سال چهاردهم، شماره‌های ۵۱ و ۵۲ ، بهار و تابستان ۱۴۰۱
صص: ۵۵_۷۲

دو فصلنامه علمی
فناوری اطلاعات و ارتباطات ایران

مدلی جدید برپایه معماری کدگذار-کدگشا و سازوکار توجه برای خلاصه‌سازی چکیده‌ای خودکار متون

حسن علی اکبرپور[*] محمد تقی منظوری[*×] امیر مسعود رحمانی[***]**

[*] دانشجوی دکتری دانشگاه آزاد اسلامی، واحد علوم و تحقیقات، گروه مهندسی کامپیوتر، تهران، ایران

[**] دانشیار گروه مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران

[***] استاد دانشگاه آزاد اسلامی، واحد علوم و تحقیقات، گروه مهندسی کامپیوتر، تهران، ایران

تاریخ دریافت: ۱۴۰۰/۰۱/۱۱ تاریخ پذیرش: ۱۴۰۰/۱۰/۰۴

نوع مقاله: پژوهشی

چکیده

با گسترش وب و در دسترس قرار گرفتن حجم زیادی از اطلاعات در قالب اسناد متنی، توسعه سیستم‌های خودکار خلاصه‌سازی متون به‌عنوان یکی از موضوعات مهم در پردازش زبان‌های طبیعی در مرکز توجه محققان قرار گرفته است. البته با معرفی روش‌های یادگیری عمیق در حوزه پردازش متن، خلاصه‌سازی متون نیز وارد فاز جدیدی از توسعه شده و در سال‌های اخیر نیز استخراج خلاصه چکیده‌ای از متن با پیشرفت قابل‌توجهی مواجه شده است. اما می‌توان ادعا کرد که تاکنون از همه ظرفیت شبکه‌های عمیق برای این هدف استفاده نشده است و نیاز به پیشرفت در این حوزه توأمان با در نظر گرفتن ویژگی‌های شناختی همچنان احساس می شود. در این راستا، در این مقاله یک مدل دنباله‌ای مجهز به سازوکار توجه کمکی برای خلاصه‌سازی چکیده‌ای متون معرفی شده است که نه‌تنها از ترکیب ویژگی‌های زبانی و بردارهای تعبیه به‌عنوان ورودی مدل یادگیری بهره می‌برد بلکه برخلاف مطالعات پیشین که همواره از سازوکار توجه در بخش کدگذار استفاده می‌کردند، از سازوکار توجه کمکی در بخش کدگذار استفاده می‌کند. به کمک سازوکار توجه کمکی معرفی‌شده که از سازوکار ذهن انسان هنگام تولید خلاصه الهام می‌گیرد، بجای اینکه کل متن ورودی کدگذاری شود، تنها قسمت‌های مهم‌تر متن کدگذاری شده و در اختیار کدگشا برای تولید خلاصه قرار می‌گیرند. مدل پیشنهادی همچنین از یک سوئیچ به همراه یک حد آستانه در کدگشا برای غلبه بر مشکل با کلمات نادر بهره می‌برد. مدل پیشنهادی این مقاله روی دو مجموعه داده CNN/Daily Mail و DUC-۲۰۰۴ مورد آزمایش قرار گرفت. بر اساس نتایج حاصل از آزمایش‌ها و معیار ارزیابی ROUGE، مدل پیشنهادی از دقت بالاتری نسبت به سایر روش‌های موجود برای تولید خلاصه چکیده‌ای روی هر دو مجموعه داده برخوردار است.

واژگان کلیدی: یادگیری عمیق، خلاصه‌سازی چکیده‌ای، ، معماری کدگذار-کدگشا، سازوکار توجه کمکی، ویژگی‌های زبانی.

۱. مقدمه

با توجه به رشد قابل‌توجه اینترنت در سال‌های اخیر، روزانه حجم زیادی از اطلاعات در قالب متن پیرامون موضوعات مختلف در سطح وب گسترده جهانی تولید می‌شود. این اطلاعات که معلول توسعه تکنولوژی‌های اطلاعاتی و ارتباطی است، حجم عظیمی از اطلاعات دیجیتالی مناسب را در اختیار افراد، سازمان‌ها و دولت‌ها قرار می‌دهد که دسترسی سریع به مفاهیم اصلی این اطلاعات می‌تواند نقش قابل‌توجهی در تصمیم‌گیری آن‌ها ایفا کند. با اینکه این اطلاعات می‌توانند راهگشای بسیاری از مسائل و تسهیل‌کننده تصمیم‌گیری‌ها

نویسنده مسئول: محمدتقی منظوری manzuri@sharif.edu

باشند، با افزایش حجم مستندات متنی و توسعه منابع اطلاعاتی، تکنیک‌های بازیابی اطلاعات دیگر به‌تنهایی قادر به پاسخگویی نیازهای اطلاعاتی کاربران نیستند و حجم بالای اطلاعات همانند شمشیر دو لبه‌ای است که از یک‌سو می‌تواند برطرف‌کننده نیاز اطلاعاتی کاربران باشد و از سوی دیگر یافتن اطلاعات مطلوب را با چالش‌هایی مواجه کند [۱, ۲].

افزایش چشمگیر این نوع اطلاعات، وجود ابزارهایی برای ارزیابی خودکار منابع متنی را بیش از هر زمان دیگری آشکار می‌کند. در این میان وجود ابزارهایی که به‌صورت خودکار اطلاعات مفید و خلاصه را از میان منابع مختلف در کمترین زمان استخراج کند، بیش از هر زمان دیگری احساس می‌شود و این مسئله محققان را در سال‌های اخیر به‌سوی طراحی و توسعه سیستم‌های خودکار خلاصه‌سازی متون هدایت کرده است [۳, ۴]. خلاصه‌سازی متون یکی از موضوعات مهم در حوزه پردازش زبان طبیعی است و به‌عنوان هسته مرکزی طیف گسترده‌ای از ابزارهای پردازشگر متن مانند خلاصه‌سازهای ماشینی، سیستم‌های تصمیم‌یار، سیستم‌های پاسخ‌گو و موتورهای جستجو به‌حساب می‌آید [۵, ۶].

روش‌های خلاصه‌سازی بر اساس روش تهیه خلاصه به دو دسته خلاصه استخراجی[۱] و چکیده‌ای[۲] تقسیم می‌شوند [۷]. خلاصه‌سازی استخراجی از روش‌های سنتی برای تولید خلاصه استفاده می‌کند که در آن قسمت‌های مهم متن اصلی جدا شده و دوباره باهم ترکیب می‌شوند تا یک خلاصه منسجم ایجاد شود. در مقابل، خلاصه‌سازی چکیده‌ای خلاصه را بدون توجه به عبارات و جملات موجود در متن اصلی تولید کرده و هدف آن تولید خلاصه‌ای مطابق با خلاصه تولیدشده توسط انسان است. با در نظر گرفتن این مسئله که ساختار نحوی و معنایی دو فاکتور اصلی در ارزیابی سیستم‌های خلاصه‌سازی به شمار می‌آیند، هیچ‌کدام از این دو نوع خلاصه نمی‌توانند هم‌زمان روی هر دو فاکتور تمرکز کنند [۸]. در خلاصه‌سازی استخراجی خلاصه‌های تولیدشده اگرچه از جملات خود متن استخراج می‌شوند و ساختار نحوی درستی دارند، اما عموماً از لحاظ معنایی منسجم نیستند. علت این مشکل را می‌توان منوط به این مسئله دانست که جملاتی که در متن خلاصه کنار هم قرار می‌گیرند، لزوماً جملاتی نیستند که در متن اصلی کنار یکدیگر قرار دارند. از طرف دیگر، در خلاصه‌سازی چکیده‌ای اگرچه خلاصه تولیدشده از لحاظ معنایی منسجم است، اما ساختار نحوی جملات تولیدشده همواره درست نیستند. مزیت این مدل خلاصه‌سازی این است که خلاصه‌های تولیدشده از لحاظ معنایی به دلیل اینکه ترتیب[۳] بین کلمات را در نظر گرفته و دنباله‌ای از کلمات کلیدی را بر اساس ترتیب بین کلمات پس از آموزش تولید می‌کنند، فراگیر بوده و بین جملات تولیدشده ارتباط معنایی برقرار است. اما مشکل این مدل خلاصه‌سازی این است جملات تولیدشده که از دنباله‌های از کلمات معمولاً دارای ساختار نحوی مناسبی نمی‌باشند [۹].

با اینکه تحقیقات متعددی در طول زمان پیرامون خلاصه‌سازی خودکار متون صورت گرفته است، عمده رویکردهای خلاصه‌سازی معطوف به خلاصه‌سازی استخراجی بوده است. درواقع، تحقیقات انجام‌شده در این حوزه بسیار متنوع بوده و روش‌ها و رویکردهای بسیاری از قبیل انواع روش‌های مبتنی بر قاعده[۴]، روش‌های آماری[۵] و یادگیری ماشین[۶]، روش‌های مبتنی بر تحلیل‌های گفتمانی[۷] و روش‌های مبتنی بر گراف[۸] برای خلاصه‌سازی استخراجی به کار گرفته شده است. در مقایسه با این حجم از تحقیقات که در حوزه خلاصه‌سازی استخراجی انجام‌شده، مطالعات انجام‌شده پیرامون خلاصه‌سازی چکیده‌ای نسبتاً محدود است. علت اینکه بیشتر تحقیقات بر خلاصه‌سازی استخراجی متمرکز بوده و نه خلاصه‌سازی چکیده‌ای این است که خلاصه‌سازی چکیده‌ای با مسئله تولید زبان[۹] سروکار دارد که بسیار دشوار و مستعد خطاهای معنایی و گرامری بسیاری است [۱۰].

اما با معرفی روش‌های یادگیری عمیق در حوزه پردازش متن، خلاصه‌سازی خودکار متون نیز وارد فاز جدیدی از توسعه شد و در سال‌های گذشته خلاصه‌سازی چکیده‌ای با پیشرفت قابل‌توجهی مواجه شده است. مدل‌های خلاصه‌سازی چکیده‌ای که به دقت بالایی در سال‌های اخیر دست یافتند، عموماً بر اساس معماری کدگذار-کدگشا بودند که عموماً از سازوکار توجه نیز برای افزایش دقت کدگشا در تولید خلاصه استفاده می‌کردند [۹]. بر اساس این معماری، کدگذار متن اصلی را خوانده و معنای متن را درک می‌کند و متن کدگذاری شده را در اختیار کدگشا قرار می‌دهد. کدگشا با استفاده از مدل زبانی کلمات خلاصه را تولید کرده و وظیفه سازوکار توجه نیز تراز کردن متن ورودی و خلاصه خروجی برای تولید خلاصه با قابلیت اطمینان بالاتر است. همان‌طور که مشخص است، در این رویکردها به ویژگی‌های شناختی و عملکرد ذهنی انسان هنگام تولید خلاصه توجهی نشده است و کل متن ورودی که ممکن است حاوی اطلاعات اضافی و غیرضروری در تولید خلاصه باشد، کدگذاری شده و در اختیار کدگشا برای تولید خلاصه قرار می‌گیرد. در این حالت، اگر متن اصلی حاوی اطلاعات زیادی باشد که به خلاصه ربطی نداشته باشند، رمزگذار نمی‌تواند به‌درستی معنای متن را کدگذاری کند و در نتیجه،

[۱] Extractive
[۲] Abstractive
[۳] Collocation
[۴] Rule-based
[۵] Statistical
[۶] Machine learning
[۷] Discourse
[۸] Graph-based
[۹] Language generation

کدگشا تحت تأثیر اطلاعات نامرتبط خواهد بود و پیرو آن خلاصه‌های تولیدشده نیز ممکن است حاوی نکات برجسته متن ورودی نباشند [۱۱]. مشکل دیگر مدل‌های خلاصه‌سازی چکیده‌ای مبتنی بر یادگیری عمیق، نادیده گرفتن تأثیر کلمات نادر در تولید خلاصه است [۱۲].

برای غلبه بر این چالش‌ها در مدل پیشنهادی این مقاله یک سازوکار توجه کمکی با الهام از ذهن و شناخت انسان هنگام تولید خلاصه معرفی شده است. به‌طورکلی، دانشمندان علوم شناختی ذهن انسان را شبکه پیچیده‌ای می‌دانند که اطلاعات را دریافت، نگهداری و بازیابی می‌کند و می‌تواند آن را تغییر شکل یا انتقال دهد. .به بیان کلی‌ترو از دید علوم شناختی، ذهن را می‌توان با تساهل نوعی رایانه دانست که در آن اطلاعات توسط دستگاه‌های ورودی مانند صفحه کلید یا نشانگر وارد سامانه می‌شود و سپس عملیات متفاوتی برروی آن انجام می‌گیرد. اطلاعات وارد شده را می‌توان در رایانه ذخیره و توسط نرم‌افزارهای مربوطه (گاها بصورت هوشمندانه!) پردازش کرد؛ نتیجه این پردازش می‌تواند به خروجی رایانه تبدیل گردد.

