Journal of Applied And Basic Machine Intelligence Research (ABMIR)

پژوهش‌های نظری و کاربردی هوش ماشینی

سال اول، شماره دوم، پاییز و زمستان ۱۴۰۲، صفحات ۱۵۹-۱۴۵

خلاصه‌سازی استخراجی متن با استفاده از مجموعه الگوریتم‌های خلاصه‌سازی و روش Sa-TRB

ابوالفضل صدرالساداتی[۱]*، محمدرضا فیضی درخشی[۲]

[۱]دانشجوی دکتری مهندسی کامپیوتر فنی و مهندسی پردیس ارس دانشگاه تبریز، تبریز، ایران

[۲]استاد گروه مهندسی کامپیوتر ، دانشکده مهندسی برق و کامپیوتر دانشگاه تبریز، تبریز، ایران

مقاله پژوهشی

تاریخ دریافت:

۱۴۰۲/۰۶/۲۸

تاریخ پذیرش:

۱۴۰۲/۰۹/۱۳

کلیدواژه‌ها:

خلاصه‌سازی استخراجی متن، پردازش زبان طبیعی، Luhn، TF-IDF، TextRank، LSA، BERT، نرخ فشردگی، ROUGE

نویسنده‌مسئول:

Abolfazl.Sadrolsadati@Gmail.com

چکیده

خلاصه‌سازی استخراجی متن یک تکنیک ضروری در پردازش زبان طبیعی است که با استخراج مهم‌ترین جملات به تولید نسخه‌های فشرده از متن کمک می‌کند. ازآنجاکه کوتاه کردن و خلاصه‌سازی یک سند متنی، کاری زمان‌بر و طاقت فرساست، یک سیستم خودکار برای ایجاد نسخه‌های کوتاه از متن ضروری به نظر می‌رسد. در خلاصه‌سازی استخراجی جملاتی که حاوی اطلاعات مفید و مرتبط هستند برای خلاصه نهایی انتخاب می‌شوند. به‌منظور شناسایی این جملات الگوریتم‌های متفاوتی وجود دارند که عملکرد و خلاصه ایجادشده از هرکدام بر اساس نوع متن و اندازه خلاصه موردنیاز متفاوت است. در این مقاله روشی بانام **Sa-TRB** ارائه‌شده است، که برگرفته از دو الگوریتم **TextRank** و **BERT** بوده و علاوه بر استفاده از این دو روش از اشتراک جملات ایجادشده سایر الگوریتم‌ها نیز بهره می‌برد تا دقت بالایی در انتخاب جملات خلاصه نهایی داشته باشد. مهم‌ترین معیار برای ارزیابی عملکرد الگوریتم‌ها کیفیت خلاصه نهایی آن‌هاست، چنانکه هرچقدر خلاصه نهایی ایجادشده توسط این الگوریتم‌ها به خلاصه ایجادشده توسط انسان مشابه باشد، کیفیت خلاصه ایجادشده بهتر است. برای به دست آوردن اندازه این تشابه از معیارهای روش **ROUGE** استفاده می‌شود. درنهایت با انجام آزمایش‌هایی روی دیتاست **cnn-dailymail** با اندازه خلاصه‌های مختلف نشان داده می‌شود که روش پیشنهادی با افزایش اندازه خلاصه موردنیاز باوجود کاهش در معیار فراخوانی دارای دقت، امتیاز و درنتیجه کیفیت بالاتر خلاصه نهایی است، به‌طوری‌که در دو آزمایش آخر که نرخ فشردگی ۲۰ و ۲۵ درصد است، امتیاز روش پیشنهادی به ۲۴،۶۸ و ۲۳،۳۴ درصد رسیده است که تقریباً یک درصد از بهترین روش‌های آزمایش‌شده دیگر بهتر است.

: 10.22034/ABMIR.2023.20650.1035

E-ISSN: 2821-2037

۱- مقدمه

در دنیای اطلاعات محور امروزی، اینترنت حجم عظیمی از محتوا را در قالب متن، تصویر، صدا و ویدئو در اختیار قرار می‌دهد. با چنین حجم وسیعی از اطلاعات موجود، خواندن کامل متن برای به دست آوردن اطلاعات مهم به‌طور فزاینده‌ای دشوار و وقت‌گیر شده است. اینجاست که خلاصه‌سازی متن می‌تواند مفید باشد، به‌ویژه زمانی که باید اسناد طولانی و پیچیده را مرور کنیم. خلاصه‌سازی متن روشی برای استخراج اطلاعات ارزشمند از یک متن داده‌شده و ارائه آن به کاربر به شکلی ساده، کوتاه و فشرده است که محتوای مربوط به سند متن مبدأ را برای تشکیل خلاصه حفظ می‌کند [۱،۲]. خلاصه‌سازی خودکار متن (ATS)[1] فرآیندی است که شامل استفاده از الگوریتم‌هایی برای شناسایی و استخراج مهم‌ترین اطلاعات از یک سند متنی و ارائه آن به شکل کوتاه‌تر و فشرده‌تر است. این فرآیند به‌طور گسترده در برنامه‌های کاربردی مختلف، ازجمله مقالات خبری، اسناد حقوقی، مقالات علمی، و گزارش‌های تجاری و بسیاری دیگر استفاده می‌شود. ارتباط زیاد، افزونگی کمتر، نسبت فشرده‌سازی مناسب و پوشش بالا از عوامل ضروری یک خلاصه متن خوب هستند.

دو طبقه‌بندی کلی خلاصه‌سازی متن عبارت‌اند از خلاصه‌سازی انتزاعی و خلاصه‌سازی استخراجی [۳-۵]. در شکل (۱) این دو نوع خلاصه‌سازی مشخص است.

شکل (۱): خلاصه‌سازی استخراجی و انتزاعی

۱-۱ خلاصه انتزاعی

در این تکنیک خلاصه‌سازی، خلاصه تولیدشده شامل جملات بدیعی است که از سند منبع استخراج‌نشده‌اند. این جملات ایده اصلی سند مبدأ را می‌رسانند و با بازنویسی متن از آن شکل گرفته‌اند تا خلاصه‌هایی را ایجاد کنند که بیشتر به‌نظر انسان نوشته‌شده‌باشند [۶]. تولید این نوع خلاصه‌ها پیچیده، وقت‌گیر و چالش‌برانگیز است و نیاز به تجزیه‌وتحلیل گسترده دارند و بنابراین کمتر مورداستفاده قرار می‌گیرند. اگرچه این نوع خلاصه‌ها تمایل به بهبود انسجام و خوانایی دارند [۷]، اما خلاصه‌های تولیدشده با مشکلات متعددی مانند پوچ بودن، نادرستی و تکراری بودن مواجه هستند [۷،۸].

