و سپس این درخت‌ها در یک ساختار واحد ادغام می‌شوند. در نهایت، با خطی‌سازی درخت ادغام‌شده، جملات جدیدی تولید می‌شوند. این فرآیند، که به آن «خطی‌سازی درخت» گفته می‌شود، به انتخاب تجزیه‌کننده و حفظ وابستگی‌های نحوی بین کلمات وابسته است که می‌تواند بر کارایی تأثیر بگذارد[۳۴].

۳-۱-۲ روش مبتنی بر قالب

این روش‌ها از قالب‌های از پیش تعریف‌شده برای نمایش اسناد استفاده می‌کنند. این قالب‌ها برای انطباق با الگوها و قوانین خاص در محتوای متنی طراحی شده‌اند و امکان استخراج اطلاعات مرتبط را فراهم می‌کنند که می‌توان آن‌ها را در چارچوب این قالب‌ها ترسیم کرد. این فرآیند شامل تطبیق متن با الگوها و قوانین مذکور برای شناسایی محتوای متناسب با الگو است. این روش به دلیل تولید خلاصه‌هایی که به ساختار و قالب‌های تعیین‌شده پایبند هستند، از انسجام بالایی برخوردار است. [۳۴].

[1] tree-based
[2] template-based
[3] ontology-based
[4] lead-and-body phrase
[5] graph-based
[6] rule-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. [۳۴]. 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 هستی شناسی طبقه بندی می‌شوند. مکانیزم استنتاج فازی درجه عضویت برای هر جمله را با توجه به طبقه‌بندی کننده محاسبه می‌کند [۲۰].

۴-۱-۳ 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. [۳۴]. 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. دستورها براساس اهمیت برای نوشتن خلاصه رتبه بندی می‌شوند [۱۳].

۵-۱-۳ 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 یال‌ها به یکدیگر متصل می‌شوند و وزن یال‌ها نشان‌دهنده قدرت این روابط است. سپس، با استفاده از الگوریتم‌های رتبه‌بندی گراف، اهمیت

[7] Term 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. [۳۴]. مالیروس و اسکیانیس 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 [۳۰].

۳-۱-۶ روش مبتنی بر قانون

در روش خلاصه‌سازی مبتنی بر قاعده، اسناد به دسته‌ها و جنبه‌های مختلف تقسیم می‌شوند. سپس، ماژول 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. به‌عبارت‌دیگر، 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[۳۳].

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 کدگذار-کدگشا

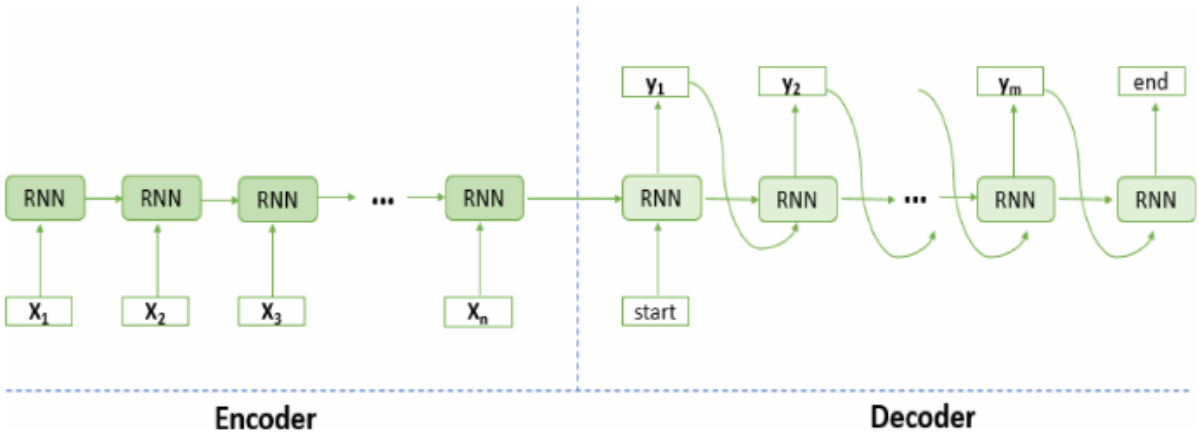
با پیشرفت‌های اخیر در پردازش زبان طبیعی، مدل‌های کدگذار-کدگشا به‌عنوان یکی از رویکردهای مؤثر در وظایف تولید متن، از جمله ترجمه ماشینی و خلاصه‌سازی متن، مطرح شده‌اند. این مدل‌ها با نگاشت ورودی به خروجی، امکان تولید نتایج مطلوب را فراهم می‌کنند. معماری کدگذار-کدگشا، که در شکل ۳-۱ نمایش داده شده است، اساس مدل‌های دنباله به دنباله را تشکیل می‌دهد.

در این ساختار، کدگذار وظیفه دارد ورودی‌ها را به یک نمایش داخلی تبدیل کند، در حالی که کدگشا از این نمایش داخلی برای تولید خروجی‌ها استفاده می‌کند. هر دو بخش از مکانیزم توجه بهره می‌برند که به مدل اجازه می‌دهد تمرکز خود را بر روی بخش‌های مهم‌تر ورودی یا خروجی تنظیم کند.

شبکه‌های عصبی بازگشتی [۸] و حافظه‌های کوتاه‌مدت طولانی [۹] برای پردازش داده‌های دنباله‌ای

[8] RNN
[9] LSTM

مانند متن طراحی شده‌اند و در این زمینه عملکرد مناسبی دارند. با این حال، این مدل‌ها در مدیریت وابستگی‌های طولانی‌مدت با چالش‌هایی مواجه هستند. برای غلبه بر این محدودیت‌ها، مدل‌های ترنسفورمر معرفی شدند که با استفاده از مکانیزم توجه، امکان پردازش موازی و مدیریت وابستگی‌های دوربرد را فراهم می‌کنند. در ادامه، به بررسی مدل‌های کدگذار-کدگشا و نقش آن‌ها در پیشرفت‌های اخیر پردازش زبان طبیعی می‌پردازیم.



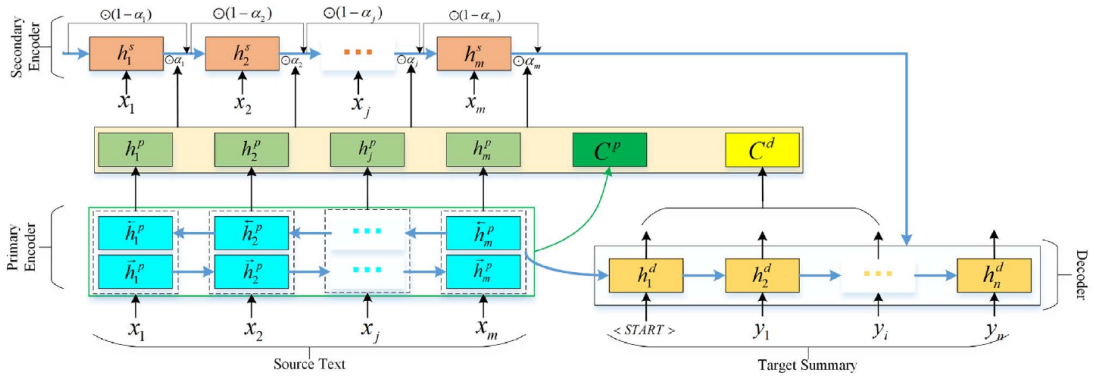
شکل ۳-۱: معماری پایه‌ی مدل کدگذار-کدگشا [۱۲]

یائو[10] و همکاران مدل کدگذاری دوگانه را برای خلاصه‌سازی انتزاعی پیشنهاد داده‌اند. این مدل برای درک بهتر روابط بین متن ورودی و خلاصه مرجع بازنمایی متن ورودی و بازنمایی خلاصه‌ی مرجع را می‌آموزد. همانطور که در شکل ۳-۲ نشان داده شده است. این مدل از یک کدگذار اولیه، یک کدگذار ثانویه و یک کدگشا مجهز به مکانیزم توجه تشکیل شده است و هر سه ماژول فوق از واحد بازگشتی دروازه‌ای [11] استفاده می‌کنند. کدگذار اولیه بردارهای معنایی هر کلمه در ترتیب ورودی را محاسبه می‌کند. کدگذار ثانویه وزن اهمیت هر کلمه در ترتیب ورودی و بردارهای معنایی مربوطه را دوباره محاسبه می‌کند. در نهایت کدگشا با مکانیسم توجه به صورت مرحله‌ای کدگشایی می‌کند و در هر مرحله یک توالی خروجی با طول ثابت جزئی ایجاد می‌کند. در این مدل کدگذار ثانویه عملیات کدگذاری را براساس ورودی هر مرحله و خروجی مرحله‌ی قبل انجام می‌دهد بنابراین کیفیت متون قبلی تولید شده توسط کدگشا بر خروجی‌های جدید تاثیر می‌گذارد [۴۷].