هدف از سازوکار توجه کمکی معرفی شده در مدل پیشنهادی نیز پیروی از عملکرد ذهنی انسان هنگام تولید خلاصه است که متنی در اختیار آن قرار گرفته و خلاصه‌ای از آن به شرطی که مفهوم اصلی متن ورودی حفظ شود، تولید شود. همان‌طور که انسان هنگام تولید خلاصه متن ورودی را خوانده و نکات مهم آن را علامت‌گذاری کرده و سپس به کمک مفاهیم مهم مشخص‌شده متن به تولید خلاصه می‌پردازد، در مدل پیشنهادی نیز سازوکار توجه همین کار را انجام می‌دهد. در این راستا بجای اینکه از کل متن ورودی برای تولید خلاصه استفاده شود، بخش‌های مهم‌تر متن به کمک سازوکار توجه پیشنهادی مشخص شده و پس از آن تنها بخش‌های مهم متن بجای کل متن ورودی کدگذاری شده و در اختیار کدگشا برای تولید خلاصه قرار می‌گیرد.

همچنین، با توجه به اینکه خلاصه‌های چکیده‌ای تولیدشده توسط معماری‌های کدگذار-کدگشا عموماً از لحاظ معنایی منسجم بوده ولی دارای ساختار نحوی مناسبی نمی‌باشند، در مدل پیشنهادی این مقاله از ترکیب بردارهای تعبیه با ویژگی‌های زبانی ازجمله نقش کلمات و موجودیت‌های نامدار به‌عنوان ورودی مدل یادگیری استفاده می‌شود. این مسئله باعث می‌شود که ضمن توجه به ویژگی‌های زبانی، خلاصه تولیدشده نیز از لحاظ نحوی منسجم‌تر باشد. همچنین، با توجه به اینکه کلمات نادر موجود در متن یکی دیگر از مشکلات این شبکه‌های یادگیری عمیق در تولید خلاصه می‌باشند، مدل پیشنهادی از یک سوئیچ به همراه یک حد آستانه در کدگشا برای حل این مشکل استفاده کرده است.

به‌طورکلی، سهم علمی این مقاله را می‌توان به‌صورت زیر بیان کرد:

۱) استفاده از ترکیب بردارهای تعبیه و ویژگی‌های زبانی به‌عنوان ورودی مدل کدگذار-کدگشا به‌منظور تولید خلاصه‌هایی که از لحاظ نحوی منسجم هستند.

۲) استفاده از سازوکار توجه کمکی در بخش کدگذار (برخلاف سایر روش‌های پیشین که از سازوکار توجه در بخش کدگشا استفاده کردند) به‌منظور استخراج اطلاعات مهم متن ورودی برای تولید خلاصه‌هایی که از لحاظ معنایی منسجم بوده و حاوی مطالب اصلی متن ورودی هستند. در واقع، سازوکار توجه کمکی پیشنهادی از عملکرد ذهن انسان برای تولید خلاصه پیروی می‌کند و در آن بجای اینکه مشابه روش‌های پیشین کل متن کدگذاری شده و در اختیار کدگذار برای تولید خلاصه قرار گیرد، ابتدا قسمت‌های مهم‌تر متن شناسایی شده و کدگذاری می‌شوند.

۳) استفاده از یک سوئیچ با حد آستانه در لایه کدگشا به‌منظور توجه به کلمات نادر هنگام تولید خلاصه. در مدل پیشنهادی برخلاف مدل‌های پیشین که در آن‌ها سوئیچ به‌عنوان یک تابع فعال‌ساز سیگموئید به کمک یک لایه خطی و بر اساس کل متن ورودی در هر مرحله زمانی مدل می‌شود ، از احتمال شرطی مورد استفاده در مدل دنباله‌ای برای مشخص کردن کلمات نادر بهره می‌برد. پیرو آن اگر احتمال به‌دست‌آمده از یک حد آستانه‌ای بالاتر بود مدل وارد حالت تولید و در غیر این صورت وارد حالت کپی می‌شود.

۴) مدل پیشنهادی روی دو مجموعه داده CNN/Daily Mail و DUC-۲۰۰۴ مورد آزمایش قرار گرفت و بر اساس نتایج حاصل از آزمایش‌ها و معیار ارزیابی ROUGE، مدل پیشنهادی از دقت بالاتری نسبت به سایر روش‌های پیشین برای تولید خلاصه چکیده‌ای متون برخوردار است.

بخش‌های بعدی مقاله نیز به‌صورت زیر سازماندهی‌شده است: مرور مختصری از مهم‌ترین تحقیقات انجام شده در زمینه خلاصه سازی متون با استفاده از یادگیری عمیق در بخش دوم ارائه‌شده است. در بخش سوم روش پیشنهادی و جزئیات آن به تفضیل بیان شده است. بخش چهارم شامل جزئیات پیاده‌سازی و نتایج حاصل از آزمایش‌ها است. توضیحاتی در راستای نتایج حاصل از پژوهش و کارهایی که در ادامه می‌توان انجام داد، در بخش پنجم ارائه شده است.

۲. پیشینه پژوهش

خلاصه‌سازی خودکار متون از اولین موضوعات تحقیقاتی در حوزه پردازش زبان طبیعی است که نظر محققان زیادی را به خود جلب کرده و پس از دهه‌ها همچنان موردتوجه است. اولین روش‌های خلاصه‌سازی خودکار متن بر ویژگی‌هایی مانند فرکانس کلمات و

عبارات متمرکز بود [۱۳]. پس از آن مسئله خلاصه‌سازی به انواع مختلف انشقاق یافت و برای هر یک، روش‌ها و رویکردهای متعددی پیشنهاد گردید. خلاصه‌سازی تک سندی، خلاصه‌سازی چند سندی، خلاصه‌سازی متمرکز بر پرسمان، خلاصه‌سازی پرونده‌های پزشکی و خلاصه‌سازی ایمیل از جمله انواع مختلف خلاصه‌سازی است که ارائه روش‌ها و الگوریتم‌های متعددی را ایجاب کرده است. با توجه به اینکه تمرکز ما در این مقاله بر روی خلاصه‌سازی چکیده‌ای با رویکرد یادگیری عمیق است، مرور تحقیقات پیشین را به این زمینه محدود کرده و در ادامه این بخش کارهای انجام شده در حوزه خلاصه‌سازی متون با تأکید بر خلاصه‌سازی چکیده‌ای و مدل‌های یادگیری عمیق مورد بررسی قرار می‌گیرند.

انتشار مقاله [۱۴] می‌توان نقطه عطفی در تحقیقات پردازش متن به شمار آورد. در این مقاله یک مدل یادگیری عمیق مبتنی‌بر معماری کدگذار-کدگشا برای ترجمه ماشینی ارائه شد که ورودی آن جمله در زبان مبدأ و خروجی آن جمله در زبان مقصد است. در این مدل کدگذار یک شبکه حافظه کوتاه بلندمدت (LSTM)[۱] چندلایه است که جمله ورودی را به یک بردار با طول ثابت نگاشت می‌کند. رمزگشا نیز خود یک LSTM چندلایه دیگر است که است که این بردار را گرفته و دنباله‌ای از کلمات را به‌عنوان خروجی تولید می‌کند. در ادامه از این مدل کدگذار و کدگشا الهام‌بخش مطالعات حوزه خلاصه‌سازی متون استفاده شد. درواقع با اینکه خلاصه‌سازی متون بسیار شبیه ترجمه ماشینی است، اما در خلاصه‌سازی دنباله خروجی همواره کوچک‌تر از دنباله ورودی می‌باشد. در این راستا، راش و همکارانش [۱۵] یک شبکه کدگذار-کدگشا برای ترجمه ماشینی معرفی کردند که از شبکه LSTM بهره می‌برد. همچنین آن‌ها از سازوکار توجه [۱۶] نیز به‌عنوان راهکاری که هنگام رمزگشایی اطلاعات کد شده ورودی را به‌عنوان یکی از پارامترها در نظر می‌گیرد تا فقدان احتمالی اطلاعات را جبران و کیفیت خروجی را افزایش دهد، استفاده کردند. در ادامه کوپرا و همکارانش [۱۷] مدلی برای خلاصه‌سازی جملات ارائه دادند که از مدل ارائه شده توسط راش و همکارانش [۱۵] مشتق شده بود. در این مدل یک سازوکار توجه کانولوشنی پیشنهاد شد که به‌جای موردتوجه قرار دادن تنها یک ورودی در هر مرحله از کدگشایی، از ترکیب وزن‌دار ورودی‌ها استفاده شد و وزن‌ها نیز خود در فرآیند آموزش شبکه آموزش دیدند.

ژنگ و همکارانش [۱۸] سه مدل GRU، LSTM و LSTM سلسله‌مراتبی را برای خلاصه‌سازی متون مورد بررسی قراردادند. در این مدل همه جملات دو بار خوانده می‌شوند یعنی دو بار به شبکه داده می‌شوند. انگیزه این کار این است که همانند انسان هنگام تولید خلاصه مجدداً به متن اصلی توجه شود. این کار باعث می‌شود که کدگذاری جملات با دقت بالاتری صورت گیرد. شن و همکارانش [۱۹] شبکه‌ای کدگذار-کدگشا برای تولید سرخط یا تیتر معرفی کردند که در آن بجای استفاده از LSTM از GRU استفاده شده است. بخش کدگذار این مدل شبکه‌ای دوطرفه[۲] و بخش کدگشای آن یک‌طرفه[۳] است. تاکاسا [۲۰] نیز یک مدل برای خلاصه‌سازی مبتنی بر سازوکار توجه معرفی کرد تا اطلاعات مرتبط با بازنمایش معنایی چکیده (AMR)[۴] جملات را کدگذاری نماید. یک تجزیه‌گر AMR جملات را تجزیه کرده و سپس یک مدل شبکه با ساختار درختی که مجهز به سازوکار توجه است برای رمزگذاری اطلاعات نحوی و معنایی به کار می‌برد.

ونگ و همکارانش نیز در پژوهشی کارایی یک شبکه کدگذار-کدگشا LSTM را برای ساده‌سازی[۵] متن مورد بررسی قرار دادند [۲۱]. در ساده‌سازی متن اعمالی مانند جایگزینی کلمات دشوار با کلمات ساده، حذف کلمات زائد و ساده‌سازی ساختارهای نحوی پیچیده مورد توجه است. در این تحقیق کارایی شبکه کدگذار-کدگشا در انجام این اعمال با داده‌های مصنوعی مورد آزمایش قرار گرفت.

در ادامه، چن و همکارانش مدل نسبتاً پیچیده‌تری را برای خلاصه‌سازی جملات ارائه دادند [۲۲]. در این مدل کدگذار یک شبکه دوطرفه است و سازوکار توجه بخش کدگشا نیز به وضعیت خروجی تا مرحله قبل توجه دارد. بدین صورت هنگام تولید یک خروجی در هر محله اطلاعات کد شده تا مرحله قبل نیز لحاظ می‌شوند. سانگ و همکارانش [۱۲] مدلی پیشنهاد دادند که از دو بخش تشکیل شده بود. در بخش اول عبارات از سند ورودی استخراج شده و در بخش دوم متن خلاصه با استفاده از روش‌های یادگیری عمیق تولید می‌شود. هدف اصلی مدل معرفی شده توسط آن‌ها تأکید روی عبارات بجای کلمات بوده است.

ژاو و همکارانش [۷] از ترکیب بردارهای بازنمایش کلمات و نقش کلمات در جمله با شبکه‌های یادگیری عمیق برای تولید خلاصه بهره بردند. در مدل معرفی‌شده توسط آن‌ها ترکیب بردارهای بازنمایش کلمات و نقش کلمات به یک شبکه حافظه کوتاه‌مدت بلند یک‌طرفه داده می‌شود. سی و همکارانش [۲۳] نیز مدلی را پیشنهاد کردند که از LSTM دوطرفه به همراه سازوکار توجه در کدگذار و از یک‌LSTM یک‌طرفه به همراه سازوکار توجه در بخش کدگشا بهره می‌برد.مدل پیشنهادی آن‌ها از یک شبکه تولید کننده اشاره‌گر ترکیبی استفاده می‌کند که می‌تواند کلمات را از طریق نوشتن متن از متن منبع کپی کند که منجر به تولید خلاصه‌های دقیق و منسجم می‌شود.

---

[۱] Long Short-Term Memory

[۲] Bi-directional

[۳] Unidirectional

[۴] Abstract Meaning Representation (AMR)

[۵] Text simplification

صباحی و همکارانش [۲۴] معماری رمزگذار-رمزگشای دوطرفه را برای خلاصه‌سازی متون معرفی کردند که در آن رمزگذار و رمزگشای LSTM دوطرفه هستند. رمزگشای جلو با آخرین حالت مخفی کننده رمزگذار عقب شروع می‌شود درحالی‌که رمزگشای عقب با آخرین حالت مخفی رمزگذار جلو، مقداردهی اولیه می‌شود. علاوه بر این ، یک مکانیسم جستجوی پرتو[۱] دوطرفه به‌عنوان یک الگوریتم استنباط تقریبی برای ایجاد خلاصه‌های خروجی از مدل دو جهته ارائه شده است که مدل را قادر می‌سازد تا در مورد گذشته و آینده استدلال کند و در نتیجه خروجی متعادلی ایجاد کند.