۱-۲ خلاصه‌سازی استخراجی

خلاصه‌سازی استخراجی متن تکنیکی است که کمک می‌کند مهم‌ترین اطلاعات از یک سند طولانی استخراج شود و نسخه کوتاه‌تر و فشرده‌تری از محتوای آن سند ایجاد شود. در خلاصه‌سازی استخراجی، همان‌طور که از نام آن پیداست، مدل جملات برجسته را از سند مبدأ استخراج می‌کند و آن‌ها را برای تشکیل خلاصه استخراجی ترکیب می‌کند. برای انتخاب جملات برجسته، ابتدا به تمام جملات سند مبدأ وزن داده می‌شود و به دنبال آن، جملات با رتبه‌بندی بالابر اساس وزن آن‌ها از سند منبع استخراج می‌شود. سپس این جملات استخراج‌شده برای تولید خلاصه متن ترکیب می‌شوند [۹].

خلاصه‌های تولیدشده از طریق این رویکرد، عموماً انسجام پایینی دارند، بااین‌حال درزمینه خلاصه‌نویسی متن به دلیل پیچیدگی کمتر زمانی و سهولت بیشتر در مقایسه با خلاصه‌های تولیدشده توسط رویکرد خلاصه‌سازی انتزاعی، بسیار رایج هستند. معیار خلاصه‌سازی خوب متن به‌صورت استخراجی این است که خلاصه تولیدشده باید تنوع موضوعی مناسب با افزونگی کم داشته باشد [۱۰]، و دستیابی به هردوی این معیارها به‌صورت موازی چالش‌برانگیز است.

یکی از دلایل اصلی توسعه خلاصه‌سازی استخراجی متن، نیاز روزافزون به دسترسی سریع به اطلاعات کلیدی است. با حجم اطلاعاتی که در دسترس است، خلاصه‌کردن راه‌حلی ارزشمند برای صرفه‌جویی در زمان و تلاش است. خلاصه‌سازی استخراجی متن یک تکنیک محبوب است که خلاصه‌ای مختصر و آموزنده از یک سند طولانی ارائه می‌کند و جایگزینی عالی برای خواندن کل اسناد ارائه می‌دهد.

[1] Automatic Text Summarization

کاربردهای متعددی از خلاصه‌سازی استخراجی متن وجود دارد، ازجمله خلاصه‌سازی اخبار، خلاصه‌سازی اسناد و مدیریت محتوا برای رسانه‌های اجتماعی. خلاصه‌سازی استخراجی متن همچنین می‌تواند در سیستم‌های پاسخگویی خودکار آنلاین، موتورهای جستجو و سیستم‌های پیشنهاد محتوا استفاده شود.

اولین گام در خلاصه‌سازی استخراجی، شناسایی جملات یا عبارات کلیدی در متن اصلی است. این گام را می‌توان با استفاده از تکنیک‌های مختلف پردازش زبان طبیعی (NLP)[1] مانند برچسب گذاری بخشی از گفتار، شناسایی موجودیت نام‌گذاری شده و تجزیه نحوی به دست آورد. این تکنیک‌ها به شناسایی معنادارترین و مرتبط‌ترین محتوا از سند اصلی کمک می‌کند.

هنگامی‌که جملات یا عبارات کلیدی شناسایی شدند، بر اساس ارتباط آن‌ها با ایده اصلی متن رتبه‌بندی می‌شوند. در بسیاری از موارد، رتبه‌بندی با استفاده از روش‌های آماری، مانند فرکانس واژه در معکوس فرکانس سند (TF-IDF)[2] یا تحلیل معنایی پنهان (LSA)[3] انجام می‌شود. این روش‌ها به شناسایی جملات و عباراتی کمک می‌کند که بیشتر نمایانگر موضوع اصلی متن هستند.

درنهایت، جملات یا عبارات انتخاب‌شده برای ایجاد خلاصه‌ای ترکیب می‌شوند که به‌طور دقیق ایده‌های اصلی متن مبدأ را نشان می‌دهد. این کار را می‌توان با استفاده از الگوریتم‌های مختلف برای استخراج جمله یا مدل‌های مبتنی بر گراف به دست آورد. این الگوریتم‌ها کمک می‌کنند تا اطمینان حاصل شود که خلاصه ایجادشده مرتبط، مهم و قابل‌فهم است.

خلاصه‌سازی استخراجی ابزاری قدرتمند برای خلاصه‌سازی خودکار اسناد متنی است. با شناسایی مهم‌ترین جملات یا عبارات و ارائه آن‌ها به شکل کوتاه‌تر و فشرده‌تر، خلاصه‌سازی استخراجی به صرفه‌جویی در زمان و بهبود کارایی پردازش و تجزیه‌وتحلیل داده‌ها کمک می‌کند. در این مقاله، با تمرکز بر خلاصه‌سازی استخراجی، به بررسی روش‌هایی پرداخته‌خواهد شد که خلاصه به‌دست‌آمده توسط آن‌ها شامل انتخاب مهم‌ترین جملات یا عبارات از سند اصلی و ترکیب آن‌ها برای ایجاد یک خلاصه است.

۲- ادبیات تحقیق

خلاصه‌سازی کامپیوتری از اواخر دهه ۱۹۵۰ موردتوجه قرارگرفته است. ایجاد خلاصه متن که شامل مجموعه‌ای از اطلاعات باشد، یک تکلیف ناخوشایند، زمان‌بر برای انسان است، اما برای ماشین‌ها یک کار چالش‌برانگیز است. ماشین باید صرف‌نظر از زمینه اطلاعات‌پایه، قادر به پردازش زبان طبیعی و ایجاد خلاصه‌ای منطقی برای انسان باشد [۱۱].

پردازش زبان طبیعی یک زمینه نوظهور برای تحقیق است که یکی از مهم‌ترین بخش‌ها و کاربردهای آن خلاصه‌سازی متن است. رویکردهای مختلفی برای رسیدگی به مشکلات مرتبط باکار خلاصه‌سازی طراحی‌شده است. اولین رویکردها بر اساس ویژگی محتوا بود که توسط Luhn H.P معرفی شد. بخش عمده‌ای از کار گذشته درزمینه خلاصه‌سازی استخراجی بوده است که شامل تشخیص جملات یا ورودی‌های کلیدی در سند مبدأ و بر اساس تکرار آن‌ها، به‌عنوان خلاصه است. نتایج این تحقیقات، امکان انتخاب جملاتی را که بیانگر موضوع کلی متن است، در قالب خلاصه‌های مرسوم نشان داد. بسیاری از مطالعات از اواسط قرن گذشته به چالش‌های خلاصه‌سازی خودکار متن پرداخته‌اند [۱۲،۱۳].