مدل کدگذاری دوگانه مدل سلسله مراتبی متغیر بر اساس مدل کدگذاری دوگانه برای خلاصه‌سازی

[10] Yao

[11] gated recurrent unit (GRU)



شکل ۳-۲: معماری پایه‌ی مدل دوگانه‌ی کدگذار [۴۷]

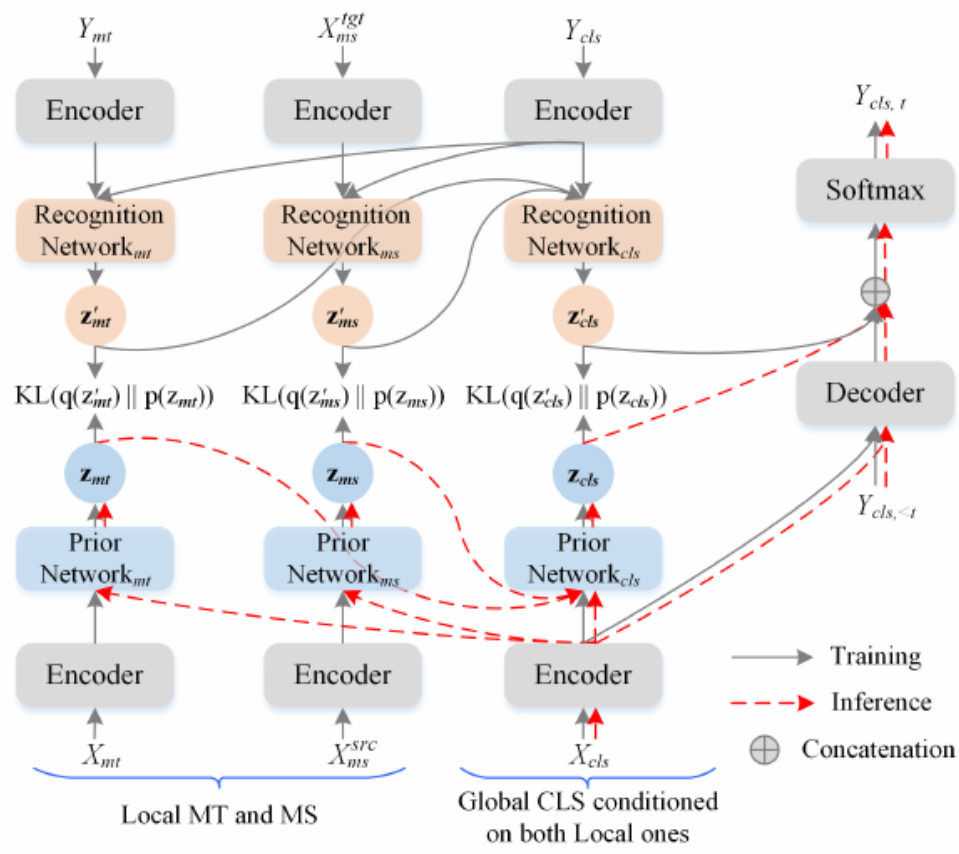
متقاطع زبانی[۱۲] پیشنهاد شده است. این مدل شامل دو متغیر نهفته محلی و یک متغیر نهفته جامع است. از متغیرهای نهفته محلی برای بازسازی ترجمه و خلاصه زبان مبدأ و از متغیر نهفته سراسری برای تولید خلاصه بین زبانی استفاده می‌شود. قسمت کد گذار این مدل دو بخش دارد که هر بخش وظیفه‌ی تولید یکی از متغیرهای نهفته محلی را دارد و بخش کدگشا با استفاده از نمایش‌های نهفته‌ی محلی خلاصه‌ی نهایی را تولید می‌کند. ساختار سلسله مراتبی این مدل به آن اجازه می‌دهد تا رابطه سلسله مراتبی بین ترجمه، خلاصه‌سازی و خلاصه‌سازی بین زبانی را بیاموزد [۲۵].

۳-۳ روش‌های مبتنی بر مدل ترنسفورمر

با ظهور ترنسفورمرها[۱۳] ، بهبودهای قابل توجهی در کیفیت نتایج خلاصه‌سازی خودکار به وجود آمد. ترنسفومرها با استفاده از مکانیزم توجه به خود[۱۴] 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 [۱۵] و همکاران یک مدل خلاصه‌سازی بر اساس مدل برت را پیشنهاد کرده‌اند. نویسندگان استدلال

[12] cross-lingual
[13] transformers
[14] Self-attention
[15] Pan



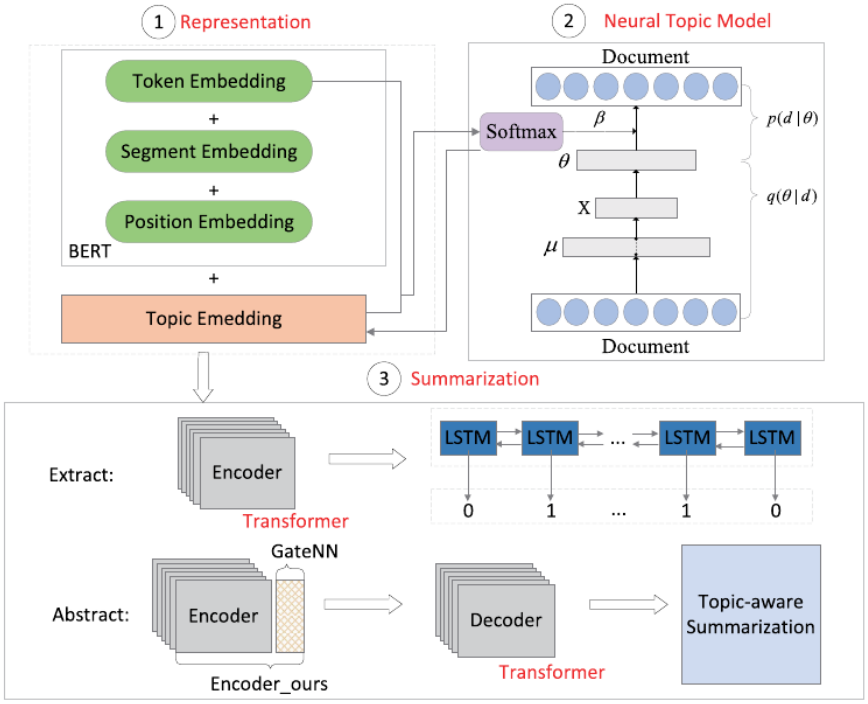
شکل ۳-۳: معماری پایه‌ی مدل سلسله مراتبی متغیر برای خلاصه‌سازی متقابل زبانی [۲۵]

متغیرهای محلی z_mt و z_ms به ترتیب برای ترجمه و خلاصه‌سازی و متغیر جامع z_cls برای خلاصه‌سازی بین زبانی طراحی شده‌اند. خطوط خاکستری نشان‌دهنده فرآیند آموزشی است که مسئول تولید (z'_{mt}، z'_{ms}، z'_{cls}) از توزیع پسین متناظر پیش‌بینی‌شده توسط شبکه است. خطوط قرمز خط چین نشان دهنده فرآیند استنتاج برای تولید نمایش‌های نهفته (z_{mt}، z_{ms}، z_{cls}) از توزیع‌های پیش بینی شده توسط شبکه‌های قبلی است.

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[۱۶] نامیده می‌شود از سه بخش ایجاد بازنمایی، مدل موضوعی عصبی[۱۷] و مدل خلاصه‌سازی تشکیل شده است. ساختار مدل را در شکل ۳-۴ نشان داده شده است.