یاو و همکارانش [۲۵] نیز مدلی را ارائه دادند که از یک رمزگذار دوگانه شامل دو بخش اولیه و ثانویه بهره می‌برد. رمزگذار اولیه رمزگذاری درشت را به روشی منظم انجام می‌دهد، درحالی‌که رمزگذار ثانویه اهمیت کلمات را مدل می‌کند و کدگذاری ظریف‌تری را بر اساس متن خام ورودی و خلاصه متن خروجی تولیدشده قبلی ایجاد می‌کند. رمزگذاری دو سطح باهم ترکیب‌شده و به‌عنوان ورودی کدگشا مورد استفاده قرار می‌گیرند تا خلاصه متنوع‌تری ایجاد کند که می‌تواند پدیده تکرار را برای تولید دنباله طولانی کاهش دهد.

لازم به ذکر است که از شبکه‌های عصبی کانولوشنی نیز در سال‌های اخیر برای خلاصه‌سازی متون استفاده شده است. در این راستا، آکویلتی و همکارانش [۲۶] از یک معماری عمیق با چندین لایه کانولوشنی و ادغام بهره بردند. ژانگ و همکارانش [۲۷] نیز یک معماری دنباله‌ای با استفاده از شبکه عصبی کانولوشنی معرفی کردند که شبکه کانولوشنی در بخش کدگذار و هم کدگشا بهره می‌برد. معماری پیشنهادی آن‌ها برای افزایش توجه روی قسمت‌های مهم‌تر متن از سازوکار توجه سلسله‌مراتبی نیز بهره می‌برد.

همان‌طور که مشخص است، با اینکه مطالعات انجام شده در این حوزه به دقت قابل توجهی در خلاصه‌سازی دست یافتند و توسعه روش‌های یادگیری عمیق نیز باعث ایجاد انقلابی در این حوزه شده است، اما هنوز فاصله قابل توجهی بین خلاصه تولید شده توسط ماشین و انسان وجود دارد. در نتیجه می‌توان ادعا کرد که در این حوزه کمتر به روش‌های شناختی که به عملکرد ذهن و مغز انسان توجه دارند، پرداخته شده است. در واقع، با اینکه در سال‌های اخیر همکاری دانشمندان حوزه‌های زبان‌شناسی، علوم‌شناختی، فلسفه ذهن و هوش مصنوعی سبب شده تا گام‌های مهمی در حوزه پردازش زبان طبیعی برداشته شود، اما همچنان راه برای رسیدن به وضعیت مطلوب در این حوزه به خصوص در کاربرد خلاصه‌سازی خودکار متون وجود دارد و نیاز است مدل‌هایی معرفی شوند که بتوانند ساختار ذهنی انسان را هنگام تولید خلاصه مدل‌سازی کنند. می‌توان ادعا کرد که تأکید بر ترکیب روش‌های پردازش زبان طبیعی با جنبه‌های شناختی متن می‌تواند تأثیر قابل‌توجهی در جامعه‌ای امروزی که به سمت هوشمند شدن درحرکت است، داشته باشد و نیاز به راهکارهایی است که بتوان با تکیه‌بر مفاهیم زبان‌شناسی شناختی معنا و مسیر فکری کاربران را از متن استخراج و از آن در کاربردهای مختلف پرداز زبان طبیعی به ویژه خلاصه‌سازی چکیده‌ای بهره برد.

۳. متدولوژی پیشنهادی

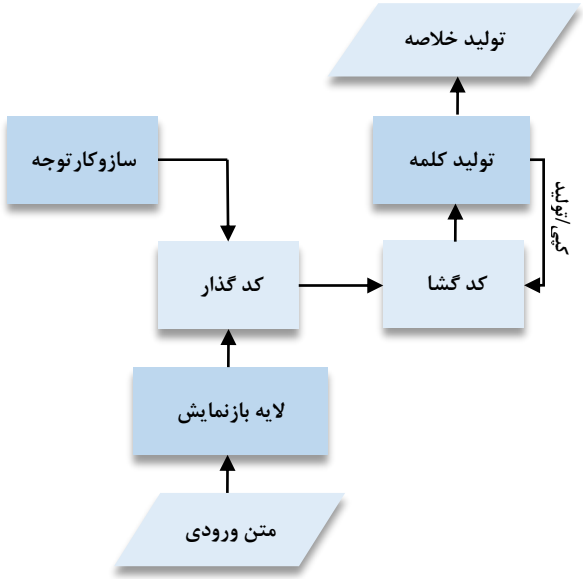
در طول زمان معماری‌های مختلفی برای پیاده‌سازی سیستم‌های خلاصه‌سازی چکیده‌ای متون با استفاده از مفهوم یادگیری عمیق معرفی شدند که از مهم‌ترین آن‌ها می‌توان به شبکه‌های عصبی برگشتی با توجه به دارا بودن خاصیت دنباله‌ای آن‌ها اشاره کرد. در تولید یک خلاصه متنی ورودی یک دنباله‌ای طولانی از کلمات (در قالب متن) و خروجی یک خلاصه از متن ورودی است. درنتیجه می‌توان این مسئله را به‌عنوان مسئله دنباله‌ای چند به چند مدل کرد. در مدل‌های دنباله‌ای دو جزء اصلی به‌عنوان کدگذار-کدگشا وجود دارد. معماری کدگذار-کدگشا عمدتاً برای حل مشکلات دنباله‌ای که در آن دنباله ورودی و خروجی دارای طول متفاوتی می‌باشند، مورد استفاده قرار می‌گیرد. با اینکه معماری کدگذار-کدگشا، معماری بسیار پرکاربردی در حوزه خلاصه‌سازی است اما با محدودیت‌هایی مواجه است.

درواقع، با اینکه خلاصه تولیدشده توسط مدل‌های خلاصه‌سازی چکیده‌ای از لحاظ معنایی منسجم و فراگیر است، اما عملاً دنباله تولیدشده دارای ساختار نحوی مناسبی نیست و ویژگی‌های زبانی مانند نقش کلمات[۲] و موجودیت‌های نامدار[۳] هنگام تولید خلاصه مدنظر قرار نمی‌گیرند [۱۲, ۲۸]. همچنین، مدل‌های کدگذار-کدگشا موجود تمام محتوای متن ورودی برای تولید خلاصه مدنظر قرار می‌دهند و آنجایی که متن ورودی ممکن است شامل بسیاری از اطلاعات نامرتبط با خلاصه باشد، کدگذار نمی‌تواند به‌خوبی معنای متن را کدگذاری کند [۱۱]. نادیده گرفتن تأثیر کلمات نادر در تولید خلاصه از دیگر چالش‌های مدل‌های موجود است. با توجه مدل‌های یادگیری عمیق داده محور می‌باشند، در بسیار از موارد ممکن است کلمات نادر با توجه به اینکه تعداد تکرار کمی دارند، بی‌اهمیت در نظر گرفته شوند درحالی‌که می‌توانند نقش مهمی در تولید خلاصه داشته باشند [۱۲]. در این راستا، در این مقاله یک مدل جدید با الهام از شناخت و ذهن انسان و بر اساس معماری کدگذار-کدگشا معرفی شده است که بتواند بر این چالش‌ها غلبه کند. مدل پیشنهادی دارای چهار بخش بازنمایش ویژگی‌ها، کدگذار، سازوکار توجه و

[۱] Beam search

[۲] Part of Speech (POS)

[۳] Named Entity (NE)



شکل ۱. دیاگرام کلی مدل پیشنهادی

کدگشا است. دیاگرام مدل پیشنهادی در شکل ۱ نشان داده‌شده است و جزئیات آن در ادامه آمده است.

۱،۳ بازنمایش ویژگی‌ها

بازنمایش ویژگی‌ها به فرآیند تبدیل متون به بردار اشاره دارد که برای روش‌های یادگیری ماشین و به‌خصوص یادگیری عمیق با توجه به اینکه این روش‌ها توانایی پردازش داده‌ها به فرم متن و یا رشته‌ای از کلمات را ندارند و به اعداد به‌عنوان ورودی نیاز دارند، از اهمیت بالایی برخوردار است. برای اینکه بتوان از کلمات به‌عنوان ورودی شبکه عصبی استفاده کرد، باید هر کلمه به یک بردار تبدیل شود. در مدل پیشنهادی از مدل اسکیپ‌گرام[۱] برای تولید بردارهای بازنمایی استفاده خواهد شد که در آن هرکدام از کلمات به یک بردار در فضای d-بعدی نگاشت می‌شود. هدفی که در آموزش مدل اسکیپ‌گرام دنبال می‌شود، یافتن بردارهای بازنمایی برای کلمات است. این بردارها باید طوری آموزش ببینند که از روی بردار یک کلمه بتوان کلماتی را که پیرامون آن در یک جمله قرار می‌گیرند تخمین زد [۲۹].

با اینکه استفاده از بازنمایی توزیع‌شده کلمات در یک فضای برداری، باعث می‌شود تا الگوریتم‌های یادگیری در کاربردهای پردازش زبان‌های طبیعی، به کارایی بهتری دست یابند، اما بازنمایش توزیع‌شده از کلمات به‌تنهایی نمی‌تواند مشخص‌کننده مفاهیم کلیدی متن ورودی به‌خصوص در سیستم‌های خلاصه‌سازی باشد. در این راستا، در مدل پیشنهادی بردارهای تعبیه به‌دست‌آمده با ویژگی‌های زبانی ازجمله نقش کلمات[۲] در جمله و موجودیت‌های نامدار[۳] تلفیق خواهند شد و بردار به‌دست‌آمده به‌عنوان ورودی مورد استفاده قرار می‌گیرد. این مسئله باعث می‌شود که ضمن توجه به ویژگی‌های زبانی، خلاصه تولیدشده نیز از لحاظ نحوی منسجم‌تر باشد.

پس از مشخص شدن نقش کلمات و موجودیت‌های نامدار به ازای هر کلمه آن‌ها در یک ماتریس ذخیره و بردار مربوط هرکدام از این ویژگی‌ها را به کمک بردار یک-روشن نشان داده می‌دهد. در ادامه برای تولید بردارهای ورودی مدل پیشنهادی به ازای هر کلمه، بردار بازنشانش مربوط به هرکدام از این ویژگی‌ها (بردار اسکیپ‌گرام، بردار نقش کلمات، بردار موجودیت‌های نامدار) با هم تلفیق‌شده[۴] شده و یک بردار را ایجاد می‌کنند که به‌عنوان ورودی کدگذار مورد استفاده قرار می‌گیرد.

لازم به ذکر است که استفاده از لایه بازنمایش ویژگی‌ها برای تولید بردار کلمات به‌عنوان ورودی شبکه کدگذار-کدگشا باعث می‌شود که تا الگوریتم‌های یادگیری عمیق در کاربردهای پردازش زبان‌های طبیعی، به کارایی بهتری دست یابند. این کارایی بهتر ناشی از گروه‌بندی کلمات با معانی نزدیک به هم است. درواقع، ویژگی که در این بردارها وجود دارد آن است که به کلمات با معانی مشابه بردارهای مشابهی تخصیص می‌یابد که این مسئله می‌تواند در تولید خلاصه معنادار تأثیر قابل‌توجهی داشته باشد.

۲،۳ کدگذار

هدف کدگذار پیروی از فرآیند موجود در ذهن و شناخت انسان هنگام خواندن و فهم یک متن می‌باشد و پیرو آن مسئول نگاشت بردارهای متن ورودی به برداری است که نشان‌دهنده معنای متن است. عموماً از شبکه‌های عصبی برگشتی با توجه به اینکه برای پردازش داده‌ها با خاصیت توالی (متن) مناسب هستند در شبکه‌های کدگذار استفاده می‌شود. با توجه به اینکه شبکه عصبی برگشتی به مرور داده‌های اولیه خود را فراموش کرده و قادر به حفظ وابستگی‌های بلندمدت نیست و همچنین با مشکل محو شوندگی گرادیان مواجه است، از شبکه حافظه طولانی کوتاه-مدت (LSTM) در این لایه استفاده شده است در شبکه LSTM نورون‌های لایه پنهان با بلوک‌های حافظه جایگزین شدند. در این معماری هر لایه پنهان در شبکه برگشتی ساده با چهار لایه جایگزین می‌شود که مسئول حفظ یا فراموشی ورودی و حالت سلول‌های قبل شبکه می‌باشند. مهم‌ترین بخش یک بلوک در LSTM وجود سلول (C_t) است که در واقع خانه حافظه می‌باشد. بر روی حالت سلول هیچ تابع فعال‌سازی اعمال نمی‌شود و تنها اطلاعاتی به آن اضافه یا کم می‌شود و به همین دلیل در هنگام انتشار روبه عقب، دچار گرادیان محو شونده هم نخواهد شد. هر بلوک دارای سه دروازه می‌باشد که عبارت‌اند از دروازه

[۱] Skip-gram
[۲] Part of Speech (POS)
[۳] Named Entity Recognition (NER)
[۴] Concatenate

فراموشی[۱]، دروازه ورودی[۲] و دروازه خروجی[۳]. هر دروازه شامل یک لایه شبکه سیگموئید و یک عملگر ضرب نقطه‌به‌نقطه[۴] می‌باشد که وظیفه کنترل حالت سلول را بر عهده دارد [۱۲]. روابط زیر مقادیر مختلف هر دروازه را در شبکه LSTM محاسبه می‌نماید.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad (۱)$$
$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (۲)$$
$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \quad (۳)$$
$$\tilde{C}_t = tanh(W_C.[h_{t-1}, x_t] + b_C) \quad (۴)$$
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (۵)$$
$$h_t = o_t * \tanh(C_t) \quad (۶)$$

مقدار f_t مشخص می‌کند چه میزان سلول‌های حالت قبلی فراموش شوند و i_t مشخص می‌کند که چه اطلاعات جدیدی در حالت سلول ذخیره شود. o_t تصمیم می‌گیرد که چه چیزی در خروجی دیده شود و مقدار خروجی نهان این بلوک را با استفاده از مقدار به‌دست‌آمده برای حالت سلول محاسبه می‌کند.