این مطالعات را می‌توان با توجه به هدف خلاصه‌سازی، ورودی و خروجی سیستم‌های خلاصه‌سازی و تکنیک‌هایی که بر اساس آن سند منبع را مدل‌سازی می‌کنند، به دسته‌های مختلفی تقسیم کرد:

اول، با توجه به هدف اولیه بیان شده از خلاصه‌ها، رویکردهای خلاصه‌سازی را می‌توان به روش‌های عمومی که شامل تولید خلاصه‌ای است که حاوی اطلاعات اصلی و حقایق مربوطه در متن است و پرس‌وجو (همچنین به‌عنوان کاربر محور یا متمرکز بر پرس‌وجو نیز نامیده می‌شود) که شامل ارائه اطلاعات مرتبط در مورد یک درخواست معین است طبقه‌بندی کرد [۱۴].

دوم، با توجه به تعداد اسنادی که باید در یک‌زمان خلاصه شوند و زبانی که خلاصه آن تولید می‌شود یا سند منبع نوشته‌شده است، خلاصه‌کننده‌ها را می‌توان به روش‌های تک سندی در مقابل

[1] Natural Language Processing
[2] Term Frequency - Inverse Document Frequency
[3] Latent Semantic Analysis

روش‌های چند سندی و رویکرد تک‌زبانه در مقابل چندزبانه دسته‌بندی کرد [۱۵،۱۶].

سوم، خروجی روش‌های خلاصه‌سازی می‌تواند به‌صورت اطلاعاتی در مقابل نماینده و استخراجی در مقابل انتزاعی باشد. یک خلاصه نماینده، برای نمایش آنچه که متن منبع در مورد آن است و موضوعاتی که پوشش داده‌شده است مورداستفاده قرار می‌گیرد. این نوع خلاصه‌ها نمی‌توانند جانشین سند مبدأ باشند و هدفشان صرفاً ارائه ایده درباره متن است. بنابراین، کاربر همچنان باید محتوای اصلی را مرور کند. یک خلاصه اطلاعاتی برای نمایش اطلاعات کافی کامل که در سند اصلی پوشش داده‌شده است استفاده می‌شود [۱۷].

در تکنیک خلاصه‌سازی استخراجی متن، مرتبط‌ترین و مهم‌ترین بخش‌ها از پیکره اصلی متن استخراج می‌شود. در برخی موارد دیگر، کاربر ممکن است مایل نباشد کل مقاله را برای به دست آوردن اطلاعات ضروری به‌طور کامل موردبررسی و تکرار قرار دهد. در چنین سناریوهایی، استخراج جمله برای خلاصه‌سازی متن ترجیح داده می‌شود، زیرا رویکرد بهتر و کارآمدتری برای این مورد است [۱۸].

در خلاصه‌سازی انتزاعی، خلاصه‌ای با عبارات جدید تولید می‌شود و درعین‌حال بر معنای متن مبدأ با استفاده از تکنیک‌ها و ابزارهای پیچیده پردازش زبان طبیعی دلالت می‌کند. اگرچه خلاصه‌نویس‌های انتزاعی پتانسیل سودمندی برای محققین دارند [۱۹،۲۰]، بیشتر مطالعات موجود بر خلاصه‌سازی استخراجی تمرکز دارند که در آن خلاصه‌ای از طریق شناسایی، انتخاب و کنار هم قراردادن مهم‌ترین بخش‌های منبع ساخته می‌شود. این تحقیق بر روی سیستم‌های خلاصه‌سازی عمومی، تک سندی، تک‌زبانه، اطلاعاتی و استخراجی تمرکز دارد.

استخراج ویژگی از سندهای متنی یکی از پرکاربردترین روش‌های خلاصه‌سازی متن است. این ویژگی‌ها بیشتر بر اساس اندازه‌گیری فرکانس سند متنی است. یکی از اولین و مهم‌ترین رویکرد ارائه‌شده روش TF-IDF است که نسخه‌های متفاوتی از آن معرفی‌شده است. بسیاری از خلاصه‌نویس‌های مختلف که به‌صورت آنلاین موجودند، با استفاده از این روش دقت و کیفیت بهتری در مقایسه با دیگر خلاصه‌نویس‌ها دارند [۲۱]. همچنین در پژوهش صورت گرفته توسط راموس، یک روش بازیابی کلمه مرتبط، مبتنی بر پرس‌وجو با استفاده از TF-IDF ارائه‌شده است [۲۲]. استفاده ترکیبی از رویکردهایی مانند خوشه‌بندی TF-IDF نیز در پژوهشی دیگر صورت گرفته است [۲۳].

یکی از راه‌های قابل‌توجه و برجسته برای مقابله با خلاصه‌سازی استخراجی متن، الگوریتم TextRank است [۲۴]. این الگوریتم یک روش برپایه گراف است که در آن جملات یا گاهی کلمات به‌عنوان رئوس یک گراف و وزن یال‌های بین آن‌ها به شباهت بین جملات یا کلمات بستگی دارد. این شباهت‌ها با ایجاد یک ماتریس شباهت [۲۵] محاسبه می‌شود. مزیت اصلی TextRank این است که یک الگوریتم مبتنی بر گراف بدون نظارت است، به‌عنوان‌مثال، برای تصمیم‌گیری در مورد ویژگی‌های اساسی یک سند متنی به هیچ خلاصه انسانی (یا مجموعه داده آموزشی) نیاز ندارد. الگوریتم‌های بدون نظارت نیاز به آماده‌سازی دستی کمتری دارند و بنابراین زمان کلی کمتری نسبت به الگوریتم‌های نظارت‌شده مشابه دارند [۲۶]. TextRank علاوه بر اینکه یک الگوریتم بدون نظارت است، یک سیستم ATS مستقل از زبان است که بر روی وقوع کلمه کارکرده و اهمیت جملات یا کلمات را محاسبه می‌کند و فقط جملات مفید و مرتبط را برای خلاصه خروجی انتخاب می‌کند.