همانطور که در شکل ۳-۵ نشان داده شده است، بازنمایی ایجاد شده برای هر جمله ورودی، با استفاده از یک شبکه‌ی ترنسفورمر دوسویه‌ی[۱۸] چند لایه و حاصل جمع چهار نوع تعبیه (تعبیه نشانه[۱۹] ، تعبیه

[16] T-BERTSum
[17] Neural Topic Model(NTM)
[18] bidirectional
[19] token embedding

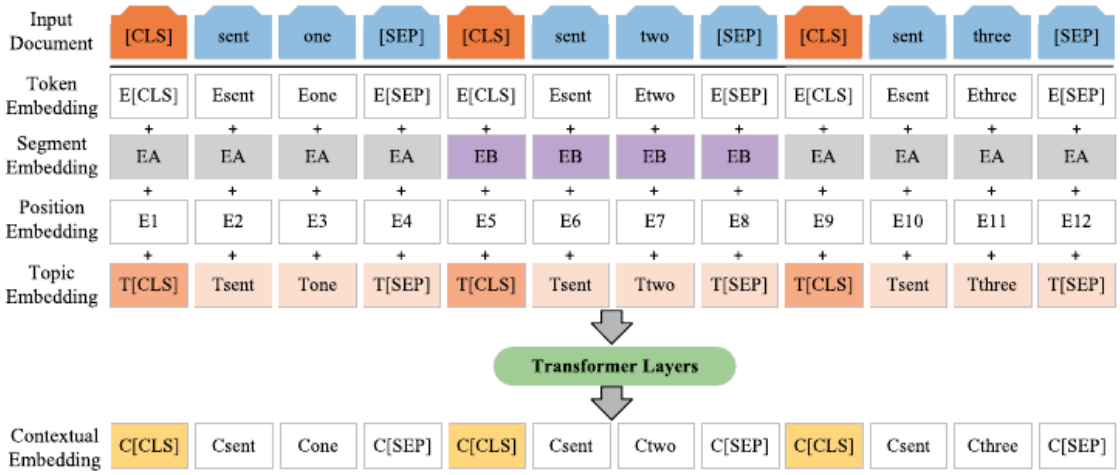


شکل ۳-۴: معماری مدل تی‌برت‌سام [۲۹]

قطعه[۲۰] ، تعبیه موقعیت و تعبیه موضوع) به دست می‌آید که تعبیه موضوع در این مقاله معرفی شده و سه تعبیه دیگر مشابه مدل برت 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.

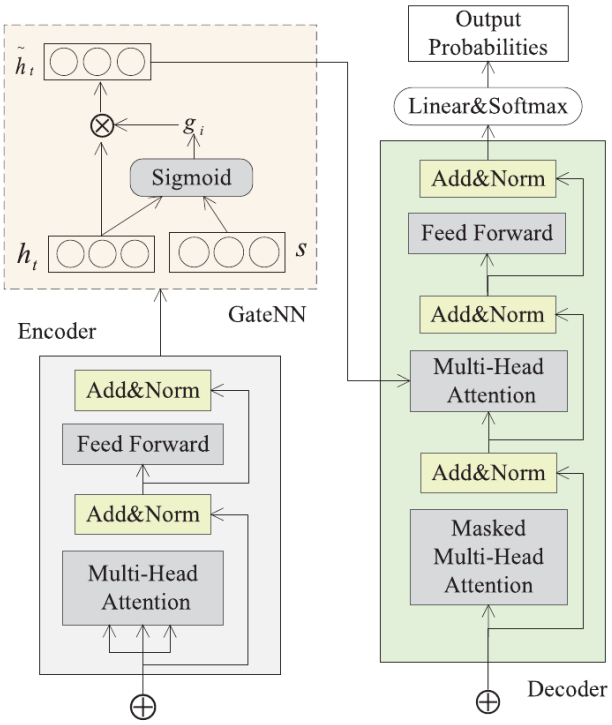
مدل موضوعی عصبی وظیفه‌ی ایجاد تعبیه موضوعی را دارد. این مدل دارای دو جزء است: یک شبکه مولد و یک شبکه استنتاج. شبکه مولد یک سند را به عنوان ورودی می‌گیرد و یک توزیع موضوعی را بر روی کلمات موجود در سند خروجی می‌دهد. شبکه استنتاج یک سند را به عنوان ورودی می‌گیرد و خروجی آن پارامترهای توزیع موضوع است. بخش 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 ۳-۶ 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

[20] segment embedding



شکل ۳-۵: تعبیه مدل تی‌برت‌سام [۲۹]

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.



شکل ۳-۶: معماری ترنسفورمر تی‌برت‌سام [۲۹]

این مدل شامل شبکه‌ی دروازه‌ای و کدگذار-کدگشا با توجه چند سر می‌باشد [۲۹]

اکثر مدل‌های خلاصه‌سازی انتزاعی موجود برای تولید خلاصه‌های با طول ثابت طراحی شده‌اند، بنابراین سو [۲۱] 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 ۳-۷مدل 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.

- 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 [۲۲] مهم‌ترین جمله را از هر بخش استخراج می‌شود.

- خلاصه‌سازی اسناد: با استفاده از جملات استخراج شده این ماژول 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 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.

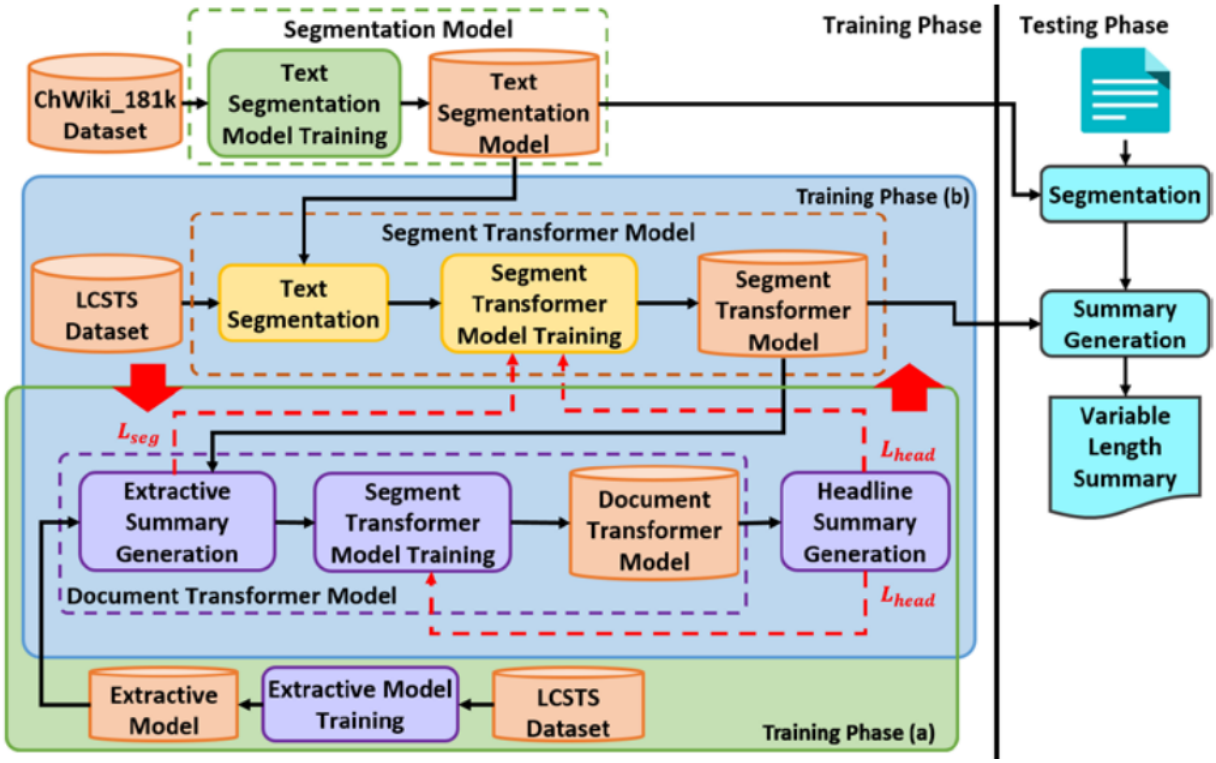
- ایجاد خلاصه با طول متغیر:پس از اینکه متن ورودی به بخش‌های مختلف تقسیم شد، هر بخش از ماژول خلاصه‌سازی بخش عبور می‌کند تا یک خلاصه انتزاعی مبتنی بر جمله ایجاد کند. سپس

[21] Ming-Hsiang Su

[22] BertSum

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 بخش های 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].

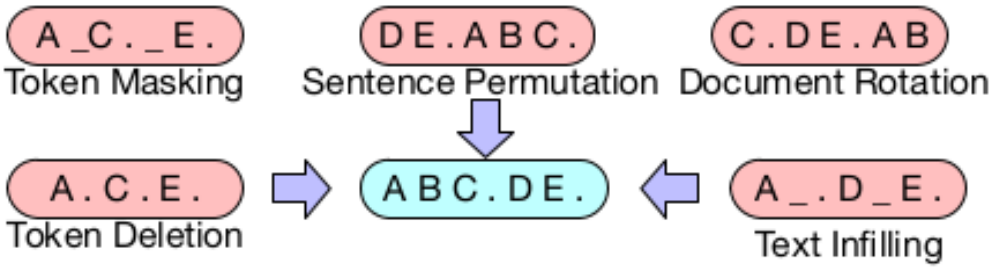


شکل ۳-۷: چهارچوب ایجاد خلاصه با طول متغیر [۴۲]

لوئیس و همکاران مدلی با نام بارت [۲۳] 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 [۲۴] دانست. بارت در عملیات تولید متن، مانند ترجمه ماشینی یا خالصه‌سازی انتزاعی متن، و همچنین در فهم متن کاربرد دارد و با استفاده از اهداف کدگذاری خودکار حذف نویز آموزش می‌بینند. برای پیش‌آموزش بارت نواقصی در سندهای ورودی ایجاد می‌شود و با بهینه کردن تابع زیان آنتروپی-متقاطع [۲۵] بین خروجی‌های کدگشا و سند اولیه، متن بازسازی

[23] BART
[24] GPT
[25] cross-entropy

می‌شود. همانطور که در شکل۳-۸ نشان داده شده است این مدل طیف گسترده‌ای از نویز ها 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 [۲۲].