لازم به ذکر است به‌منظور به دست آوردن بازنمایی کامل و دقیق‌تر متن مبدأ، از شبکه حافظه طولانی کوتاه-مدت دوطرفه (Bi-LSTM) در مدل پیشنهادی برای رمزگذاری استفاده شده است. LSTM روبه‌جلو بردار کلمات را از چپ به راست خوانده و حالت‌های نهان $(\overrightarrow{h_1}, \overrightarrow{h_2}, \overrightarrow{h_3}, \dots, \overrightarrow{h_n})$ را ایجاد می‌کند. LSTM رو به عقب نیز بردار کلمات را در جهت مخالف خوانده و حالت‌های نهان $(\overleftarrow{h_1}, \overleftarrow{h_2}, \overleftarrow{h_3}, \dots, \overleftarrow{h_n})$ تولید می‌شوند که در آن n طول متن ورودی است (روابط ۷ و ۸).

$$\overrightarrow{h_t} = \text{LSTM}\ (x_t, \overrightarrow{h_{t-1}}) \quad (۷)$$
$$\overleftarrow{h_t} = LSTM(x_t, \overleftarrow{h_{t-1}}) \quad (۸)$$

در نهایت $\overrightarrow{h_t}$ و $\overleftarrow{h_t}$ با هم تلفیق شده و h_t به‌عنوان خروجی نهان به دست می‌آید. پس از اینکه فرآیند کدگذاری به اتمام رسید، حالت مخفی h_t و سلول حالت c_t آخرین واحد زمانی به‌عنوان ورودی کدگشا مورداستفاده قرار می‌گیرند.

۳،۳ سازوکار توجه کمکی

با اینکه معماری مورد استفاده در سیستم‌های خلاصه‌سازی بسیار شبیه به ترجمه ماشینی است، اما با هم تفاوت قابل‌توجهی دارند. در ترجمه ماشینی، کدگشا باید کلیه اطلاعات را از متن منبع دریافت کند. اما در خلاصه‌سازی، کدگشا تنها به اطلاعات مهم موجود در متن برای تولید خلاصه نیاز دارد. بااین‌حال، مدل‌های فعلی خلاصه‌سازی چکیده‌ای خلاصه را بر اساس کل محتوای متن اصلی تولید می‌کنند که باعث می‌شود که کدگذار نتواند متن ورودی را به‌درستی نمایش دهد و پیرو آن اطلاعات ورودی به کدگشا و خلاصه تولیدشده غیردقیق خواهد بود [۳۰]. ایده پشت سازوکار توجه کمکی معرفی‌شده در این مقاله، پیروی از ذهن و شناخت انسان هنگام تولید خلاصه است. همان‌طور که انسان هنگام تولید خلاصه ابتدا کل متن را خوانده و سپس قسمت‌های مهم‌تر آن را برای تولید خلاصه مشخص کرده و تنها به کمک آن‌ها خلاصه را تولید می‌کند، در مدل پیشنهادی نیز به کمک راه‌حل ارائه شده روی قسمت‌های مهم متن تأکید و قسمت‌های بی‌اهمیت را نادیده گرفته می‌شوند. به‌بیان‌دیگر، برخلاف سایر مدل‌های موجود که در آن‌ها سازوکار توجه در بخش کدگشا و هنگام تولید خلاصه برای ایجاد توازن بین متن ورودی و خلاصه اعمال می‌شود، در مدل پیشنهادی سازوکار توجه کمکی معرفی شده در بخش کدگذار اعمال می‌شود تا بتوان به کمک آن بجای کل متن ورودی تنها بخش‌های مهم آن را کدگذاری کرده و در اختیار کدگشا قرار داد. جزئیات سازوکار توجه پیشنهادی در شکل ۲ نشان داده شده است.

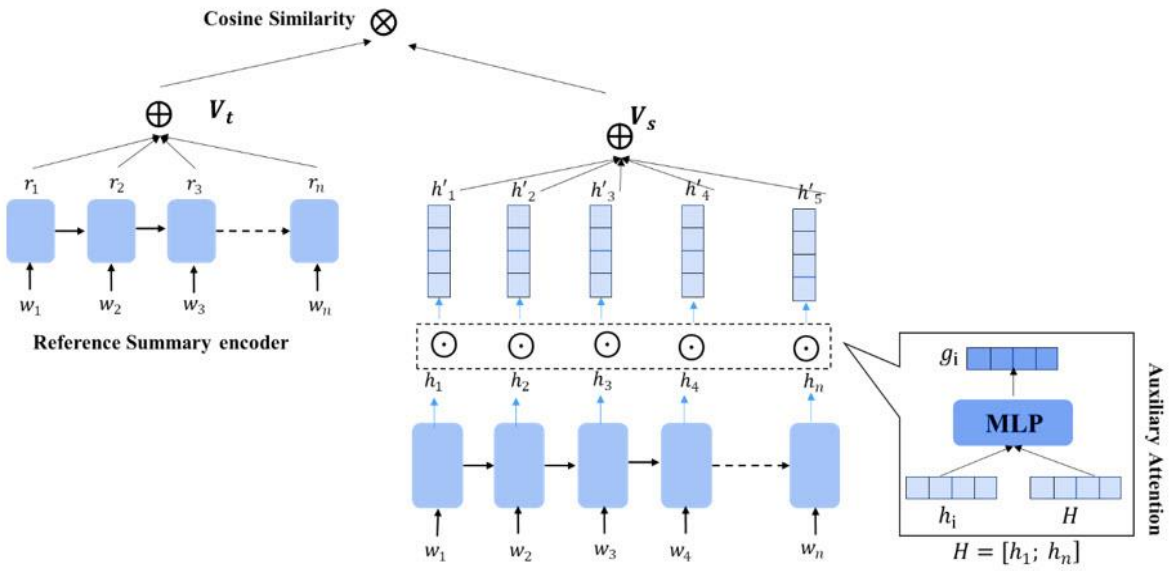
در این راستا، پس از اینکه یک مجموعه از حالت‌های مخفی $(h_1, h_2, h_3, \dots, h_n)$ توسط کدگذار تولید شد، h_1 تا h_n با هم تلفیق می‌شوند تا بتوان به کمک آن متن ورودی را به‌صورت دقیق کدگذاری کرد (رابطه ۹). در اینجا H نشان‌دهنده کل محتوای متن ورودی است.

$$H = [\ h_1;\ h_n] \quad (۹)$$

در ادامه برای استخراج اطلاعات مهم، باید سازوکار توجه کمکی اعمال شود. برای این کار، یک بردار وزن با نام g_i معرفی می‌کنیم که نشان‌دهنده اهمیت کلمه iام در کل متن است. در ادامه H و h_i به یک پرسپترون چندلایه داده می‌شوند تا بتوان به کمک آن g_i را تولید کرد (رابطه ۱۰). پس از آن بین g_i و h_i یک ضرب نقطه‌ای انجام می‌شود تا h_i' تولید شود که نشان‌دهنده اطلاعات استخراج‌شده در واحد زمانی iام است (رابطه ۱۱).

$$g_i = \sigma(W_s.H + V_i.h_i + b) \quad (۱۰)$$
$$h_i' = h_i \odot g_i \quad (۱۱)$$

[۱] Forget gate
[۲] Input gate
[۳] Output gate
[۴] Point-wise multiplication operation



شکل۲. جزئیات سازوکار توجه کمکی پیشنهادی

در اینجا W_s، V_i و b پارامترهای قابل‌آموزش و $\odot$ نشان‌دهنده ضرب نقطه‌ای است. در ادامه و پس از استخراج اطلاعات، هدف تاکید روی اطلاعات مهم متن و نادیده گرفتن اطلاعات غیرضروری است. در این راستا، حالات جدید به‌دست‌آمده $(h_1', h_2', h_3', \dots, h_n')$ در بردار V_s ترکیب می‌شوند. V_s که نشان‌دهنده اطلاعات مهم متن است که به‌عنوان ورودی به کدگشا داده خواهد شد (رابطه ۱۲).

$$V_s = \sum_{i=1}^{n} h_i' \qquad (۱۲)$$

لازم به ذکر است با اینکه هدف استفاده از پرسپترون چندلایه استخراج اطلاعات مهم متن ورودی است، اما نمی‌توان صحت اطلاعات استخراج‌شده را تضمین کرد. ازآنجایی‌که معنای خلاصه مرجع و متن ورودی عموماً سازگار هستند، در این مرحله معنای خلاصه مرجع با متن ورودی مورد مقایسه قرار می‌گیرد تا صحت اطلاعات استخراج‌شده تضمین شود. به همین دلیل از یک LSTM دوطرفه نیز برای کدگذاری خلاصه مرجع استفاده می‌شود و تمامی حالت‌های مخفی به‌دست‌آمده (روابط ۱۴و۱۳و۱۵) با هم ترکیب می‌شوند تا بردار معنای V_s ساخته شود (رابطه ۱۶).

$$\overrightarrow{r_i} = LSTM(w_i, \overrightarrow{r_{i-1}}) \qquad (۱۳)$$

$$\overleftarrow{r_i} = LSTM(w_i, \overleftarrow{r_{i+1}}) \qquad (۱۴)$$

$$r_i = [\overrightarrow{r_i}, \overleftarrow{r_i}] \qquad (۱۵)$$

$$V_t = \sum_{i=1}^{s} r_i \qquad (۱۶)$$

در اینجا s طول خلاصه مرجع و r_i نشان‌دهنده معنای تمامی محتوا قبل از کلمه iام در خلاصه مرجع است.

در ادامه از معیار شباهت کسینوسی برای اندازه‌گیری شباهت معنایی بین خلاصه مرجع و متن ورودی استفاده می‌شود. شباهت معنایی مشخص می‌کند که اطلاعات استخراج‌شده تا چه اندازه به خلاصه مرجع شبیه می‌باشند. هرچه نمره شباهت بزرگ‌تر باشد، اطلاعات استخراج‌شده نیز از ارزش معنایی بالاتری برخوردار می‌باشند (رابطه ۱۷).

$$\cos(V_s, V_t) = \frac{V_s . V_t}{||V_s|| . ||V_t||} \qquad (۱۷)$$

برای افزایش دقت اطلاعات استخراج شده توسط پرسپترون چندلایه، نمره شباهت دوباره به شبکه بازگردانده می‌شود. در فرآیند آموزش نمره شباهت معنایی بین خلاصه مرجع و اطلاعات استخراج‌شده باید بیشینه شود که برای این کار تابع لگاریتم درست نمایی منفی[۱] شباهت باید کمینه شود (رابطه۱۸).

$$\text{Loss}_s = -\log(\cos(V_s, V_t)) \qquad (۱۸)$$

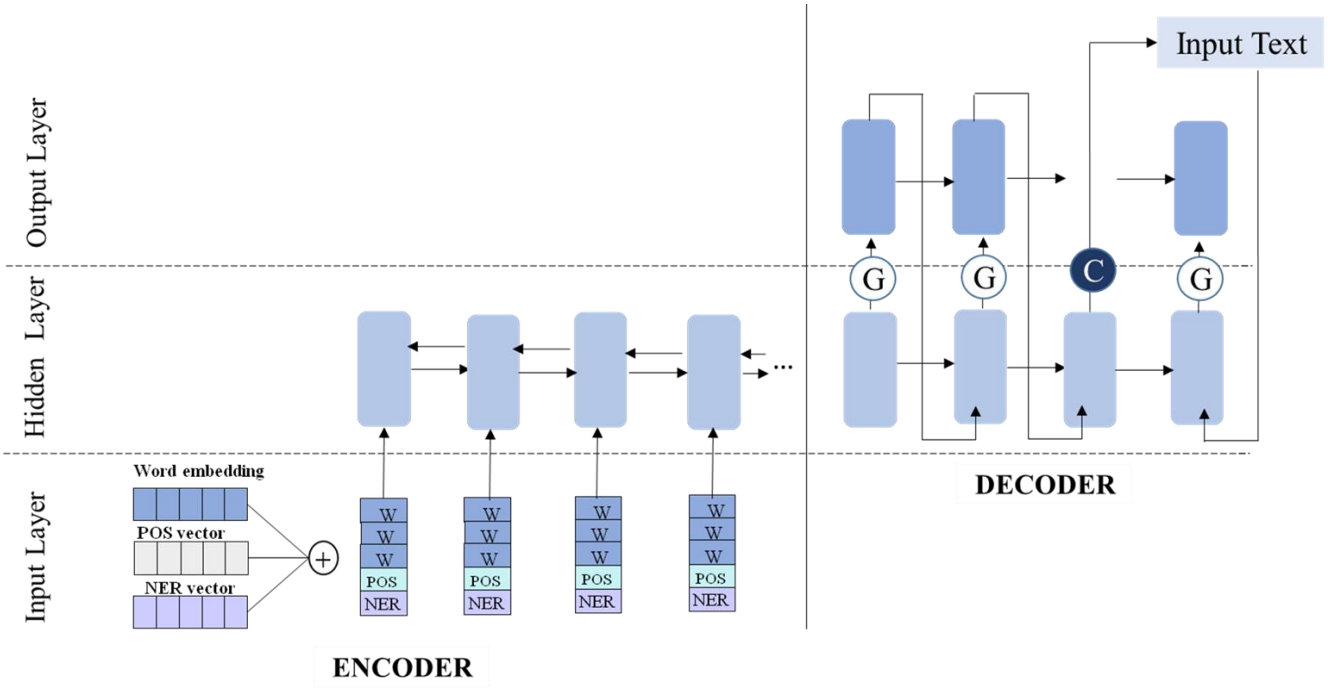
لازم به ذکر است برای اینکه کدگشا هنگام تولید کلمات خلاصه تحت تأثیر خلاصه مرجع قرار نگیرد، نمره شباهت تنها به بخش کدگذار و پرسپترون چندلایه بازگردانده می‌شود و کدگشا بر اساس نمره شباهت آموزش نمی‌بیند.