الگوریتم TextRank در اوایل قرن بیست و یکم با الهام از روش PageRank معرفی‌شده است [۲۷،۲۸]. درواقع این روش برگرفته از الگوریتم PageRank گوگل است که یک الگوریتم رتبه‌بندی برای صفحات وب در دسترس بر اساس نتایج جستجو است [۲۹]. از این الگوریتم برای استخراج کلمات کلیدی و از رویکرد مبتنی بر گراف برای بازیابی اطلاعات مربوطه استفاده می‌شود. این الگوریتم از روش‌هایی برای محاسبه رابطه بین جملات استفاده می‌کند، شباهت کسینوسی یکی از آن‌ها است که با استفاده از ماتریس شباهت و فرمول TextRank این محاسبات انجام می‌گیرد [۳۰]. در پژوهش‌های دیگر ادغام TextRank و TF-IDF موردبررسی قرارگرفته [۳۱] و همچنین تغییراتی در TF-IDF سنتی

برای ایجاد نسخه‌های متفاوت با عملکرد و رویکرد بهتر اعمال‌شده است [۳۲].

از دیگر روش‌های پژوهش شده برای خلاصه‌سازی اسناد روش LDA است. در رویکرد LDA، داده‌های متنی در سطح کلمه و سند مدل‌سازی می‌شوند [۳۳]. خلاصه‌سازی چند سندی با استفاده ازLDA [۳۴]، برای داده‌های متنی با ورودی پیچیده مانند اسناد قانونی اعمال می‌شود [۳۵]، همان‌طور که در پژوهشی ثابت شد این روش یک رویکرد بسیار کارآمد برای خلاصه‌سازی است [۳۵]. رویکرد دیگری که رویکردی مبتنی بر انسجام است، روابط پایدار بین محتوا با استفاده از زنجیره‌های واژگانی را در نظر می‌گیرد، برای مثال (متضادها، تکرارها، مترادف‌ها و غیره) [۳۶]. نتایج نشان داد که خلاصه‌ها در این روش باکیفیت بالایی ایجاد می‌شوند. رویکردهای مبتنی بر یادگیری ماشینی شامل یادگیری با نظارت یا رویکردهای یادگیری نیمه نظارت‌شده و بدون نظارت است [۳۷]. در چند سال گذشته پیشرفت‌های زیادی در رویکردهای یادگیری بدون نظارت، به‌ویژه در تکنیک‌های خوشه‌بندی و یادگیری عمیق صورت گرفته است. ازجمله یک خلاصه‌کننده اسناد، مبتنی بر پرس‌وجو بر اساس ابزار OpenNLP و تکنیک خوشه‌بندی در سال ۲۰۱۲ معرفی شد [۳۸].

هر یک از روش‌ها و راهکارهای اشاره‌شده، با توجه معیارهای خود برای امتیازدهی جملات به خلاصه‌سازی متن می‌پردازند. در این تحقیق برخی از این روش‌ها و الگوریتم‌ها برای خلاصه‌سازی استخراجی متن را بررسی کرده و علاوه بر مقایسه آن‌ها با معیارهایی مانند فراخوانی، دقت و امتیاز به پیشنهاد روشی بانام Sa-TRB که برگرفته از TextRank و Bert است و از ترکیب جملات خروجی ایجادشده از این الگوریتم‌ها با دیگر الگوریتم‌ها استفاده می‌کند پرداخته می‌شود. درنهایت با انجام آزمایش‌هایی ثابت می‌شود که ایجاد یک سیستم شامل الگوریتم‌های متفاوت و ایجاد یک خلاصه استخراجی به کمک مجموعه‌ای از الگوریتم‌ها دقت خلاصه ایجادشده را افزایش داده و گاه نتایج و کیفیت خلاصه‌سازی را بهتر می‌کند.