شکل ۳-۸: عمل‌های پیش‌آموزش بارت [۲۲]

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[۲۶] توسط ژنگ 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[۲۷] از قبل آموزش داده شده است [۴۹].

این مدل دو روش پیش‌آموزش را معرفی کرده است که در ادامه به شرح آنها می‌پردازیم:

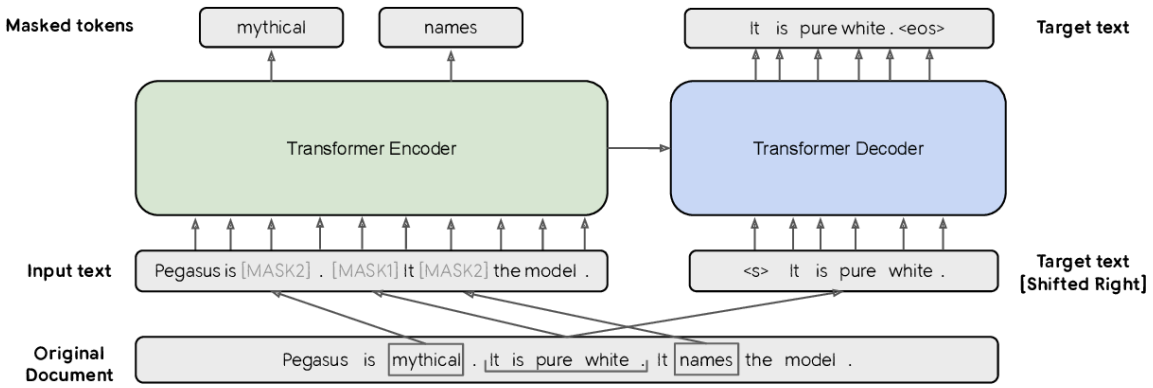
۱. تولید جملات فاصله‌افتاده : فرضی مطرح شده است که اگر عمل پیش‌آموزش مدل به وظایف پایین‌دست[۲۸] نزدیک‌تر باشد، نتیجه نهایی بهتر و همچنین تنظیم دقیق پارامترها [۲۹] سریع‌تر خواهد بود. با توجه به این که این مدل قرار است فقط برای خلاصه‌سازی انتزاعی متن استفاده شود، عمل پیش‌آموزش مشابه تولید متن‌های خلاصه از یک سند ورودی تعریف شده‌است. 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: mجمله به صورت تصادفی از متن انتخاب شده و پنهان می‌شوند.

[26] PEGASUS
[27] gap sentences generation
[28] downstream task
[29] fine-tuning

- انتخاب جملات اول متن: mجمله اول متن پنهان می‌شوند زیرا اغلب جملات ابتدای متن نسبت به جملات بعدی مهم‌ترهستند.
- انتخاب جملات مهم متن: برای انتخاب m جمله مهم متن از تقریب معیار ارزیابی روژ-۱ استفاده می‌شود. به ازای هر جمله از متن، یک دوتایی از آن جمله و متن سند فاقد آن جمله ساخته شده و ارزیابی می‌شود که چقدر ممکن است این جمله، خلاصه سند فاقد آن جمله باشد. جملاتی که امتیاز بالاتر گرفته‌اند از نظر خلاصه بودن مهم‌تر هستند و پنهان می‌شوند.

۲. مدل زبانی پوشیده شده: مشابه مدل برت ۵۱ درصد از توکن‌های متن ورودی انتخاب می‌شوند و سپس ۸۰ درصد از این توکن‌ها، با توکن $[MASK2]$ و۱۰ درصد توکن‌ها با یک توکن تصادفی جایگزین می‌شوند. ۱۰ درصد دیگر بدون تغییر باقی می‌ماند. شکل ۳-۹ اعمال همزمان این دو عمل، یعنی تولید جمالت فاصله افتاده و مدل زبانی پوشیده شده را بر روی یک ورودی نشان می‌دهد.



شکل ۳-۹: ساختار مدل پگاسوس [۴۹]

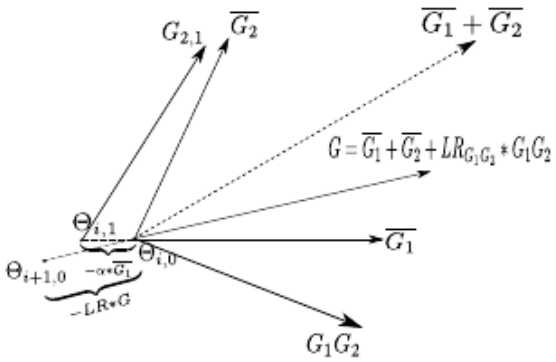
معماری پایه پگاسوس یک کدگذار-کدگشا ترنسفورمر استاندارد است. جملات فاصله‌افتاده و مدل زبانی پوشیده شده به طور همزمان در این مثال به عنوان اهداف پیش‌آموزش اعمال می‌شوند. در اصل سه جمله وجود دارد. یک جمله با $[MASK1]$ پوشانده شده و به عنوان متن تولید هدف جملات فاصله‌افتاده استفاده می‌شود. دو جمله دیگر در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

کدیا [۳۰] و همکاران الگوریتم حداکثرسازی نقطه-محصول فرایادگیری (امدات)[۳۱] را برای بهبود پگاسوس پیشنهاد دادند. این الگوریتم بر اساس ایده به حداکثر رساندن حاصل‌ضرب نقطه‌ای بین

[30] Kedia

[31] Meta-Learned Dot-Product Maximization (MDot)

گرادیان‌های مدل در نقاط مختلف آموزش با استفاده از تکنیکی به نام تفاوت‌های محدود[۳۲] است. این الگوریتم از نظر محاسباتی کارآمد است و می‌تواند برای مدل‌های بزرگ مانند برت اعمال شود و سربار محاسباتی را کاهش بدهد [۴۰]. عملکرد مناسب مدل پگاسوس در خلاصه‌سازی متون باعث شده بهترین مدل خلاصه‌سازی متون کوتاه مبتنی بر مدل پگاسوس و تکنیک تنظیم[۳۳] امدات باشد.



شکل ۳-۱۰: الگوریتم امدات [۴۰]

محاسبه گرادیان برای به حداکثر رساندن محصول نقطه‌ای با استفاده از تقریب تفاضل محدود و استفاده از آن برای تنظیم گرادیان استاندارد [۴۰].

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

یکی از مشکلات مدل ترنسفومر 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[۳۴] برای حل چالش‌های محاسباتی مرتبط با پردازش دنباله‌های طولانی متن ارائه شده است. لایه‌های برگشت‌پذیر[۳۵] 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

[32] finite differences
[33] regularization
[34] Reformer
[35] reversible layers

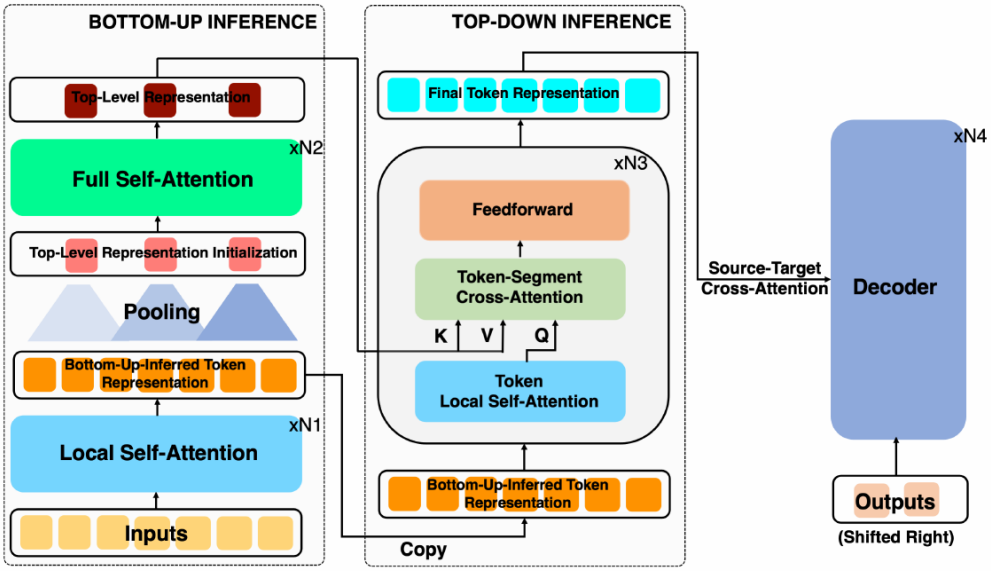
استفاده از درهم‌سازی حساس به مکان[۳۶] در مکانیسم توجه منجر به محاسبه توجه کارآمدتر می‌شود. درهم‌سازی حساس به مکان با توجه به زیرمجموعه‌ای از نشانه‌ها بر اساس مقادیر هش آنها، محاسبه توجه کامل را تقریب می‌زند. علاوه بر این، ریفورمر از کدگذاری‌های موقعیت محوری برای کدگذاری اطلاعات موقعیت توکن‌ها به صورت 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 [۱۷].