۴،۳ کدگشا

قبل از اینکه توضیحات مرتبط با لایه کدگشا پرداخته شود، لازم به ذکر است که خلاصه‌سازی خود یک مدل زبانی[۲] است که به متن ورودی مشروط شده است. در نتیجه می‌توان بین چندین توالی داده شده برای یک جمله، آن‌که محتمل‌تر است را انتخاب کرد. در واقع

[۱] Negative log likelihood

[۲] Language model



شکل۳. جزئیات کلی مدل پیشنهادی شامل هر دو بخش کدگذار و کدگشا

یک مدل زبانی برای توالی $x =< x^{(1)}, x^{(2)}, \dots, x^{(n)} >$ به‌صورت رابطه ۱۹ خواهد بود.

$$p(x) = \prod_{t=1}^{n} p(x^{(t)} | x^{(<t)}) \quad (19)$$

در این راستا، خلاصه‌سازی را می‌توان یک مدل زبانی برگشتی در نظر گرفت که مستقیماً احتمال شرطی $p(y|x)$ را در تولید خلاصه از سند مبدأ $x =< x^{(1)}, x^{(2)}, \dots, x^{(n)} >$ به خلاصه مقصد $y =< y^{(1)}, y^{(2)}, \dots, y^{(m)} >$ مدل می‌کند. لازم به ذکر است که طول n (جمله مبدأ) همواره بزرگ‌تر از طول m (جمله مقصد) است و هدف در خلاصه‌سازی به کمک شبکه کدگذار-کدگشا نیز محاسبه این احتمال و سپس استفاده از آن در تولید خلاصه به کمک شبکه یادگیری عمیق است [۳۱].

جزئیات کلی مدل پیشنهادی شامل هر دو بخش کدگذار و کدگشا در شکل ۳ نشان داده شده است. در مدل پیشنهادی پس از کدگذار و سازوکار توجه کمکی، در شبکه کدگشا از یک LSTM یک‌طرفه استفاده شده است که در آن در اینجا s_t و y_t به ترتیب حالت مخفی و ورودی LSTM در واحد زمانی tام می‌باشند (رابطه ۲۰). لازم به ذکر است که هدف از استفاده از شبکه LSTM تخمین احتمال شرطی (رابطه ۲۱) است. شبکه این احتمال شرطی را ابتدا با اقتباس بازنمایی بعد ثابت V_s (نشان‌دهنده کلیه اطلاعات مهم در متن ورودی که از کدگذار به‌دست‌آمده است) برای توالی ورودی $< x^{(1)}, x^{(2)}, \dots, x^{(n)} >$ و در ادامه با محاسبه احتمال $< y^{(1)}, y^{(2)}, \dots, y^{(m)} >$ از رابطه استاندارد مطرح در مدل زبانی (رابطه ۲۱) محاسبه می‌کند. در این رابطه هر توزیع احتمالی $p(y^{(t)} | V_s, y^{(1)}, \dots, y^{(t-1)})$ به کمک یک تابع بیشینه هموار روی همه واژه‌ها داخل واژه‌نامه بازنمایی می‌شود. هر جمله در این مدل با یک علامت خاص مانند <EOS> خاتمه می‌یابد و این امر مدل را قادر می‌سازد تا بتواند توزیع احتمالی را روی هر توالی با طول دلخواهی تعریف کند [۳۱].

$$s_t = LSTM(s_{t-1}, y_t), s_0 = V_s \quad (20)$$

$$p(< y^{(1)}, y^{(2)}, \dots, y^{(m)} > | < x^{(1)}, x^{(2)}, \dots, x^{(n)} >) = \prod_{t=1}^{m} p(y^{(t)} | < V_s, y^{(1)}, y^{(2)}, \dots, y^{(t-1)}) \quad (21)$$

با توجه به اینکه کلمات نادر موجود در متن یکی از مشکلات موجود در تولید خلاصه می‌باشد، کدگشای مدل پیشنهادی با اینکه از قابلیت مدل دنباله‌ای برای تولید کلمات بهره می‌برد، دارای یک سوئیچ به همراه دو حالت تولید و کپی است. لازم به ذکر است که استفاده از سوئیچ به همراه دو حالت تولید و کپی برای حل مشکل کلمات نادر مفهوم جدیدی نیست. اما در مطالعات پیشین سوئیچ عموماً به‌عنوان یک تابع فعال‌ساز سیگموئید به کمک یک لایه خطی و بر اساس کل متن ورودی در هر مرحله زمانی مدل می‌شود. درحالی‌که در مدل پیشنهادی از احتمال شرطی مورد استفاده در مدل دنباله‌ای به‌عنوان سوئیچ و برای مشخص کردن کلمات نادر استفاده می‌شود.

در این راستا، احتمال $\prod_{t=1}^{m} p(y^{(t)} | < V_s, y^{(1)}, y^{(2)}, \dots, y^{(t-1)})$ همانند مدل دنباله‌ای محاسبه می‌شود. اگر این مقدار محاسبه‌شده بیشتر از حد آستانه δ باشد، کدگشا وارد حالت تولید می‌شود. در حالت تولید، کدگشا کلمه بعدی را $y^{(t)}$ با توجه به‌تمامی حالات قبلی و کلماتی که پیش‌ازاین تولید شده‌اند، تولید می‌کند. در غیر این صورت و درحالی‌که مقدار احتمال کمتر از حد آستانه δ باشد،

مدل وارد حالت کپی می‌شود. در حالت کپی فرض بر این است که کلمه تولیدشده توسط کدگشا مناسب نیست. در این حالت مکان کلمه فعلی در متن اصلی پیدا شده و کلمه بعد از آن در متن اصلی در خلاصه قرار می‌گیرد و می‌توان این‌گونه استنباط کرد که کلمه موجود در متن اصلی بهتر از کلمه تولیدشده توسط کدگشا است. لازم به ذکر است که δ یک ابر پارامتر است که مقدار آن در طول آموزش مدل تعیین خواهد شد (رابطه ۲۲).

$$y^* = \prod_{t=1}^{m} p(y^{(t)} | < V_s, y^{(1)}, y^{(2)}, \dots, y^{(t-1)}),$$
$$\begin{Bmatrix} if\ y^* > \delta & \rightarrow Genrate \\ if\ y^* \leq \delta & \rightarrow Copy \end{Bmatrix} \quad (۲۲)$$

۴. نتایج و آزمایش‌ها

۱،۴ مجموعه داده

همان‌طور که پیش از این اشاره شد، کلیه آزمایش‌های این مقاله روی دو مجموعه داده CNN / Daily Mail [۲۸] و DUC-۲۰۰۴ [۳۲] صورت گرفت. مجموعه داده CNN / Daily Mail شامل ۳۱۲،۰۸۵ سند با خلاصه‌هایی شامل چندین جمله است که برای آموزش و ارزیابی مورد استفاده قرار می‌گیرند. هر سند این مجموعه داده شامل حدود ۲۹/۷۴ جمله و به‌طور متوسط ۷۶۶ کلمه است. خلاصه‌های مربوط به هر سند نیز شامل حدود ۳/۷۲ جمله و به‌طور متوسط ۵۳ کلمه است. این مجموعه داده به‌طورکلی شامل ۲۸۶،۸۱۷ نمونه آموزشی، ۱۳،۳۶۸ نمونه اعتبار سنجی و ۱۱،۴۸۷ نمونه تست است.

مجموعه داده DUC-۲۰۰۴ نیز شامل ۵۰۰ سند است و هر سند دارای چهار خلاصه مرجع است که طول هر یک از آن‌ها به‌طور متوسط ۷۵ بایت است. با توجه به اینکه مجموعه داده DUC-۲۰۰۴ یک مجموعه داده بسیار کوچک برای آموزش یک شبکه عصبی عمیق است، این مجموعه داده فقط برای آزمایش مدل پیشنهادی مورد استفاده قرار گرفت. شایان‌ذکر است که اگرچه این یک مجموعه داده قدیمی و کوچک است اما شامل تعداد زیادی خلاصه است که توسط افراد مختلف نوشته شده و به‌عنوان یک مجموعه داده استاندارد نیز در تحقیقات دانشگاهی متعددی مورد استفاده قرار گرفته است. با در نظر داشتن این موضوع که برخی از مطالعات پیشین آزمایش‌های خود را بر روی این مجموعه داده انجام دادند، از این مجموعه داده نیز در آزمایش‌های این مقاله استفاده شد تا بتوان مدل پیشنهادی را با طیف وسیعی ازمدل‌های موجود مورد مقایسه قرار داد.

۲،۴ روش ارزیابی

ارزیابی یکی از مهم‌ترین قسمت‌های مدل‌سازی به شمار می‌آید که به‌وسیله آن می‌توان عملکرد مدل پیشنهادی را با سایر مدل‌های موجود مورد مقایسه قرارداد. برای ارزیابی مدل پیشنهادی این مقاله از معیار ارزیابی ROUGE [۱] استفاده شد. معیار ROUGE با محاسبه تعداد کلمات مشترک بین یک خلاصه با مجموعه‌ای از خلاصه‌های مرجع که توسط انسان تولید شده است، میزان کیفیت خلاصه تولید شده را مورد ارزیابی قرار می‌دهد [۳۳]. به‌بیان‌دیگر این معیار تعداد همپوشانی بین n -گرام‌ها، توالی کلمات و جفت کلمات بین خلاصه تولیدشده توسط کامپیوتر و خلاصه تولیدشده توسط انسان را می‌شمارد و هرچه میزان همپوشانی بین این دو خلاصه بیشتر باشد، کیفیت خلاصه تولیدشده نیز بالاتر خواهد بود . ROUGE از طریق رابطه ۲۳ محاسبه می‌شود.

$$\text{ROUGE} - \text{N} = \frac{\sum_{S \in \{\text{Refrences Summaries}\}} \text{gram}_n \in S \sum \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{S \in \{\text{Refrences Summaries}\}} \text{gram}_n \in S \sum \text{Count}(\text{gram}_n)} \quad (۲۳)$$

در این رابطه n نشان‌دهنده طول n–گرام و $\text{Count}_{\text{match}}(\text{gram}_n)$ نشان‌دهنده بیشینه تعداد رخداد هم‌زمان n–گرام در خلاصه تولیدشده و یک مجموعه از خلاصه‌های مرجع است. لازم به ذکر است که معیار ROUGE-N دارای زیر بخش‌هایی مانند ROUGE-۱ و ROUGE-۲ می‌باشد. ROUGE-۱ تعداد یک-گرام‌های مشترک بین یک خلاصه با یک خلاصه مرجع را ملاک محاسبه قرار می‌دهد و ROUGE-۲ به‌جای یک-گرام‌ها به دو-گرام‌ها توجه دارد.

۳،۴ توصیف آزمایش‌ها

برای نشان دادن کارایی مدل پیشنهادی، لازم است مقایسه‌ای جامع بین مدل پیشنهادی و سایر مدل‌های پیشین ارائه شود. با توجه به اینکه آزمایش‌های این مقاله روی دو مجموعه داده مختلف CNN Daily Mail/ و DUC-۲۰۰۴ انجام شده است، روش‌های موجود که آزمایش‌هایشان را روی این دو مجموعه داده انجام دادند، به‌عنوان خط مبنا برای مقایسه با مدل پیشنهادی مورد بررسی قرار گرفتند. با توجه به مجموعه داده‌های مورد استفاده در آزمایش‌ها، مدل‌های پیشین به دو گروه تقسیم می شوند.

گروه اول: مطالعاتی که آزمایشات خود را روی مجموعه داده CNN Daily Mail / انجام دادند.

- Words-lv۲k-temp-att: این مدلی است که از رمزگذار غنی از ویژگی و مکانیزم اشاره به ترتیب برای بازنشانش کلمه و غلبه بر مشکلات کلمات نادر استفاده می کند [۲۸].