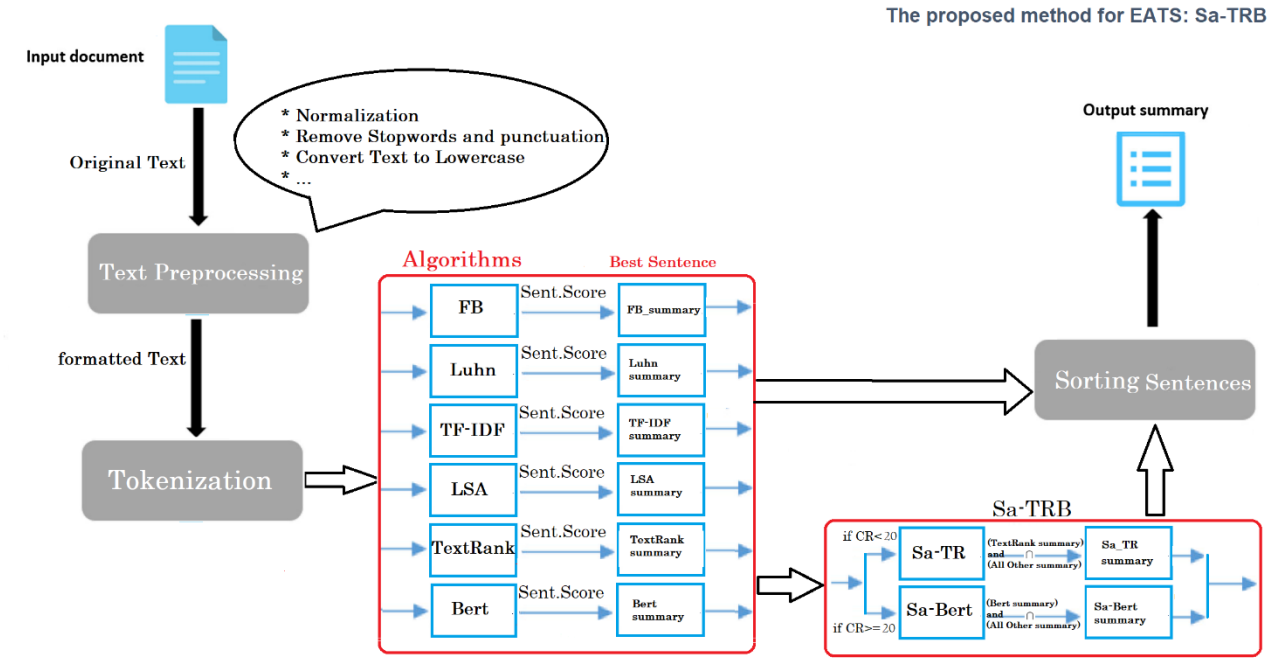
۳- راهکار پیشنهادی

در بخش‌های پیشین خلاصه‌سازی متن به‌صورت استخراجی، برخی از روش‌ها و الگوریتم‌هایی که در این زمینه وجود دارند موردبررسی قرار گرفت و حال نوبت آن است تا روش ارائه‌شده در این مقاله بیان شود. الگوریتم‌های مختلف، جملات متفاوتی را در یک متن برای خلاصه خود انتخاب می‌کنند و ازآنجاکه هیچ‌کدام از این الگوریتم‌ها دارای بالاترین امتیاز و بهترین کیفیت خلاصه‌سازی در تمامی انواع متن‌ها نیستند و معمولاً هرکدام با معیار خود مانند فرکانس و شباهت به امتیازدهی جملات می‌پردازد، استفاده از روشی که بتواند تمامی این الگوریتم‌ها را درنظر گرفته و اشتراک جملات به‌دست‌آمده در الگوریتم‌های مهم TextRank و Bert با اجتماع جملات سایر الگوریتم‌ها را به دست آورد، تضمین‌کننده افزایش دقت جملات به‌دست‌آمده برای خلاصه ایجادشده است. البته این کار باعث کاهش فراخوانی خلاصه ایجادشده نیز می‌شود، اما همان‌گونه که در بخش آزمایش خواهیم دید هنگامی‌که نرخ فشردگی خلاصه از یک آستانه بیشتر می‌شود نسبت افزایش دقت این روش به کاهش فراخوانی بیشتر بوده و باعث افزایش امتیاز خلاصه یا همان F1-score می‌گردد.

به بررسی بیشتر روش ارائه‌شده Sa-TRB در بخش الگوریتم‌ها خواهیم پرداخت، اما اکنون همان‌گونه که در شکل (۲) مشخص است، مراحل مختلفی که برای خلاصه‌سازی متن در این مقاله استفاده می‌شود نمایش داده‌شده است، که به توضیح هر بخش می‌پردازیم:

۳-۱ سند ورودی Input Document

سند ورودی، متن ورودی به سیستم خلاصه‌سازی است که می‌تواند هر سند متنی باشد اما به دلیل اینکه نیاز است خلاصه ایجادشده از این متن مورد ارزیابی قرار گیرد از دیتاست‌ها یا هر متنی که خلاصه آن قبلاً توسط انسان ایجادشده استفاده می‌گردد. در ادامه دیتاست یا همان مجموعه دادگان مورداستفاده در این مقاله بررسی می‌شود.



شکل (۲): طرح کلی خلاصه‌ساز پیشنهادی

مجموعه دادگان (دیتاست)

درزمینه پردازش زبان طبیعی و خلاصه‌سازی خودکار، دیتاست‌های بزرگ باکیفیت بالا از اهمیت ویژه‌ای برخوردار هستند. دیتاست CNN-DailyMail، یکی از منابع گرانبها در این زمینه است که به‌عنوان یک پلتفرم مناسب برای تحقیقات و آزمون مدل‌های پردازش زبان طبیعی مورداستفاده قرار می‌گیرد.

دیتاست CNN-DailyMail شامل بیش از ۳۰۰ هزار مقاله اخباری انگلیسی است که از دو منبع معتبر خبری تهیه‌شده است. هر مقاله شامل عنوان و متن کامل مقاله است، به‌علاوه یک خلاصه مناسب که بر اساس متن اصلی توسط انسان تولیدشده است. این دیتاست برای آموزش مدل‌های مبتنی بر شبکه‌های عصبی که قادر به تولید خلاصه‌های خودکار از متون هستند نیز کاربرد فراوان دارد.

از محدودیت‌های این دیتاست می‌توان به نداشتن دقت و کیفیت لازم برخی از خلاصه‌ها به دلیل ترجمه و تولید دستی بعضی از دستیاران انسانی، و همچنین حذف برخی اطلاعات موجود در مقالات مانند عکس‌ها، گفتار یا کدهای برنامه‌نویسی در دیتاست اشاره کرد.

دیتاست CNN-DailyMail با حجم بزرگ و کیفیت مناسب، یک منبع مهم برای تحقیقات و پژوهش‌های پردازش زبان طبیعی و خلاصه‌سازی خودکار به‌حساب می‌آید. این پژوهش‌ها شامل بررسی جمله‌بندی خودکار، تشخیص جنبه، مدل‌سازی زبان و سیستم‌های ترجمه ماشینی است.

با استفاده از این دیتاست، می‌توان مدل‌های پردازش زبان طبیعی و خلاصه‌ساز بهتری توسعه داد و نتایج واقعی‌تری را در این زمینه به دست آورد.

دیتاست CNN-DailyMail شامل سه بخش article یا متن اصلی، highlights یا خلاصه ایجادشده توسط انسان و id است.

۳-۲ پیش پردازش متن Text Preprocessing

در این مرحله متن ورودی ما برای استفاده بهتر الگوریتم‌ها و عملیاتی که روی آن صورت می‌گیرد پیش پردازش می‌شود. پیش پردازش شامل تبدیل تمام حروف به حروف کوچک، حذف کلمات غیرمهم (StopWords)، حذف علامت‌های اضافه (punctuation) و دیگر موارد می‌شود.

۳-۳ توکن‌سازی متن Tokenization

در این قسمت ما متن خود را توکن سازی می‌کنیم، در واقع متن به واحدهای کلمات و جملات تقسیم می‌شود، تا برای دسترسی به عناصر مختلف آن توسط الگوریتم‌ها راحت‌تر شود. با استفاده از کتابخانه NLTK زبان پایتون به راحتی می‌توان تمام واژگان و جملات متن را استخراج کرد. مرحله توکن سازی مرحله بسیار مهمی است زیرا الگوریتم‌ها علاوه بر استفاده مستقیم از توکن‌های مهم، از ویژگی‌های دیگر این توکن‌ها مانند فرکانس یا بردار مربوط به آن‌ها برای امتیازدهی جملاتی که آن توکن‌ها در جملات حضور دارند خواهند پرداخت.

۴-۳ الگوریتم‌ها Algorithms

در این بخش انواع مختلفی از الگوریتم‌ها وجود دارد که با امتیازدهی به جملات استخراج‌شده از متن سند اصلی می‌توانند در خلاصه‌سازی متن استفاده گردند. این الگوریتم‌ها عبارت‌اند از:

۱-۴-۳ الگوریتم FB[1]

الگوریتم FB الگوریتمی بر پایه فرکانس یا تکرار واژگان مهم است که یک روش اولیه بوده، و از رویکردی مستقیم برای محاسبه امتیاز جملات و درنهایت خلاصه‌سازی استفاده می‌کند. روش کار این الگوریتم به این صورت است که با جمع فرکانس واژگان مهم در هر جمله و تقسیم آن بر تعداد واژگان آن جمله امتیاز آن جمله را مشخص می‌کند. اگرچه این الگوریتم از لحاظ هوشمندی درجه پایینی دارد و فقط از معیار تکرار برای خلاصه‌سازی بهره می‌برد، اما به دلیل سادگی و سریع بودن برای مواردی که تکرار واژگان مهم نشان دهنده اهمیت آن واژه و جمله مربوط به آن است مورداستفاده قرار می‌گیرد.