شبکه‌ی ترنسفورمر پراکنده[۳۷] 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[۳۸] با استفاده از مکانیزم توجه پراکنده[۳۹] عملکرد ترنسفورمر را در مواجه با دنباله‌ی کلمات طولانی بهبود می‌بخشد، نوآوری‌های دیگری مانند توجه جامع[۴۰] 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].

در سال‌های اخیر ایده‌های مختلفی برای بهبود کیفیت خروجی مدل خلاصه‌سازی خودکار اسناد بلند ارائه شده است. به عنوان مثال پایل[۴۱] 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[۴۲] 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 [۳۷]. پانگ[۴۳] و همکاران یک ساختار سلسله مراتبی برای اسناد طولانی فرض کرده‌اند. در این ساختار سطح

[36] locality-sensitive hashing (LSH)
[37] sparse
[38] Big Bird
[39] Sparse attention
[40] global attention
[41] Pilault
[42] GPT-2
[43] Pang

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 ۳-۱۱ نشان داده شده است. روش پیشنهادی یک رویکرد جدید امیدوارکننده برای خلاصه‌سازی اسناد طولانی است و نسبت به روش‌های قبلی کارآمدتر و موثرتر است [۳۵].



شکل ۳-۱۱: معماری مدل ترنسفورمر از بالا به پایین [۳۵]

جیدیوتیس و همکاران شیوه‌ی تقسیم و غلبه (دنسر)[۴۴] را برای بهبود خلاصه‌سازی اسناد طولانی پیشنهاد کرده اند.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 ساختاربندی شده‌اند.

برای مطابقت هر قسمت از خلاصه با بخشی از سند در دنسر از معیار روژ[۴۵] استفاده می‌شود. در این روش معیار روژ-ال بین هر یک از جملات خلاصه و تمام جملات سند محاسبه می‌شود و هر جمله‌ی

[44] Divide-and-ConquER (DANCER)

[45] ROUGE

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 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[۴۶] 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[۴۷] 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

[46] Pointer-Generator model

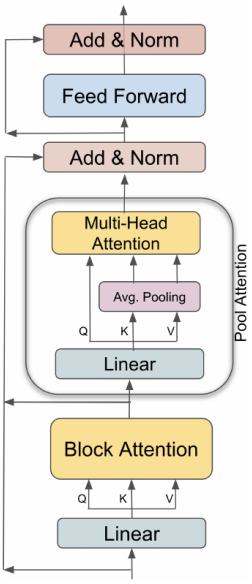
[47] pretraining corpus

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[۴۸] در قسمت کدگذار است. این مکانیزم امکان استفاده‌ی مجدد از وزن‌های مدل‌های از پیش آموزش دیده را فراهم می‌کند.
- مکانیرم توکن سراسری[۴۹]: 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[۵۰]: 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 نیمی‌از 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[۵۱]: این لایه به عنوان جایگزین لایه خود توجهی برای اتصالات دوربرد معرفی شده است. این رویکرد به واحدهای توجه درون بلوک‌ها اجازه می‌دهد تا به جای توجه به همسایگان بلافصل خود، بر خلاصه‌ای از اطلاعات کلی در بلوک‌ها تمرکز کنند. این لایه در تصویر ۳-۱۲نشان داده شده است. این مدل را قادر می‌سازد تا از اطلاعات گسترده تری در سراسر سند برای تصمیم گیری 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 تمامی‌وظایف 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

[48] Block-sparse self-attention
[49] Global-token mechanism
[50] Overlapping attention windows
[51] Pooling-augmented blockwise attention



شکل ۳-۱۲: لایه خودتوجهی تقویت شده ادغام شده [۴۶]

ی یادگیری تقویتی[۵۲] 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 تصمیم گیری 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 طور کلی، یادگیری تقویتی در چهار حوزه مختلف برای بهبود خلاصه‌سازی خودکار استفاده می‌شود:

۳-۴ روش‌های مبتنی بر یادگیری تقویتی

۳-۴-۱ 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[۵۳] بهبود یافته است. علاوه بر این مدل‌های

[52] reinforcement learning
[53] policy learning

دنباله به دنباله عمیق را نمی‌توان برای خلاصه کردن طیف گسترده ای 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 ازگرادیان خط مشی انتقادی [۵۴] و ترکیب آن با یادگیری انتقالی [۵۵] برای انتقال دانش از یک مجموعه داده به مجموعه دیگر برطرف کنند[۱۵].

فریم‌ورک پوبرل [۵۶] (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 حاشیه ای [۵۷] 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 بهینه سازی هر هدف به صورت جداگانه استفاده می کند و خط مشی های جداگانه ای را برای اهمیت، ارتباط و طول می آموزد[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 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.

سلیکیلماز [۵۸] و همکاران مدل کدگذار-کدگشای چندعامله را برای بهبود خلاصه‌سازی اسناد طولانی با استفاده از عامل تعامل‌کننده [۵۹] 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

[54] self-critic policy gradient
[55] Transfer Learning (TL)
[56] PoBRL
[57] Maximal Marginal Relevance (MMR)
[58] Celikyilmaz
[59] communicating agent

مدل‌های دنباله به دنباله را بهبود می‌بخشد.

۳-۴-۲ یادگیری تقویتی برای ترکیب خلاصه‌های استخراجی و انتزاعی

از یادگیری تقویتی برای ترکیب ویژگی‌های استخراجی با خلاصه انتزاعی برای استفاده از هر دو نوع خلاصه ی 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 استخراج‌کننده و انتزاعی، آنها را انتزاع می‌کنند. به عنوان مثال لیو[۶۰] 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[۲۷]. همچنین چن و بانسال[۶۱] یک مدل خلاصه‌سازی سریع پیشنهاد کردند که جملات برجسته را استخراج می‌کرد و سپس با استفاده از گرادیان خط مشی سطح جمله مبتنی بر یادگیری‌تقویتی بازنویسی می‌کرد[۲]. کریسینسکی[۶۲] 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 آموزش‌دیده، و بهبود معیار جدید از طریق یادگیری خط‌مشی. تکنیک اول شامل یک شبکه‌ی محتوایی[۶۳] و یک مدل زبانی از پیش آموزش دیده است. شبکه‌ی محتوایی 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 مسئولیت ها امکان استخراج بهتر و تولید جملات مختصر را فراهم می کند. 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 [۱۹].

۳-۴-۳ 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[۶۴] مورد انتقاد قرار گرفته است. درست‌نمایی بیشینه کیفیت خلاصه‌ی تولید شده

[60] Liu
[61] Chen and Bansal
[62] Kryscinski
[63] contextual network
[64] maximum likelihood

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 [۳۶]. معیارهای ارزیابی خلاصه‌سازی مانند روژ-۱[۶۵]، روژ-۲[۶۶]، روژ-ال[۶۷] و امتیازبرت[۶۸] به عنوان پاداش در رویکردهای یادگیری تقویتی استفاده شده است. با این حال، پارنل و همکاران استدلال می‌کند 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)[۶۹] را پیشنهاد می‌دهند[۳۶، ؟]. ریلکس یک برآوردگر گرادیان خط مشی[۷۰] با واریانس کم و بدون سوگیری[۷۱] است که برای مسائل یادگیری تقویتی با فضاهای کنش مداوم، مانند خلاصه‌سازی متن، مناسب است[۸].

در عبارت۳-۱ 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 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. $p(ys)$ 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.