[۱] Recall-Oriented Understudy for Gisting Evaluation.

- Controlled summarization: این مدلی است که کاربران را قادر می‌سازد برخی ویژگی‌های سطح بالا را برای کنترل شکل خلاصه تولید شده تعیین کنند [۳۴].
- Pointer-Generator + Coverage: این مدلی است که نه تنها از مکانیزم اشاره برای غلبه بر مشکل کلمات نادر بهره می‌برد، بلکه برای حل مسئله تکرارپذیری پوشش گسترده واژگان را نیز ارائه می‌دهد [۲۳].
- ML+ intra-attention: این مدلی است که برای غلبه بر مشکل تکرارپذیری از مکانیزم توجه در داخل کدگشا استفاده می‌کند [۳۵].
- ATSDL: این مدلی است که از استخراج عبارت در مرحله پیش‌پردازش برای تهیه ورودی رمزگذار استفاده می کند (این مدل عبارات را به‌عنوان ورودی کدگذار استفاده می‌کند). در این مدل از مکانیسم اشاره برای حل مشکل کلمات نادر نیز استفاده می‌شود [۱۲].
- End۲end w/inconsistency loss: این مدل از ترکیب خلاصه‌سازی چکیده‌ای و خلاصه‌سازی استخراجی برای تولید خلاصه بهره می‌برد [۳۶].
- Attentive information extractor: این مدلی است که از یک استخراج‌کننده اطلاعات دقیق در کدگذار و یک مکانیسم توجه سنتی در کدگشا برای تولید خلاصه استفاده می‌کند [۱۱].
- DCA MLE + SEM + RL: این مدلی است که از عوامل ارتباطی عمیق در کدگذار و کدگشا استفاده کرده و سعی در حل مشکل اسناد طولانی در تولید خلاصه با استفاده از یادگیری تقویتی دارد [۳۷].
- DCA MLE + SEM: این مدلی است که از عوامل ارتباطی عمیق در کدگذار و کدگشا بدون یادگیری تقویتی استفاده می‌کند [۳۷].
- Two-Stage decoder + RL: این مدلی است که یک کدگشای دومرحله‌ای مبتنی بر تبدیل‌کننده‌ها و سازوکار توجه برای تولید خلاصه بهره می‌برد [۳۸].

گروه دوم: مطالعاتی که آزمایش‌های خود را روی مجموعه داده DUC-۲۰۰۴ انجام دادند.

- +ABS: مدلی است که در آن شبکه عصبی کانولوشنی و مدل زبان عصبی به ترتیب به‌عنوان کدگذار و کدگشا مورد استفاده قرار می‌گیرند [۱۵].
- AC-ABS: مدلی است که از یک چارچوب بازیگر-منتقد برای افزایش کارایی خلاصه‌سازی چکیده‌ای استفاده می کند [۳۹].
- Words-lv۵k-۱sent: این مدل مبتنی بر یک کدگذار-کدگشای مجهز به سازوکار توجه است [۲۸].
- SEASS: این مدل که از رمزگذاری انتخابی برای توسعه مدل دنباله‌ای برای خلاصه متن استفاده می‌کند [۴۰].
- C۲R + Atten: مدلی است که در آن شبکه عصبی کانولوشنی و شبکه عصبی برگشتی به ترتیب در کدگذار و کدگشا مورد استفاده قرار گرفته‌اند [۱۷].

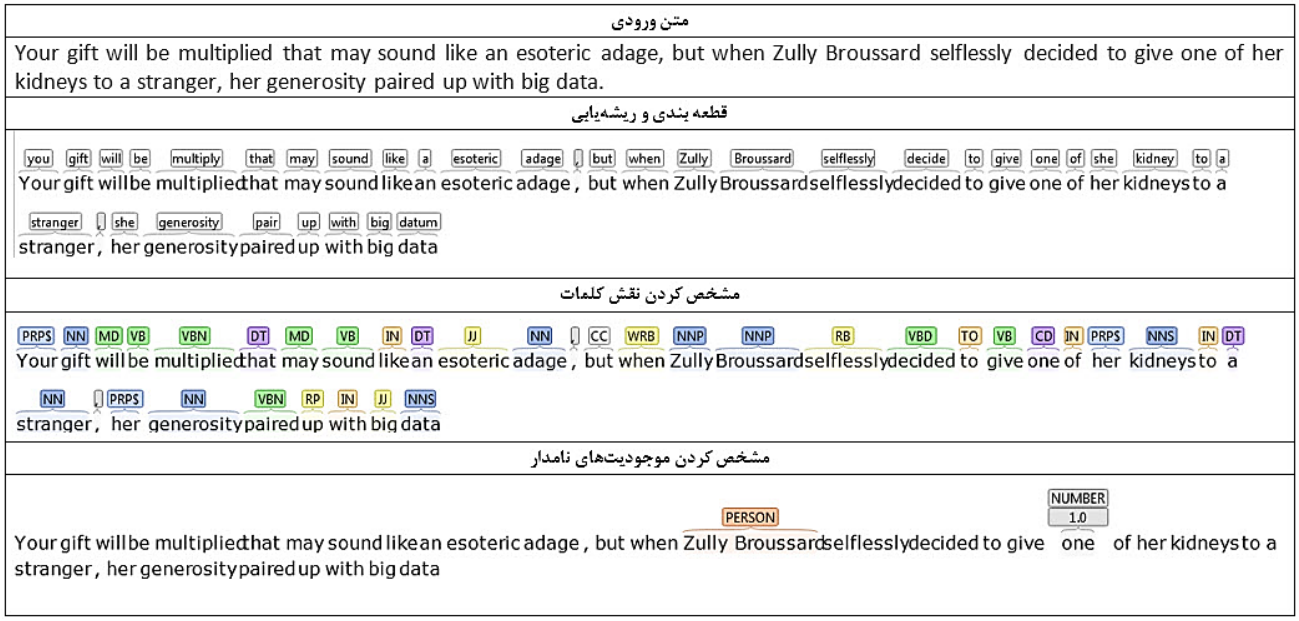
۴،۴ آموزش مدل و ابر پارامترها

پیش‌پردازش به‌عنوان یکی از مراحل برجسته در پردازش زبان طبیعی شناخته می‌شود و ابزارهای بسیاری مانند CoreNLP و NLTK وجود دارند که می‌توان از آن‌ها برای انجام عملیات پیش‌پردازش و کاهش افزونگی کلمات استفاده کرد. در آزمایش‌های این مقاله از CoreNLP با توجه به اینکه دارای مجموعه‌ای از ابزارهای تحلیلی زبان‌شناسی را برای پردازش محتوای متن است، برای انجام پیش‌پردازش استفاده شد. درواقع،همچنین CoreNLP دارای افزونه‌هایی برای تشخیص نقش کلمات (POS)[۱] و تشخیص موجودیت‌های نامدار (NER)[۲] نیز می‌باشد که در این مقاله مورد استفاده قرار گرفتند. در این راستا مجموعه داده ورودی به‌عنوان ورودی به این ابزارها داده شده و خروجی به‌دست آمده نشان‌دهنده نوع موجودیت نامدار و نقش کلمات بود. شکل ۴ نمونه‌ای از متن پیش پردازش‌شده به همراه برچسب موجودیت‌های نامدار و نقش کلمات را نشان می‌دهد.

پس از اعمال عملیات پیش‌پردازش و استخراج کلمات، از مدل اسکیپ‌گرام برای تبدیل کلمه به بردار استفاده شد. در این مرحله ابعاد بردارهای کلمه و اندازه پنجره به ترتیب برابر ۲۰۰ و ۳ در نظر گرفته شد و بردارهای کلمات با نرخ یادگیری ۰/۰۱ به‌روزرسانی شدند. در بخش کدگذار از شبکه حافظه کوتاه‌مدت بلند برای کدگذاری بردارهای مربوط به کلمات استفاده شده و در این بخش اندازه حالت پنهان شبکه برابر ۲۵۶ در نظر گرفته شد. با توجه به این واقعیت که میانگین خلاصه مرجع در مجموعه داده‌های CNN \ Daily Mailو DUC-۲۰۰۴ به ترتیب برابر ۵۳ کلمه و ۷۵ بایت بود، ماکزیمم اندازه واحدهای کدگشا برای این دو مجموعه داده نیز به ترتیب برابر ۱۰۰ و ۲۵ در نظر گرفته شد.

همچنین در آزمایش‌ها اندازه دسته برابر ۱۶ تنظیم و از الگوریتم جستجوی پرتو با اندازه ۴ برای تولید خلاصه استفاده شد. قانون

[۱] https://stanfordnlp.github.io/CoreNLP/pos.html
[۲] https://stanfordnlp.github.io/CoreNLP/ner.html



شکل ۴. نمونه‌ای شماتیک از فرآیند قطعه‌بندی، تشخیص موجودیت‌های نامدار و نقش کلمات توسط ابزار CoreNLP

به‌روزرسانی نیز AdaGrad با نرخ یادگیری ۰/۰۱ بود. بر اساس آزمایش‌های مختلف همچنین مشخص شد که بهترین مقدار برای حد آستانه δ برابر ۰/۵ است. لازم به ذکر است که کلیه پیاده‌سازی‌ها به کمک زبان برنامه‌نویسی پایتون ۳ و کتابخانه تنسورفلو صورت گرفت

۵،۴ نتایج و بحث

همان‌طور که پیش از این بیان شد، آزمایش‌های این مقاله روی دو مجموعه داده مختلف صورت گرفت تا نه‌تنها بتوان کارایی مدل پیشنهادی را با طیف گسترده‌ای از مدل‌های موجود مقایسه کرد بلکه بتوان قابلیت انعطاف‌پذیری و تعمیم‌پذیری مدل پیشنهادی را نیز تأیید کرد زیرا به دست آوردن بالاترین دقت روی دو مجموعه داده با ویژگی‌های مختلف نمی‌تواند تصادفی باشد. نتایج حاصل از آزمایش‌ها روی دو مجموعه داده CNN \ Daily Mail و DUC-۲۰۰۴ بر اساس معیارهای ROUGE در جدول ۱ و ۲ خلاصه شده است.

شایان‌ذکر است که اگرچه استفاده از سازوکار توجه روی شبکه‌های کدگذار-کدگشا یک راه‌حل جدید برای بهبود عملکرد سیستم‌های خلاصه‌سازی نیست، اما مدل‌های خلاصه‌سازی موجود عموماً از سازوکار توجه در کدگشا و برای استخراج کلمات کلیدی هنگام تولید خلاصه بهره می‌برند اما در مدل پیشنهادی از سازوکار توجه کمکی در کدگذار استفاده می‌شود تا بتوان به کمک آن با الهام از ذهن انسان بجای اینکه کل متن ورودی کدگذاری شده و در اختیار کدگشا برای تولید خلاصه قرار گیرد، تنها روی قسمت‌های مهم‌تر متن برای تولید خلاصه تأکید شده و در نتیجه اطلاعات اضافی در اختیار کدگشا قرار نمی‌گیرد.

همچنین در مدل پیشنهادی از ترکیب بردارهای نقش کلمات، موجودیت‌های نامدار و بردارهای به‌دست‌آمده از اسکیپ‌گرام به‌عنوان ورودی مدل کدگذار استفاده می‌شود. بعلاوه، با توجه به اینکه کلمات نادر موجود در متن یکی از مشکلات موجود در تولید خلاصه می‌باشد، کدگشای مدل پیشنهادی با اینکه از قابلیت مدل دنباله‌ای برای تولید کلمات بهره می‌برد، دارای یک سوئیچ به همراه دو حالت تولید و کپی است تا بتواند بر مشکل کلمات نادر نیز غلبه کند.

همچنین در مدل پیشنهادی از ترکیب بردارهای نقش کلمات، موجودیت‌های نامدار و بردارهای به‌دست‌آمده از اسکیپ‌گرام به‌عنوان ورودی مدل کدگذار استفاده می‌شود. بعلاوه، با توجه به اینکه کلمات نادر موجود در متن یکی از مشکلات موجود در تولید خلاصه می‌باشد، کدگشای مدل پیشنهادی با اینکه از قابلیت مدل دنباله‌ای برای تولید کلمات بهره می‌برد، دارای یک سوئیچ به همراه دو حالت تولید و کپی است تا بتواند بر مشکل کلمات نادر نیز غلبه کند.

همان‌طور که مشخص است، مدل پیشنهادی بر اساس معیار ROUGE از دقت بالاتری روی هر دو مجموعه CNN\ Daily Mail و DUC-۲۰۰۴ نسبت به سایر مدل‌های موجود برخوردار است. در واقع، مقادیر ROUGE-۱، ROUGE-۲ و ROUGE-L به ترتیب برابر ۴۲/۳۵، ۲۰/۳۸ و ۳۹/۱۴ روی مجموعه داده CNN\Daily Mail و برابر ۳۴/۱۸، ۱۱/۰۸ و۲۹/۲۱ روی مجموعه داده DUC-۲۰۰۴ هستند که نشان‌دهنده کارایی بالاتر مدل پیشنهادی برای خلاصه‌سازی چکیده‌ای روی هر دو مجموعه داده است.