۲-۴-۳ الگوریتم Luhn

الگوریتم لون یک الگوریتم قدیمی است که اگرچه این الگوریتم نیز از فرکانس واژگان برای امتیازدهی استفاده می‌کند اما به دلیل استفاده از دو مفهوم top_n_words یا n واژه مهم و همسایگی کلمات مهم، از رویکرد هوشمندتری نسبت به الگوریتم‌های بر پایه فرکانس بهره می‌برد. این الگوریتم در ابتدا n واژه مهم بر اساس تکرار در متن را پیدا می‌کند، سپس فاصله همسایگی این کلمات را بررسی می‌کند به این ترتیب هر جمله از یک یا چند گروه از این واژگان مهم با توجه به فاصله همسایگی تعریف شده تقسیم می‌شود و سپس به امتیازدهی هر گروه بر اساس فرمول (۱) می‌پردازد و درنهایت امتیاز هر جمله برابر با بالاترین امتیاز گروه در آن جمله است.

$$Group\_Score = \frac{|\text{Significant words in group}|^2}{|\text{All words in group}|}$$

$$Sentence\_Score = max\,(Group\_Score: for\ every\ groups\ in\ Sentence) \quad (۱)$$

۳-۴-۳ الگوریتم TF-IDF

این رویکرد که به‌عنوان فرکانس واژه در معکوس فرکانس سند نامیده می‌شود، یک رویکرد استخراج آماری است که با مقایسه فراوانی واژه در یک سند خاص در معکوس فرکانس آن واژه در اسناد دیگر کار می‌کند. به این معنی که اگر کلمه‌ای به‌طور مکرر در یک سند ظاهر شود، ممکن است فرض شود که برای سند مهم است، اما اگر همین کلمه به‌صورت پرتکرار در اسناد دیگر نیز آمده باشد، آن کلمه مهم نیست [۳۹]. TF-IDF با استفاده از فرمول (۲) برای یک واژه w محاسبه می‌شود.

$$TF(w) = \frac{Total\ count\ of\ appearance\ of\ a\ term\ w\ in\ D}{Total\ count\ of\ terms\ in\ D} \quad (۲)$$

$$IDF(w) = Loge\ \frac{Total\ count\ of\ Dn}{Total\ count\ of\ Dn\ with\ term\ w\ in\ it}$$

$$TF\_IDF(w) = TF(w) \times IDF(w)$$

D در اینجا نشان دهنده یک سند خاص است، Dn در اینجا مجموعه‌ای از اسناد را نشان می‌دهد.

با همین رویکرد برای محاسبه امتیاز یک جمله درمتن ابتدا امتیاز TF-IDF تمام واژگان مهم در متن محاسبه می‌شود، سپس جمع امتیاز واژگان تشکیل دهنده آن جمله تقسیم بر تعداد واژگان استفاده شده در جمله، امتیاز مربوط به آن جمله را محاسبه می‌کند. بنابراین امتیاز هر جمله در این روش با استفاده از فرمول (۳) محاسبه می‌شود.

$$Sentence\_Score = \frac{\sum (TF - IDF(w): for\ every\ words\ in\ Sentence)}{|All\ words\ in\ Sentence|} \quad (۳)$$

[1] Frequency Based

۳-٤-٤ الگوریتم TextRank

TextRank روشی است که به‌طور گسترده مورداستفاده قرار می‌گیرد، زیرا نیازی به دانش زبانی یا دامنه خاصی ندارد. این الگوریتم یک رویکرد بدون نظارت برای تولید خلاصه‌های مبتنی بر استخراج است.

در این الگوریتم ابتدا متن ورودی پیش پردازش می‌شوند. سپس متن به‌دست‌آمده از این مرحله به جملات تقسیم می‌شود. برای هر جمله به‌دست‌آمده، نمایش برداری آن (پس از حذف کلمات stopwords و به دست آمدن کلید واژه در مرحله پیش پردازش) استفاده می‌شود. معیارهای تشابه مختلفی وجود دارد که برای تعیین رابطه شباهت بین جملات بر اساس محتوای همپوشانی بین آن‌ها استفاده می‌شوند [٤٠]. همان‌طور که در رابطه (٤) نمایش داده‌شده به‌طور معمول از معیار تشابه کسینوسی در رویکرد TextRank استفاده می‌شود.

$$Cosine\ Similarity\ (S1,S2) = \frac{\vec{S1}.\vec{S2}}{|S1\ .\ S2\ |} \qquad (٤)$$

S1 و S2 بردارهایی هستند که برای نمایش جملات استفاده می‌شوند.

امتیاز شباهت به‌دست‌آمده از فرمول بالا در ماتریس شباهت جملات ذخیره می‌گردد. این رویکرد ماتریس شباهت را به گراف مدل می‌کند، جایی که گره‌های گراف جملات موجود در اسناد را نشان می‌دهند و لبه‌ها (یال‌ها) نشان‌دهنده رابطه معنایی است که از طریق آن جملات به هم متصل می‌شوند [٤١]. شباهت بین گره‌ها معادل وزن یال بین دو گره است. پس از محاسبه نمرات شباهت، رتبه بندی جملات انجام شد و خلاصه نهایی شامل جملات با رتبه برتر می‌شود.

برای نمونه اگر متنی دارای سه جمله A,B,C باشد، امتیازجمله A با استفاده از فرمول (٥) محاسبه می‌شود.

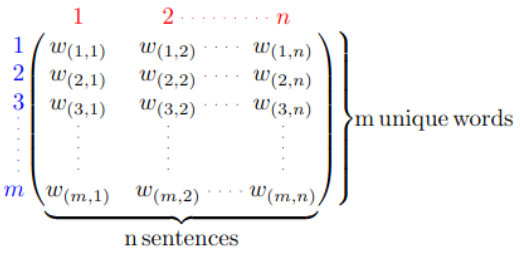
$$TextRank(A) = (1-d) + d \times (TextRank(B) \times M[A,B] + TextRank(C) \times M[A,C]) \qquad (٥)$$

که در آن d مقداری ثابت است و در ابتدا امتیاز TextRank تمام جملات برابر یک مقداردهی می‌شود و M ماتریس شباهت بین دو جمله است که می‌تواند با استفاده از فاصله کسینوسی، شباهت واژگان (کلمات مشترک) و یا روش‌های دیگر محاسبه شود. این امتیازدهی به‌صورت چندین چرخه تکرار می‌شود تا امتیاز تمام جملات آن به دست آید.