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

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

درخلاصه‌سازی 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

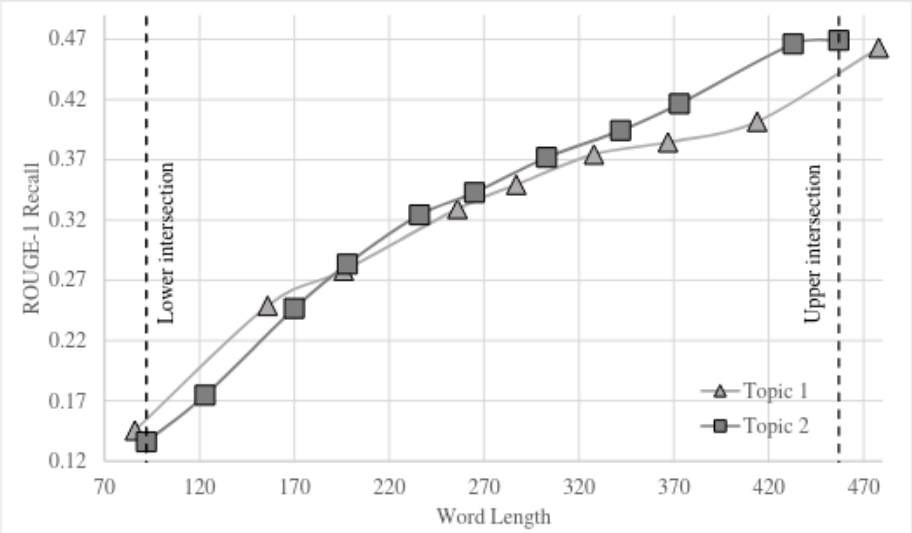
[65] ROUGE-1
[66] ROUGE-2
[67] ROUGE-L
[68] BERTScore
[69] modified coverage reward along with a principled policy gradient estimator (RELAX)
[70] policy
[71] bias

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 تولیدشده نشان بدهند.

مدل کنترل‌سام [۷۲] 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 [۷۳] است [۷]. این هدف آموزش هیچ سیگنال نظارتی صریحی ندارد. برای حل این مشکل چان [۷۴] و همکاران با اعمال محدودیت بر روی هدف آموزشی با استفاده از فرآیند تصمیم‌گیری مارکوف محدود [۷۵] یک چهارچوب خلاصه سازی پیشنهاد کرده‌اند که شامل یک تابع پاداش همراه با مجموعه‌ای از محدودیت ها است و 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 ، مبزان 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 [۷۶] و برای بهینه سازی از الگوریتم بهینه سازی مبتنی بر گرادیان، مانند ادام 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 [۷۷] ابتدا ارجاع موجودیت نامدار به سند اضافه می‌شود سپس یک محدودیت سوال و جواب اعمال می‌شود. این محدودیت بر روی امتیاز اف-۱

[72] controlSum
[73] maximum liklihood loss
[74] Chan
[75] Constrained Markov Decision Process (CMDP)
[76] Lagrangian relaxation
[77] named entity

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 ترای‌گرم [۷۸] و موجودیت‌های درخواستی برای افزایش خوانایی و کاهش تکرار در متن اعمال می‌شود. مدل اینتسام [۷۹] یک مدل خلاصه‌سازی تعاملی با هدف خلاصه کردن اطلاعات مهم بر اساس کوئری‌های [۸۰] 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 [۸۱] 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 ۳-۱۳ نمایش داده شده است [۳۹].



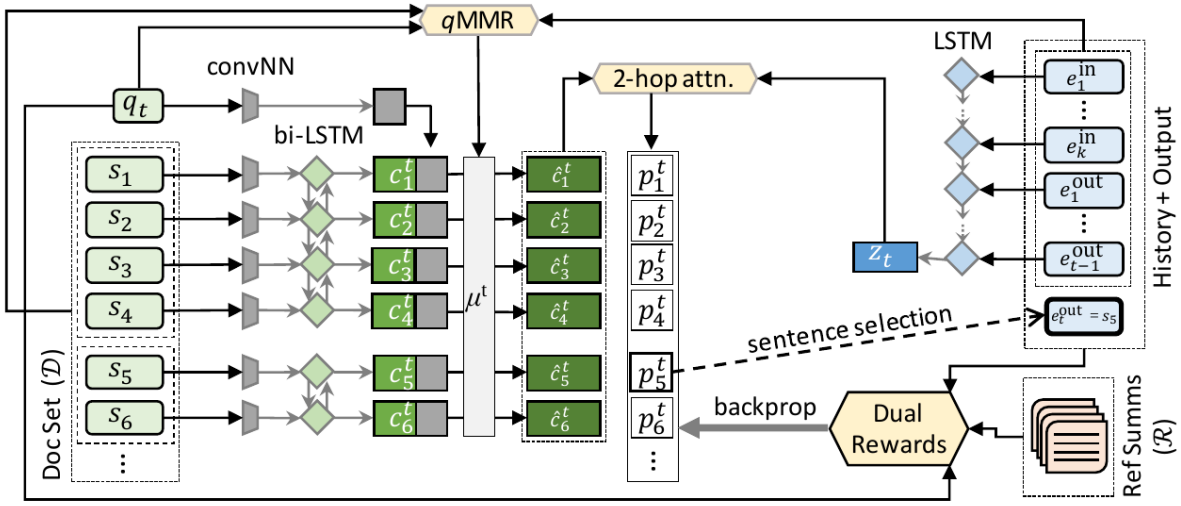
شکل ۳-۱۳: یک نمونه از نمودار منحنی بازیابی بر اساس طول [۳۹]
این نمودار دو تعامل متفاوت با سیستم خلاصه‌سازی را مقایسه می‌کند. هر نقطه نمایانگر خروجی هر مرحله تعامل با کاربر است.

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

[78] trigram
[79] IntSumm
[80] query
[81] recall

ارائه داده‌اند. ورودی این مدل مجموعه‌ی اسناد، کوئری و تاریخچه‌ی تعاملات با کاربر به همراه خروجی 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 [۸۲] بر اساس کدگذاری به دست آمده از تاریخچه و مدل $RL-MMR$

توزیع احتمال هر جمله را به دست می‌آید. مدل امسام[۸۳] یک مدل خودرگرسیون[۸۴] است که برای آموزش آن از یادگیری تقویتی به همراه مکانیسم پاداش دوگانه استفاده می‌شود. معیار دلتا-روژ[۸۵] 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 مکانیزیم پاداش دوگانه تعادل موجب ایجاد تعادل اطلاعات می‌شود. ساختار مدل در شکل ۳-۱۴ نشان داده شده است [۳۸].



شکل ۳-۱۴: معماری مدل امسام [۳۸]

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

[82] two hub attention
[83] MSumm
[84] Autoregressive
[85] Delta-ROUGE

مفاهیم برجسته در کل سند [۸۶] . 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 ؟؟.

۳-۵ روش های مبتنی بر مدل‌های زبانی بزرگ و چالش‌ها

۳-۵-۱ توهم در مدل های زبانی بزرگ

زیوی جی و همکارانش به بررسی مسئله توهم 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 توهم و افزایش قابلیت اطمینان سیستم‌ها در پاسخگویی به سوالات پزشکی است[۱۴].

. ییجون شیاو و همکارانش رابطه بین توهم و عدم‌قطعیت پیش‌بینی در تولید زبان شرطی را بررسی کرده‌اند. آن‌ها با تاکید بر نقش مهم عدم‌قطعیت اپیستمیک، روشی برای بهبود الگوریتم جستجوی پرتو پیشنهاد می‌دهند که با کاهش عدم‌قطعیت، میزان توهمات 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 توهمات را کاهش دهد، بدون آنکه بر روانی متن تاثیر زیادی بگذارد[۱۶].

[86] global salient content identification

، جاشوا ماینز و همکارانش به بررسی مسئله توهم 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 "توهم درونی" و "توهم بیرونی" طبقه‌بندی می‌کنند. این مقاله از طریق ارزیابی انسانی نشان می‌دهد که بیش از ۷۰٪ خلاصه‌ها شامل محتوای توهم‌آمیز هستند که اکثریت آن‌ها به‌ویژه در توهم بیرونی، نادرست می‌باشند. نویسندگان همچنین پیشنهاد می‌کنند که مدل‌های از پیش آموزش‌دیده مانند $BERTS2S$ نسبت به مدل‌های دیگر، خلاصه‌های دقیق‌تر و با توهم کمتر تولید می‌کنند. این پژوهش با ارائه معیارهایی جدید برای ارزیابی دقت و اعتبار خلاصه‌ها، مسیر بهتری برای بهبود ارزیابی‌های خودکار و روش‌های تولید خلاصه‌سازی باز می‌کند[۳۱]. جورج کریسوستومو و همکارانش تأثیر هرس مدل‌های زبان بزرگ بر کاهش توهم در خلاصه‌سازی متون را بررسی کرده‌اند. آن‌ها با استفاده از روش‌های پیشرفته هرس، مانند $SparseGPT$ و $Wanda$، نشان داده‌اند که مدل‌های هرس‌شده نسبت به مدل‌های اصلی توهم کمتری دارند. این مدل‌ها بیشتر بر متن منبع تکیه کرده و خلاصه‌هایی با همپوشانی واژگانی بالاتر و محتوای واقع‌گرایانه‌تر تولید می‌کنند. آزمایش‌ها روی پنج مجموعه داده مختلف و چندین مدل، کاهش قابل‌توجه ریسک توهم را با افزایش میزان هرس نشان داده است. این پژوهش، استفاده از مدل‌های هرس‌شده را به‌عنوان راهکاری مؤثر برای کاهش توهم در خلاصه‌سازی پیشنهاد می‌کند.