با توجه به اینکه هدف خلاصه‌سازی چکیده‌ای تولید خلاصه‌هایی دربردارنده مفاهیم اصلی متن ورودی در کنار تولید n-گرام‌های جدیدی است که در خلاصه مرجع وجود دارند، تجزیه‌وتحلیل آماری را برای کشف توانایی انتزاعی مدل پیشنهادی در تولید n-گرام‌های

جدید صورت گرفت که نتایج در شکل ۵ نشان داده شده است. در این راستا، درصد تشابه n -گرام‌های خلاصه به‌دست‌آمده از مدل پیشنهادی را با خلاصه مرجع و درصد تشابه n -گرام‌های خلاصه به‌دست‌آمده توسط مدل Pointer-Generator + Coverage[۲۳] و خلاصه مرجع را روی هر دو مجموعه داده CNN / Daily Mail و DUC۲۰۰۴اندازه‌گیری کردیم. همان‌طور که مشاهده می‌شود، اگرچه تعداد n -گرام‌های جدید تولیدشده توسط مدل پیشنهادی با خلاصه مرجع برابر نیست، اما در مقایسه با مدل Pointer-[۲۳] Generator + Coverage به‌طور چشمگیری بیشتر است که کارآیی مدل پیشنهادی را در تولید خلاصه چکیده‌ای اثبات می‌کند.

۶,۴ آزمایشات مقایسه‌ای

به‌منظور اثبات عملکرد مدل پیشنهادی در این بخش نتایج یکسری از آزمایشات مقایسه‌ای گزارش شده است تا بتوان تأثیر اجزای مختلف مدل پیشنهادی را مشخص کرد. همان‌طور که پیش‌ازاین اشاره شده، مدل پیشنهادی در لایه بازنمایش از ترکیب بردارهای تعبیه، نقش کلمات و موجودیت‌های نامدار بهره می‌برد. در بخش کدگذار نیز در مدل پیشنهادی از یک سازوکار توجه کمکی با الهام از فرآیند ذهنی انسان هنگام تولید خلاصه استفاده می‌شود. در بخش کدگشا نیز مدل پیشنهادی از یک سوئیچ با حد آستانه برای حل مشکل کلمات نادر بهره می‌برد. برای مشخص کردن تأثیر هرکدام از این اجزا روی فرآیند خلاصه‌سازی سه گروه آزمایش مقایسه‌ای صورت گرفت که نتایج آن‌ها در ادامه گزارش شده است.

به‌منظور اثبات تأثیر بردارهای تعبیه، نقش کلمات و موجودیت‌های نامدار روی مدل پیشنهادی، یک‌بار مدل پیشنهادی با بردارهای ورودی که به‌صورت تصادفی مقداردهی شده بودند، یک‌بار با بردارهای تعبیه به‌دست‌آمده از مدل اسکیپ‌گرام و یک‌بار با تلفیق بردارهای تعبیه، نقش کلمات و موجودیت‌های نامدار روی مجموعه داده CNN / Daily Mail آموزش داده شد. نتایج حاصل از آزمایشات

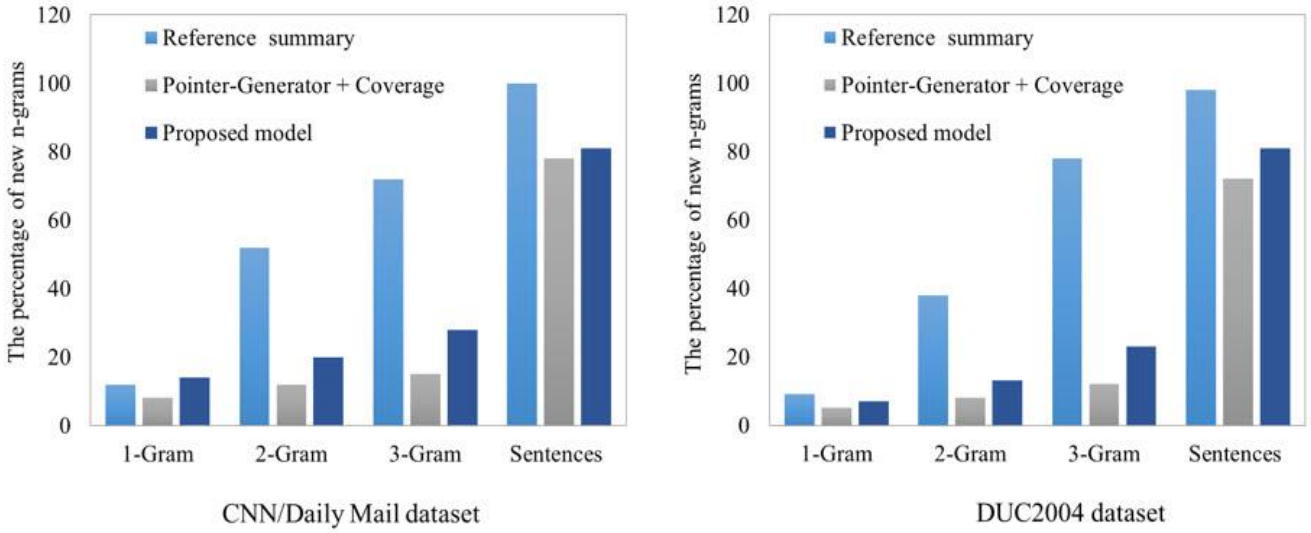
این بخش در شکل ۶ نشان داده شده است. همان‌طور که مشخص است، نه‌تنها استفاده از بردارهای بازنمایش به‌دست‌آمده از مدل اسکیپ‌گرام تأثیر قابل‌توجهی روی دقت خلاصه تولیدشده دارد بلکه تلفیق آن با بردارهای نقش کلمات و موجودیت‌های نامدار دقت خلاصه تولیدشده را افزایش می‌دهد.

جدول ۱. نتایج حاصل از آزمایش‌ها بر اساس معیار ROUGE روی مجموعه داده CNN / Daily Mail

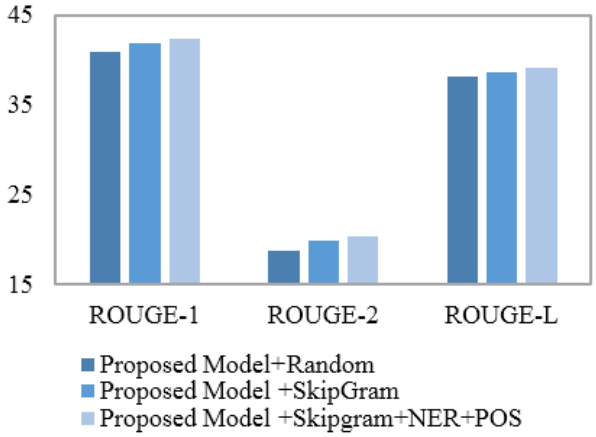
مدل	ROUGE-۱	ROUGE-۲	ROUGE-L
Words-lv۲k-temp-att [۲۸]	۳۵/۴۶	۳۰/۳۰	۳۲/۶۵
Controlled summarization [۳۴]	۳۹/۷۵	۱۷/۲۹	۳۶/۵۴
Pointer-Generator + Coverage [۲۳]	۳۹/۵۳	۱۷/۲۸	۳۶/۳۸
ML+ intra-attention [۳۵]	۳۸/۳۰	۱۴/۸۱	۳۵/۴۹
ATSDL [۱۲]	۳۴/۹۰	۱۷/۸۰	-
End۲end w/inconsistency loss [۳۶]	۴۰/۶۸	۱۷/۹۷	۳۷/۱۳
Attentive information extractor [۱۱]	۴۲/۰۱	۲۰/۰۹	۳۸/۷۸
DCA MLE + SEM + RL [۳۷]	۴۱/۶۹	۱۹/۴۷	۳۷/۹۲
DCA MLE + SEM [۳۷]	۴۱/۱۱	۱۸/۲۱	۳۶/۰۳
Two-Stage decoder + RL	۴۱/۷۱	۱۹/۴۹	۳۸/۷۹
Proposed model (مدل پیشنهادی)	**۴۲/۳۵**	**۲۰/۳۸**	**۳۹/۱۴**

جدول ۲. نتایج حاصل از آزمایش‌ها بر اساس معیار ROUGE روی مجموعه داده DUC-۲۰۰۴

مدل	ROUGE-۱	ROUGE-۲	ROUGE-L
ABS+[۱۵]	۲۸/۱۸	۸/۴۹	۲۳/۸۱
AC-ABS [۳۹]	۳۲/۰۳	۱۰/۹۹	۲۷/۸۶
Words-lv۵k-۱sent[۲۸]	۲۸/۶۱	۹/۴۲	۲۵/۲۴
SEASS [۴۰]	۲۹/۲۱	۹/۵۶	۲۵/۵۱
Attentive information extractor [۱۱]	۳۳/۹۴	۸/۹۹	۲۸/۴۴
C۲R + Atten[۱۷]	۲۸/۹۷	۸/۲۶	۲۴/۰۶
Proposed model (مدل پیشنهادی)	**۳۴/۱۸**	**۱۱/۰۸**	**۲۹/۲۱**

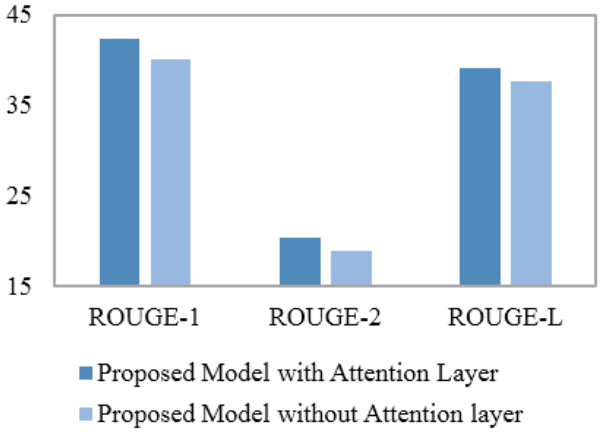


شکل ۵. مقایسه n-گرام‌های تولیدشده توسط مدل پیشنهادی با خلاصه مرجع و مدل Pointer-Generator + Coverage روی دو مجموعه داده CNN \ Daily Mail و DUC-۲۰۰۴

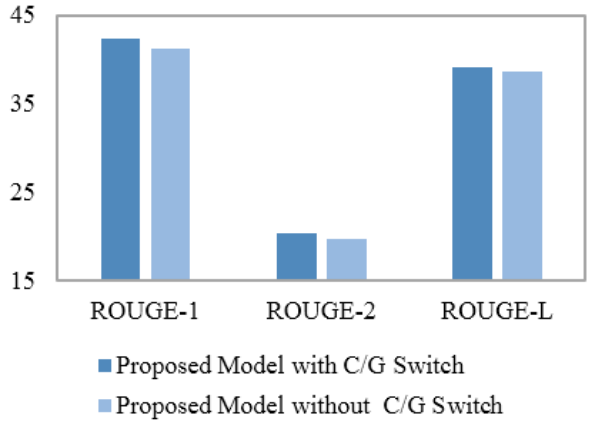


شکل ۶. بررسی تأثیر بازنمایش ویژگی‌های روی دقت مدل پیشنهادی

در بخش دوم آزمایشات این بخش تأثیر سازوکار توجه کمکی پیشنهادی سنجیده شد. در این راستا یک‌بار مدل پیشنهادی به همراه سازوکار توجه و یک‌بار بدون آن درحالی‌که سایر پارامترهای آموزش مدل ثابت نگه‌داشته شده بودند، روی مجموعه داده CNN / Daily Mail آموزش داده شد. نتایج حاصل از آزمایشات این بخش در شکل ۷ نشان داده شده است. همان‌طور که مشخص است، استفاده از سازوکار توجه کمکی پیشنهادی تأثیر قابل‌توجهی روی دقت خلاصه‌های تولیدشده دارد.



شکل ۷. بررسی تأثیر سازوکار توجه کمکی روی دقت مدل پیشنهادی



شکل ۸. بررسی تأثیر سوئیچ با حد آستانه روی دقت مدل پیشنهادی

در بخش سوم آزمایشات، تأثیر سوئیچ با حد آستانه پیشنهادی روی فرآیند تولید خلاصه بررسی شد. در این راستا، یک‌بار مدل پیشنهادی بدون استفاده از سوئیچ درحالی‌که سایر پارامترهای آموزش مدل ثابت نگه داشته شده بودند، روی مجموعه داده CNN / Daily Mail آموزش داده شد. نتایج حاصل از آزمایشات این بخش در شکل ۸ نشان داده‌شده است. همان‌طور که مشخص است، استفاده از سوئیچ دقت مدل پیشنهادی را در تولید خلاصه افزایش می‌دهد.

۵. نتیجه‌گیری و کارهای آینده

امروزه افزایش بیش‌ازپیش حجم اطلاعات متنی موجود در وب باعث پیچیده شدن دسترسی به این داده‌ها شده است. یکی از راه‌حل‌های کاهش این پیچیدگی استفاده از الگوریتم‌های خودکار خلاصه‌سازی متون می‌باشد. خلاصه‌سازی خودکار متن یکی از مهم‌ترین کاربردهای پردازش زبان طبیعی است که هدف آن تولید یک نسخه مختصرتر از سند اصلی توسط یک برنامه رایانه‌ای به نحوی که ویژگی‌ها و نکات اصلی سند اولیه حفظ شود.