۳-٤-٥ الگوریتم LSA

الگوریتم بعدی از تحلیل معنایی پنهان (Latent semantic analysis) که یک تکنیک پردازش زبان طبیعی چند منظوره است استفاده می‌کند. یک سند به‌صورت یک ماتریس نشان داده می‌شود، و اعداد داخل ماتریس بیانگر ارتباط پنهان بین متن و واژه‌ها است. در ابتدا، کلمات منحصر به فرد از کل سند انتخاب می‌شود. سپس الگوریتم ماتریس $A_{m\times n}$ ساخته می‌شود، که در آن m تعداد کلمات منحصر به فرد و n تعداد جملات است، در حالی که $w_{i\times j}$ اهمیت کلمه i را در جمله j نشان می‌دهد. هر ستون A یک جمله است و سطرها هر کلمه منحصر به فرد را در رابطه با جمله نشان می‌دهند. الگوریتم اصلی از فرکانس–واژه وزنی به‌عنوان معیار اهمیت کلمه استفاده می‌کند. A ماتریس خلوت خواهدبود زیرا هر جمله حاوی تک تک کلمات سند نیست.

در شکل (٣) ماتریس A در این روش نمایش داده‌شده است.



شکل (٣): ماتریس A در روش LSA

در مرحله بعد، ماتریس A به‌صورت $A=U\Sigma V^T$ به سه بخش تجزیه می‌شود. $U_{m\times n}$ شامل بخشی از بردارهای ردیف در ماتریس A است و Σ n×n ماتریس مورب با مقادیر مرتب شده به ترتیب نزولی و $V^T_{n\times n}$ ماتریسی است که در آن هر جمله i با بردار ستونی نمایش داده می‌شود.

SVD یک نگاشت بین ماتریس A به مفاهیم استخراج‌شده در فضای برداری به‌صورت منفرد ایجاد می‌کند. این مفاهیم را می‌توان به‌عنوان الگویی از کلمات یا عباراتی در نظر گرفت که درزمینه‌های مشابه ظاهر می‌شوند، برای مثال، کلمات ماشین و اتوبوس در مفهوم وسیله نقلیه قرار می‌گیرند. V^T با توجه به اهمیت مفهومی

که ازجمله استخراج‌شده و در انتخاب جمله استفاده می‌شود، با درایه‌هایی که نشان دهنده رابطه بین مفاهیم و جملات هستند به ترتیب نزولی مرتب می‌شوند. در اجرای اصلی، جملات از مهم‌ترین مفاهیم استخراج‌شده انتخاب می‌شوند تا اینکه خلاصه‌ای که به دست می‌آید حاوی تعداد از پیش تعریف‌شده جملات باشد.

۳-٤-٦ الگوریتم BERT

خلاصه‌سازی استخراجی با استفاده از BERT یک روش مورداستفاده در پردازش زبان طبیعی است که بهترین مجموعه زیرمتن را از یک متن اصلی استخراج می‌کند و این مجموعه‌ها را به‌عنوان خلاصه‌ای از متن اصلی تولید می‌کند.BERT مخفف "Bidirectional Encoder Representations from Transformers" است و یک معماری مدل زبانی پیشرفته مبتنی بر شبکه‌های تبدیل‌دهنده است. این مدل به‌عنوان یکی از معروف‌ترین و قدرتمندترین مدل‌های زبانی در حوزه NLP شناخته می‌شود.

خلاصه‌سازی استخراجی متن با روش این خلاصه‌ساز شامل خلاصه‌کردن یک متن با استفاده از اطلاعات کلیدی استخراج‌شده از طریق مدل زبان طبیعی BERT است. این روش برای درک زبان‌های انسانی از RNN، مکانیسم‌های توجه و ترانسفورماتورها استفاده می‌کند. خلاصه‌سازی استخراجی BERT کنترل بر تعداد جمله و تعداد کاراکتر خلاصه‌سازی را فراهم می‌کند و برجسته‌ترین جملات و نهادهای معنادار برای توصیف معنای کلی یک سند متنی انتخاب می‌شود.

برای انجام خلاصه‌سازی با استفاده از این روش، ابتدا متن اصلی به بخش‌های کوچکتری مانند جملات تقسیم می‌شود. سپس هر بخش متن واردBERT شده تا بردارهای بازنمایی مختلفی از آن بدست آید. این بردارها عموماً به‌عنوان بردارهای کلیدی استفاده می‌شوند. پس از اعمال BERT روی تمام بخش‌ها، مجموعه‌ای از بردارهای کلیدی بدست می‌آید. سپس با اعمال الگوریتم‌هایی همچون الگوریتم خوشه‌بندی یا روش‌های دیگر، جملات مهم و جذاب برای خلاصه‌سازی را از میان این بردارهای بازنمایی برمی‌گزینیم.

[1] TextRank

جملات انتخاب شده به‌عنوان خلاصه تولید می‌شوند و به‌عنوان خروجی نهایی تحویل داده می‌شوند.

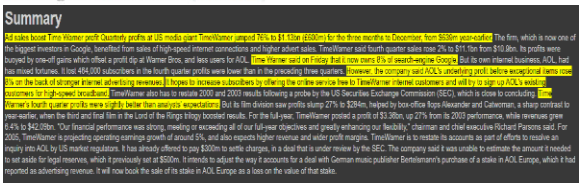
این روش خلاصه‌سازی، به علت داشتن مدلی بسیار قوی و توانایی فهم جملات به دلیل استفاده از بردار معنایی، خروجی‌هایی باکیفیت بالا تولید می‌کند.