مسئله توهم در مدل‌های زبان بزرگ یکی از چالش‌های اساسی در تولید زبان طبیعی است که می‌تواند اعتبار و اطمینان به این مدل‌ها را تحت تأثیر قرار دهد. استفاده از روش‌های مبتکرانه و دقیق می‌تواند به طور قابل توجهی میزان توهم را کاهش داده و انسجام و دقت خروجی‌های مدل را بهبود بخشد. این پیشرفت‌ها راه را برای استفاده ایمن‌تر و موثرتر از مدل‌های زبان در کاربردهای حساس، از جمله پزشکی و خلاصه‌سازی متون، هموار می‌کند.

۳-۵-۲ هرس مدل های زبانی

در حوزه بهینه‌سازی مدل‌های زبانی بزرگ، روش‌های مختلفی برای کاهش پیچیدگی محاسباتی و منابع مورد نیاز ارائه شده‌اند. فانگ و همکاران الگوریتمی برای هرس ساختاریافته مدل‌های زبانی بزرگ ارائه داده‌اند 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

می‌شود، سپس اهمیت هر بخش وابسته به‌صورت مستقل از وظایف [87] ارزیابی می‌شود و در نهایت، با استفاده از روش لورا، عملکرد مدل با حداقل داده بازیابی می‌شود. این روش توانسته با هرس ۲۰ درصد از مدل، ۹۴ درصد از کارایی اولیه را حفظ کند. نتایج این تحقیق نشان می‌دهد که رویکرد پیشنهادی با شناسایی و هرس بخش‌های وابسته مدل به صورت داده‌محور و بدون نیاز به به‌روزرسانی وزن‌ها، روشی کارآمد و سریع برای کاهش پیچیدگی مدل‌های زبانی بزرگ ارائه می‌دهد[۳۲].

ژانگ و همکاران الگوریتم $D-PRUNER$را معرفی کرده‌اند. این الگوریتم با هدف ارائه یک روش 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 [88] 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 [89] 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 [90]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 [91] 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

[87] task-agnostic
[88] regularization term
[89] perplexity
[90] redundancy
[91] hidden states

و امکان کاهش بیشتر محاسبات و پارامترها را فراهم می‌کند.

$$BI_i = 1 - \mathbb{E}_{X,t}\left[\frac{X_{i,t}^{\top} X_{i+1,t}}{\|X_{i,t}\|_2 \|X_{i+1,t}\|_2}\right] \quad (۲-۳)$$

در این رابطه‌ی ۳-۲ $X_{i,t}$ 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. $\|\cdot\|_2$ نُرم اقلیدسی را نشان می‌دهد و $\mathbb{E}_{X,t}$ بیانگر امید 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 [۹۲] را برای هرس لایه‌های مدل‌های زبانی بزرگ ارائه می‌دهد. این روش برخلاف روش‌های پیشین، لایه‌های خودتوجهی[۹۳] و شبکه عصبی پیشخور[۹۴] 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 ۲۵٪ از لایه‌های لاما ۳-B۸، ۹۰٪ از عملکرد مدل حفظ می‌شود و با حذف ۳۰٪ از لایه‌های لاما ۳-B۷۰، مدل توانسته است ۹۵٪ 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].

[92] FINERCUT
[93] self-attention
[94] FFN

۳-۵-۳ توهم در خلاصه‌سازی

یکی از مطالعات برجسته در زمینه کاهش توهم در مدل‌های زبانی بزرگ و بهبود کیفیت خلاصه‌سازی انتزاعی متون طولانی، مقاله‌ای از یو شیا است. این تحقیق اولین چارچوب یادگیری فعال برای کاهش توهم 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 $FactKB$، $UniEval$ و $BERT-P$ برای ارزیابی این خطاها استفاده می‌کند.

الگوریتم $HADAS$، با رویکردی داده‌محور و متنوع‌محور، نمونه‌های داده‌ای را که شامل انواع مختلفی از توهمات هستند برای بهبود مدل انتخاب می‌کند. این انتخاب با استفاده از واگرایی جنسن-شانون (JSD) به منظور سنجش تنوع توزیع انواع توهمات انجام می‌شود. این روش نه تنها به طور مؤثری توهمات 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 توهم و افزایش دقت در خلاصه‌سازی مدل‌های زبان بزرگ ارائه داده‌اند. این روش شامل سه مرحله کلیدی است:

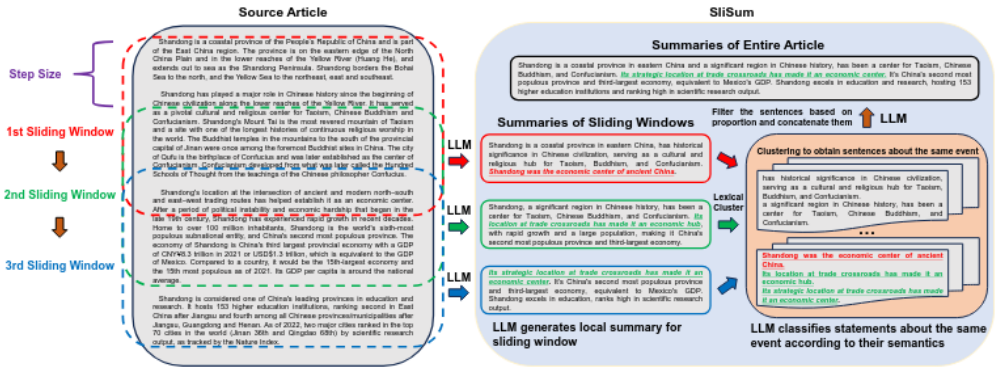
- تولید خلاصه‌های محلی با پنجره‌های لغزنده: متن منبع به بخش‌های همپوشانی‌شده تقسیم می‌شود (پنجره‌های لغزنده)، و مدل زبان برای هر پنجره یک خلاصه محلی تولید می‌کند. این همپوشانی‌ها کمک می‌کنند که متن به طور عادلانه در سراسر مقاله پردازش شود و مشکل سوگیری موقعیتی کاهش یابد.

- 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 توهم و افزایش دقت و انسجام خلاصه‌های تولیدشده می‌شود و در متون کوتاه و بلند عملکردی مؤثر نشان می‌دهد. تصویر ۳-۱۵ نشان‌دهنده مراحل مختلف این فرآیند است که شامل پردازش هر پنجره و ترکیب نتایج به روش ساختاریافته است[۲۴].



شکل ۳-۱۵: 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[۲۴].

۳-۶ 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 [۹۵] ، فراخوان [۹۶] 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 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، ۲۶].

- 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[۹۷] 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 امتیازبرت توانایی بهتری در ارزیابی هالوسینیشن دارد[۱۸].

۳-۶-۱ 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، مانند $Maynez$ و همکاران روش‌های جایگزینی را معرفی کرده‌اند که مستقیماً هالوسینیشن را در سطح توکن شناسایی می‌کنند و به تحلیل دقیق‌تر ناهماهنگی‌های موجود در متن

[95] Precision
[96] Recall
[97] Natural Language Inference(NLI)
[98] rouge
[99] BERTScore

می‌پردازند. این پیشرفت‌ها با ارائه ارزیابی‌های جزئی‌تر، امکان بهبود قابل‌توجه در کیفیت و دقت خلاصه‌سازی انتزاعی را فراهم کرده‌اند[۳۱].

فصل چهارم

روش‌ارائه شده

فصل پنجم

نتایج

کتاب‌نامه

[1] Better language models and their implications.

[2] Chen, Yen-Chun and Bansal, Mohit. Fast abstractive summarization with reinforce-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] Elman, Jeffrey L. Finding structure in time. Cognitive science, 14(2):179–211, 1990.

[7] 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.

[8] 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.

[9] 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.

[10] Hochreiter, Sepp and Schmidhuber, Jürgen. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

[11] 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.

[12] Idris, Norisma, Alomari, Ayham, Sabri, Aznul Qalid Md, and Alsmadi, Izzat. Deep reinforcement and transfer learning for abstractive text summarization: A review. Computer Speech and Language, 71:101276, 2022.

[13] 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.

[14] 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.

[15] Keneshloo, Yaser, Ramakrishnan, Naren, and Reddy, Chandan K. Deep transfer reinforcement learning for text summarization. ArXiv, abs/1810.06667, 2018.

[16] 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.

[17] Kitaev, Nikita, Kaiser, Lukasz, and Levskaya, Anselm. Reformer: The efficient transformer. In International Conference on Learning Representations, 2019.

[18] 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.

[19] 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.

[20] 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.

[21] Lester, Brian, Al-Rfou, Rami, and Constant, Noah. The power of scale for parameter-efficient prompt tuning, 2021.

[22] Lewis, Mike, Liu, Yinhan, Goyal, Naman, Ghazvininejad, Marjan, Mohamed, Abdelrahman, Levy, Omer, Stoyanov, Veselin, and Zettlemoyer, Luke. BART: Denois-

ing 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.