با اینکه در طول زمان مطالعات مختلفی پیرامون روش‌های خودکار خلاصه‌سازی متون صورت گرفته است و محققان به نتایج قابل‌توجهی در این حوزه دست یافته‌اند، اما این مفهوم با توجه به کاربرد گسترده‌اش در حوزه‌های مختلف هنوز یکی از جذاب‌ترین موضوعات تحقیقاتی در حوزه پردازش زبان طبیعی به شمار می‌آید و نیاز به پیشرفت در این حوزه همچنان وجود دارد. از طرف دیگر، در سال‌های اخیر روش‌های یادگیری عمیق به پیشرفت قابل‌توجهی در

کاربردهای مختلف پردازش زبان طبیعی به‌خصوص خلاصه‌سازی متون دست یافته‌اند. لازم به ذکر است که با اینکه تحقیقات متعددی در راستای استفاده از روش‌های یادگیری عمیق برای خلاصه‌سازی چکیده‌ای متون صورت گرفته است اما این روش‌ها همچنان با چالش‌های فراوانی در این حوزه مواجه هستند و به‌نوعی در ابتدای مسیر پیشرفت قرار دارند.

با توجه به اینکه عدم توجه به ویژگی‌های زبانی، ناتوانی در استخراج و تأکید روی اطلاعات با اهمیت بالاتر در کدگذار و نادیده گرفتن تأثیر کلمات نادر را می‌توان به‌عنوان چالش‌های روش‌های خلاصه‌سازی چکیده‌ای نام برد، در این مقاله یک مدل مبتنی بر معماری کدگذار-کدگشا برای خلاصه‌سازی چکیده‌ای متون معرفی شده است تا بتواند ضمن غلبه بر چالش‌های موجود به‌دقت بالاتری نسبت به روش‌های پیشین دست یابد. مدل پیشنهادی این مقاله با توجه به این واقعیت که تولید بازنمایش توزیع‌شده از کلمات به‌تنهایی نمی‌تواند مشخص‌کننده مفاهیم کلیدی متن ورودی باشد، از ترکیب بردارهای تعبیه با ویژگی‌های زبانی ازجمله نقش کلمات در جمله و موجودیت‌های نامدار به‌عنوان ورودی مدل کدگذار بهره می‌برد. همچنین، مدل پیشنهادی مجهز به یک سازوکار توجه کمکی در بخش کدگذار می‌باشد. ایده پشت سازوکار توجه کمکی معرفی شده پیروی از ذهن انسان هنگام تولید خلاصه است و هدف آن این است که بجای اینکه کل متن ورودی که ممکن است حاوی بخش زیادی از اطلاعات نامرتبط با خلاصه نهایی باشد کدگذاری شود، قسمت‌های مهم‌تر متن استخراج و کدگذاری شده و در اختیار کدگشا قرار گیرد. لازم به ذکر است که مدل پیشنهادی برای غلبه بر مشکلات کلمات نادر در تولید خلاصه از یک از یک سوئیچ به همراه یک حد آستانه در کدگشا بهره می‌برد. مدل پیشنهادی این مقاله روی دو مجموعه داده CNN / Daily Mail و DUC-۲۰۰۴ مورد آزمایش قرار گرفت و بر اساس نتایج حاصل از ارزیابی‌ها مدل پیشنهادی بر اساس معیار ارزیابی ROUGE از دقت بالاتری روی هر دو مجموعه داده برای خلاصه‌سازی چکیده‌ای متون برخوردار است.

با توجه به اینکه مفاهیم خلاصه‌سازی چکیده‌ای متون و یادگیری عمیق از مهم‌ترین موضوعات تحقیقاتی در حوزه پردازش زبان طبیعی و هوش مصنوعی به‌حساب می‌آیند که نتایج حاصل از آن‌ها می‌تواند تأثیر قابل‌توجهی در دنیای واقعی داشته باشد توسعه و ادامه فرآیند پیشرفت آن‌ها از اهمیت فراوانی برخوردار است. در ادامه این پژوهش نیز می‌توان از سایر شبکه‌های یادگیری ژرف و یا ترکیب آن‌ها در بخش‌های کدگذار و کدگشا بهره برد. استفاده از سایر روش‌های بازنمایش ویژگی‌ها برای تولید ورودی مناسب روش پیشنهادی نیز از کارهایی است که می‌توان در ادامه انجام داد. هم‌چنین می‌توان کاربرد روش پیشنهادی را در سایر حوزه‌ها و زبان‌های دیگر، به‌ویژه زبان فارسی سنجید.

مراجع

[۱] M. Dey and D. Das, "A Deep Dive into Supervised Extractive and Abstractive Summarization from Text," in *Data Visualization and Knowledge Engineering*: Springer, ۲۰۲۰, pp. ۱۰۹-۱۳۲.

[۲] T. Shi, Y. Keneshloo, N. Ramakrishnan, and C. K. Reddy, "Neural abstractive text summarization with sequence-to-sequence models," *ACM Transactions on Data Science*, vol. ۲, no. ۱, pp. ۱-۳۷, ۲۰۲۱.

[۳] A. M. Al-Numai and A. M. Azmi, "The Development of Single-Document Abstractive Text Summarizer During the Last Decade," in *Trends and Applications of Text Summarization Techniques*: IGI Global, ۲۰۲۰, pp. ۳۲-٦۰.

[٤] S. Chakraborty, X. Li, and S. Chakraborty, "A more abstractive summarization model," *arXiv preprint arXiv:*۲۰۰۲,۱۰۹۵۹, ۲۰۲۰.

[٥] L. Abualigah, M. Q. Bashabsheh, H. Alabool, and M. Shehab, "Text Summarization: A Brief Review," in *Recent Advances in NLP: The Case of Arabic Language*: Springer, ۲۰۲۰, pp. ۱-۱٥.

[٦] Y. Dong, "A survey on neural network-based summarization methods," *arXiv preprint arXiv:*۱۸۰٤,۰٤٥۸۹, ۲۰۱۸.

[۷] F. Zhao, B. Quan, J. Yang, J. Chen, Y. Zhang, and X. Wang, "Document Summarization using Word and Part-of-speech based on Attention Mechanism," in *Journal of Physics: Conference Series*, ۲۰۱۹, vol. ۱۱٦۸, no. ۳: IOP Publishing, p. ۰۳۲۰۰۸ .

[۸] D. Suleiman and A. Awajan, "Deep Learning Based Abstractive Text Summarization: Approaches, Datasets, Evaluation Measures, and Challenges," *Mathematical Problems in Engineering*, vol. ۲۰۲۰, ۲۰۲۰.

[۹] H. Lin and V. Ng, "Abstractive Summarization: A Survey of the State of the Art," in *Proceedings of the AAAI Conference on Artificial Intelligence*, ۲۰۱۹, vol. ۳۳, pp. ۹۸۱۵-۹۸۲۲ .

[۱۰] W. Kryściński, N. S. Keskar, B. McCann, C. Xiong, and R. Socher, "Neural text summarization: A critical evaluation," *arXiv preprint arXiv: ۱۹۰۸.۰۸۹۶۰*, ۲۰۱۹.

[۱۱] X. Xiang, G. Xu, X. Fu, Y. Wei, L. Jin, and L. Wang, "Skeleton to Abstraction: An Attentive Information Extraction Schema for Enhancing the Saliency of Text Summarization," *Information*, vol. ۹, no. ۹, p. ۲۱۷, ۲۰۱۸.

[۱۲] S. Song, H .Huang, and T. Ruan, "Abstractive text summarization using LSTM-CNN based deep learning," *Multimedia Tools and Applications*, vol. ۷۸, no. ۱, pp. ۸۵۷-۸۷۵, ۲۰۱۹.

[۱۳] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of research and development*, vol. ۲, no. ۲, pp. ۱۵۹-۱۶۵, ۱۹۵۸.

[۱۴] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, ۲۰۱۴, pp. ۳۱۰۴-۳۱۱۲ .

[۱۵] A. M. Rush, S. Chopra, and J .Weston, "A neural attention model for abstractive sentence summarization," *arXiv preprint arXiv:*۱۵۰۹.۰۰۶۸۵, ۲۰۱۵.

[۱۶] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv: ۱۴۰۹.۰۴۷۳،۲۰۱۴.*

[۱۷] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in *Proceedings of the ۲۰۱۶ Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, ۲۰۱۶, pp. ۹۳-۹۸ .

[۱۸] W. Zeng, W. Luo, S. Fidler, and R. Urtasun, "Efficient summarization with read-again and copy mechanism," *arXiv preprint arXiv:*۱۶۱۱.۰۳۳۸۲, ۲۰۱۶.

[۱۹] S. Shen, Y. Zhao, Z. Liu, and M. Sun, "Neural headline generation with sentence-wise optimization," *arXiv preprint arXiv:*۱۶۰۴.۰۱۹۰۴, ۲۰۱۶.

[۲۰] S. Takase, J. Suzuki, N. Okazaki, T. Hirao, and M. Nagata, "Neural headline generation on abstract meaning representation," in *Proceedings of the ۲۰۱۶ conference on empirical methods in natural language processing*, ۲۰۱۶, pp. ۱۰۵۴-۱۰۵۹ .

[۲۱] T. Wang, P. Chen, K. Amaral, and J. Qiang, "An experimental study of LSTM encoder-decoder model for text simplification," *arXiv preprint arXiv:*۱۶۰۹.۰۳۶۶۳, ۲۰۱۶.

[۲۲] Q. Chen, X. Zhu, Z. Ling, S. Wei, and H. Jiang, "Distraction-based neural networks for document summarization," *arXiv preprint arXiv:*۱۶۱۰.۰۸۴۶۲, ۲۰۱۶.

[۲۳] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," *arXiv preprint arXiv:*۱۷۰۴.۰۴۳۶۸, ۲۰۱۷.

[۲۴] K. Al-Sabahi, Z. Zuping, and Y. Kang, "Bidirectional attentional encoder-decoder model and bidirectional beam search for abstractive summarization," *arXiv preprint arXiv:*۱۸۰۹.۰۶۶۶۲, ۲۰۱۸.

[۲۵] K. Yao, L. Zhang, D. Du ,T. Luo, L. Tao, and Y. Wu, "Dual encoding for abstractive text summarization," *IEEE transactions on cybernetics*, ۲۰۱۸.

[۲۶] W. H. Alquliti and N. B. A. Ghani, "Convolutional Neural Network based for Automatic Text Summarization".

[۲۷] Y. Zhang, D. Li, Y .Wang, Y. Fang, and W. Xiao, "Abstract Text Summarization with a Convolutional Seq٢seq Model," *Applied Sciences*, vol. ۹, no. ۸, p. ۱۶۶۵, ۲۰۱۹.

[۲۸] R. Nallapati, B. Zhou, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence rnns and beyond," *arXiv preprint arXiv:*۱۶۰۲.۰۶۰۲۳, ۲۰۱۶.

[۲۹] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality, Nips," ۲۰۱۳.

[۳۰] W. Yoon, Y. S. Yeo, M. Jeong, B.-J. Yi, and J. Kang, "Learning by Semantic Similarity Makes Abstractive Summarization Better," *arXiv preprint arXiv:*۲۰۰۲.۰۷۷۶۷, ۲۰۲۰.

[۳۱] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:*۱۳۰۸.۰۸۵۰, ۲۰۱۳.

[۳۲] P. Over, H. Dang, and D .Harman, "DUC in context," *Information Processing & Management*, vol. ۴۳, no. ۶, pp. ۱۵۰۶-۱۵۲۰, ۲۰۰۷.

[۳۳] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in *Association for Computational Linguistic*, Barcelona, Spain, ۲۰۰٤ .

[۳٤] A. Fan, D. Grangier, and M. Auli, "Controllable abstractive summarization," *arXiv preprint arXiv:*۱۷۱۱,۰۵۲۱۷, ۲۰۱۷.

[۳۵] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," *arXiv preprint arXiv:*۱۷۰۵,۰٤۳۰٤, ۲۰۱۷.

[۳٦] W-.T. Hsu, C.-K. Lin, M.-Y. Lee, K. Min, J. Tang, and M. Sun, "A unified model for extractive and abstractive summarization using inconsistency loss," *arXiv preprint arXiv:*۱۸۰۵,۰٦۲٦٦, ۲۰۱۸.

[۳۷] A. Celikyilmaz, A. Bosselut, X. He, and Y. Choi, "Deep communicating agents for abstractive summarization," *arXiv preprint arXiv:*۱۸۰۳,۱۰۳۵۷, ۲۰۱۸.

[۳۸] H. Zhang, J. Xu, and J. Wang, "Pretraining-based natural language generation for text summarization," *arXiv preprint arXiv:*۱۹۰۲,۰۹۲٤۳, ۲۰۱۹.

[۳۹] P. Li, L. Bing, and W. Lam, "Actor-critic based training framework for abstractive summarization," *arXiv preprint arXiv:*۱۸۰۳,۱۱۰۷۰, ۲۰۱۸.

[٤۰] Q. Zhou, N. Yang, F. Wei, and M. Zhou, "Selective encoding for abstractive sentence summarization," *arXiv preprint arXiv:*۱۷۰٤,۰۷۰۷۳, ۲۰۱۷.