[23] Li, Junyi, Tang, Tianyi, Zhao, Wayne Xin, and rong Wen, Ji. Pretrained language models for text generation: A survey. ArXiv, abs/2105.10311, 2021.

[24] 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.

[25] Liang, Yunlong, Meng, Fandong, Zhou, Chulun, Xu, Jinan, Chen, Yufeng, Su, Jinsong, and Zhou, Jie. A variational hierarchical model for neural cross-lingual summarization. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2088–2099, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[26] 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.

[27] 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.

[28] Luhn, Hans Peter. The automatic creation of literature abstracts. IBM Journal of research and development, 2(2):159–165, 1958.

[29] 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.

[30] 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.

[31] 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.

[32] 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.

[33] 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.

[34] 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.

[35] Pang, Bo, Nijkamp, Erik, Kryscin?ski, 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.

[36] 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.

[37] 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.

[38] 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.

[39] 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.

[40] 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.

[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] Yao, Kaichun, Zhang, Libo, Du, Dawei, Luo, Tiejian, Tao, Lili, and Wu, Yanjun. Dual encoding for abstractive text summarization. IEEE transactions on cybernetics, 50(3):985–996, 2018.

[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

تابع زیان درست‌نمایی بیشینه maximum liklihood loss

ترای‌گرم . . . . . trigram

ترنسفورمر . . . . . transformer

تعبیه . . . . . embedding

تعبیه نشانه . . . . . token embedding

تنظیم . . . . . regularization

تنظیم دقیق پارامترها . . . . . fine-tuning

توالی . . . . . sequence

توجه با مرکزیت دوگانه . . . . . two hub attention

توجه به خود . . . . . Self-attention

توجه جامع . . . . . global attention

توسعه . . . . . deploy

ج

جامع . . . . . global

ا

الگوریتم حداکثر سازی نقطه-محصول فرا یادگیری . . . . . Meta-Learned Dot-Product Maximization

امتیازبرت . . . . . BERTScore

ب

بارت . . . . . BART

بازنمایی . . . . . Representation

بازیابی . . . . . recall

بیگ‌برد . . . . . Big Bird

پ

پراکنده . . . . . sparse

پیش‌آموزش . . . . . pretraining

پگاسوس . . . . . PEGASUS

ت

دوسویه . . . . . bidirectional

ر

روژ-ال . . . . . ROUGE-L

روژ-۱ . . . . . ROUGE-1

روژ-۲ . . . . . ROUGE-2

ریلکس . . . . . modified cover-age reward along with a principled policy gradient estimator (RELAX)

س

ساده‌سازی لاگرانژ . . . . . Lagrangian relaxation

سوگیری . . . . . bias

ش

شبکه‌های عصبی بازگشتی . . . recurrent neural network (RNN)

ع

عامل . . . . . agent

عامل تعامل کننده . . . . communicating agent

عبارت مقدمه و بدنه . . . lead-and-body phrase

عمل . . . . . task

جملات فاصله‌افتاده . . . . gap sentences

جی‌پی‌تی . . . . . GPT

چ

چند معنایی . . . . . polysemy

ح

حافظه‌ی بلند مدت طولانی . . . . . long short-term memory networks(LSTM)

حاوی اطلاعات مفید . . . . . informative

حداکثر ارتباط حاشیه ای . . . . Maximal Marginal Relevance (MMR)

خ

خط مشی . . . . . policy

خود رگرسیون . . . . . Autoregressive

د

درست‌نمایی بیشینه . . . . . maximum likelihood

درهم‌سازی حساس به مکان . . . locality-sensitive hashing

دلتا-روژ . . . . . Delta-ROUGE

دنسر . . . . . Divide-and-ConquER (DANCER)

نمایش نهفته . . . . latent representation

و

واحد بازگشتی دروازه‌ای gated recurrent unit (GRU)

وظایف . . . . . tasks

وظایف پایین‌دست . . downstream tasks

ه

هدف آموزش . . . . . training objective

هستان‌شناسی . . . . . ontology

هم‌پوشانی بلوک‌های توجه . Overlapping attention windows

ی

یادگیری تقویتی reinforcement learning

یادگیری خط مشی . . . policy learning

عمل پایین‌دست . . . . downstream task

ف

فرا یادگیری . . . . . Meta-Learning

ک

کوئری . . . . . query

ل

لایه‌های برگشت‌پذیر . . reversible layers

م

متقاطع زبانی . . . . . cross-lingual

مجموعه‌ی دادگان . . . . . corpus

مدل موضوعی عصبی . . . . Neural Topic Model(NTM

مدل مولد نقطه‌ای . . Pointer-Generator model

مکانیرم توکن سراسری . . . Global-token mechanism

ن

واژه‌نامه‌ی انگلیسی به فارسی

A

Agent . . . . . عامل

Autoregressive . . . . . خودرگرسیون

B

Bart . . . . . بارت

Bertscore . . . . . امتیازبرت

Bias . . . . . سوگیری

Bidirectional . . . . . دوسویه

Big bird . . . . . بیگ‌برد

Block- . . توجه به خود پراکنده‌ی بلوکی
sparse self-attention

C

Communicating . . . عامل تعامل‌کننده
agent

فرآیند تصمیم‌گیری مارکوف محدود
Constrained markov decision process
(cmdp)

Contextual network . شبکه‌ی محتوایی

Corpus . . . . . مجموعه‌ی دادگان

Cross-lingual . . . . . متقاطع زبانی

D

Decoder . . . . . کدگشا

Delta-rouge . . . . . دلتا-روژ

Deploy . . . . . توسعه

Divide-and-conquer (dancer) . . . دنسر

Dot-product . . . . . نقطه-محصول

Downstream tasks . . وظایف پایین‌دست

E

Embedding . . . . . تعبیه

Encoder . . . . . کدگذار

F

Fine-tuning . . . . . تنظیم دقیق پارامترها

G

Gap sentences . . . . جملات فاصله‌افتاده

Gated recurrent unit (gru) واحد بازگشتی دروازه‌ای

Global . . . . . جامع

Global attention . . . . . توجه جامع

Global-token mechanism . . . مکانیرم توکن سراسری

Gpt . . . . . جی‌پی‌تی

I

Informative . . . . . حاوی اطلاعات مفید

L

Lagrangian relaxation . . . . . سازه‌سازی لاگرانژ

Latent representation . . . نمایش نهفته

Lead-and-body phrase . . . عبارت مقدمه و بدنه

Locality-sensitive hashing (lsh) . . درهم‌سازی حساس به مکان

Long short-term memory networks(lstm) . . . . . حافظه‌ی بلند مدت طولانی

M

Maximum likelihood . . . . . درست‌نمایی بیشینه

Maximum liklihood loss تابع زیان درست‌نمایی بیشینه

Meta-learned . . . . . فرا یادگیری

Meta-learned dot-product maximization . . . . . الگوریتم حداکثر سازی نقطه-محصول فرا یادگیری

Modified cover-age reward along with a principled policy gradient estimator (relax) . . . . . ریلکس

N

Named entity . . . . . موجودیت نامدار

Neural topic model(ntm . . . . مدل موضوعی عصبی

O

Ontology . . . . . هستان‌شناسی

هم‌پوشانی بلوک‌های توجه . Overlapping attention windows

P

پگاسوس . . . . . Pegasus

مدل مولد نقطه‌ای . . . Pointer-generator model

خط مشی . . . . . Policy

یادگیری خط مشی . . Policy learning

چند معنایی . . . . . Polysemy

خود توجهی مبتنی بر ادغام بلوکی تقویت شده . . . . . Pooling-augmented blockwise attention

پیش‌آموزش . . . . . Pretraining

Q

کوئری . . . . . Query

R

بازیابی . . . . . Recall

شبکه‌های عصبی بازگشتی . . . Recurrent neural network (rnn)

تنظیم . . . . . Regularization

یادگیری تقویتی . . . . . Reinforcement learning

بازنمایی . . . . . Representation

لایه‌های برگشت‌پذیر . Reversible layers

روژ-۱ . . . . . Rouge-1

روژ-۲ . . . . . Rouge-2

روژ-ال . . . . . Rouge-l

S

تعبیه قطعه . . . . Segment embedding

توجه به خود . . . . . Self-attention

گرادیان خط مشی انتقادی . . Self-critic policy gradient

توالی . . . . . Sequence

پراکنده . . . . . Sparse

T

تی‌برت‌سام . . . . . T-bertsum

وظایف . . . . . Tasks

فرکانس تکرار عبارت . Term frequency

تعبیه نشانه . . . . . Token embedding

هدف آموزش . . . . . Training objective

ترنسفورمر . . . . . Transformer

ترای‌گرم . . . . . Trigram

توجه با مرکزیت دوگانه . . . . . Two hub attention