۳-٤-۷ الگوریتم Sa-TRB

این الگوریتم که روش پیشنهادی در این مقاله است برگرفته از دو الگوریتم TR[1] و Bert است که از اشتراک مجموعه جملات خلاصه ایجادشده توسط TR یا Bert با اجتماع مجموعه جملات خلاصه ایجادشده توسط الگوریتم‌های دیگر استفاده می‌کند. بر اساس آزمایش‌های انجام شده مشخص گردید الگوریتم TR هنگامیکه CR[2] یا همان نرخ فشردگی کمتر از ۱۵ درصد است عملکرد بهتری نسبت به دیگر الگوریتم‌ها دارد، به این معنی که بهترین جمله‌ها را برای خلاصه انتخاب می‌کند اما هنگامی‌که نرخ فشردگی افزایش می‌یابد و مقدار آن بیش از ۲۰ درصد می‌شود، گاهی جملاتی که کمتر اهمیت دارند نیز در این الگوریتم انتخاب می‌شوند. ازآنجاکه خود الگوریتم TR به تنهایی نمی‌تواند این جملات کم اهمیت را پیدا کند از اجتماع خلاصه روش‌های دیگر برای یافتن این جملات استفاده می‌شود، به این معنی که اگر جمله انتخاب شده برای خلاصه توسط TRدر اجتماع جملات انتخاب شده توسط سایر الگوریتم‌ها حضور داشته باشند احتمال اینکه این جمله اهمیت زیادی دارد بیشتر است و برعکس یعنی اگر جمله انتخابی TR در مجموعه اجتماع جملات نباشد به احتمال زیاد جمله کم اهمیتی است. این موضوع هنگامی‌که CR از یک آستانه کمتر باشد چندان اهمیت خود را نشان نمی‌دهد، اما پس از عبور از یک آستانه اعمال این روش باعث افزایش دقت جملات انتخابی می‌شود. به همین صورت نیز هنگامی‌که CR بزرگ‌تر از ۲۰ درصد است، Bert عملکرد خوبی دارد بنابراین از اشتراک بین مجموعه جملات ایجادشده برای خلاصه توسط Bert با اجتماع جملات دیگر الگوریتم‌ها برای خلاصه‌سازی در این روش استفاده می‌شود.

در الگوریتم (۱) نحوه به دست آمدن جملات خلاصه نهایی در این روش نمایش داده‌شده است:

[2] Compression Rate



شکل (٤): خلاصه انتخاب شده بر روی متن سند اصلی

٤- آزمایش و ارزیابی

همان‌گونه که در بخش‌های قبل گفته‌شد سیستمی شامل چندین الگوریتم برای خلاصه‌سازی استخراجی متن طراحی شد. در اینجا برای آزمایش مدل خود از دیتاست cnn_dailymail به‌عنوان متن ورودی در مرحله اول استفاده می‌کنیم. این آزمایش با استفاده از زبان پایتون و در محیط colab پیاده‌سازی شده است. نتایج و ارزیابی‌ها نیز به‌صورت مستقیم در همین محیط به‌دست‌آمده است. پس از انجام پیش پردازش و توکن‌سازی متن از الگوریتم‌های مدل برای خلاصه‌سازی متن استفاده شده به‌صورتی که هر الگوریتم خلاصه متفاوتی از الگوریتم دیگر ایجاد می‌کند. این خلاصه‌های ایجادشده توسط این الگوریتم‌ها با خلاصه ایجادشده مرجع در دیتاست (highlights) مقایسه و ارزیابی می‌شوند.

برای بررسی کیفیت خلاصه نهایی معمولاً از معیار ROUGE[1] استفاده می‌شود، که در آن خلاصه‌ای که توسط ماشین ایجادشده با خلاصه‌ای که قبلاً توسط انسان ایجادشده مقایسه می‌گردد و با استفاده از معیارهای recall ،precision و F-score برای ROUGE-1(unigram)، ROUGE-2 (bigrams) ،ROUGE-L (longest matching sequence) کیفیت خلاصه ایجادشده توسط ماشین ارزیابی و امتیازدهی می‌گردد. در فرمول (٧) روش محاسبه این معیارها مشخص گردیده است.

$$Recall(R1) = \frac{Number\ of\ overlapping\ words}{Total\ words\ in\ reference\ summary}$$

$$Precision(R1) = \frac{Number\ of\ overlapping\ words}{Total\ words\ in\ system\ summary} \qquad (٧)$$

$$ROUGE1\ (F1-Score) = \frac{2\times(Recall(R1)\times Precision(R1))}{(Recall(R1)+\ Precision(R1))}$$

Sa-TRB:

1. if (CR<20) then: Sa-TRB =(TextRank-Summary) ∩ (Summary of all other algorithms)
   if Sa-TRB=Ø then: Sa-TRB= TextRank-Summary
2. if (CR>=20) then: Sa-TRB =(Bert-Summary) ∩ (Summary of all other algorithms)
   if Sa-TRB=Ø then: Sa-TRB= Bert-Summary (١)

٣-٥ مرتب‌سازی جملات Sorting Sentences

بعد از به دست آوردن امتیاز تمام جملات در متن سند، الگوریتم‌ها بهترین جملات انتخاب شده را بر اساس امتیازشان مرتب می‌کنند، اما نمی‌توان به همین ترتیب آن‌ها را برای خلاصه نمایش داد چراکه ممکن است باعث از بین رفتن بعضی از مفاهیم که به ترتیب نمایش آن‌ها بستگی دارد شود. بنابراین جملات انتخابی باید به ترتیبی که در سند متنی اصلی آمدند نمایش داده شوند. در این بخش خلاصه ایجادشده مرتب و آماده نمایش می‌شود.

CR متغیر کنترلی است که مشخص کننده درصد فشردگی متن خلاصه است، به این ترتیب اندازه خلاصه خروجی نیز با استفاده از فرمول (٦) به دست می‌آید.

$$FN = CR \times |S| \qquad (٦)$$

در این فرمول FN تعداد جملات خلاصه نهایی، CR نرخ فشردگی مورد نظر و |S| تعداد جملات متن اصلی است. بنابراین به تعداد FN به‌دست‌آمده در فرمول بالا از جملات با امتیاز بالابرای خلاصه نهایی انتخاب می‌شوند.

CR متغیر بسیار مهمی است چراکه بر روی نتایج نهایی که به‌صورت Rouge هستند تأثیر می‌گذارد بنابراین یافتن یک آستانه مناسب برای این متغیر بسیار مهم است.

٣-٦ خلاصه خروجی Output Summary

در مرحله آخر خلاصه نهایی به‌دست‌آمده نمایش داده می‌شود. همان‌گونه که پیشتر گفته شد این خلاصه شامل جملاتی از متن اصلی است که مهمتر از سایر جملات هستند و در مرحله قبل انتخاب شده و اکنون به‌صورت یک متن جدا یا مشخص شده در متن اصلی نمایش داده می‌شوند. در شکل (٤) خلاصه انتخاب شده بر روی متن سند اصلی نمایش داده‌شده است.

[1] Recall-Oriented Understudy for Gisting Evaluation

آزمایش در ۵ حالت مختلف CR یا همان نرخ فشردگی خلاصه مورد نظر، برای تمامی الگوریتم‌ها در ۱۰۰ سند متنی متفاوت در دیتاست cnn_dailymail نسخه سوم انجام شده است. در این آزمایش‌ها این نرخ فشردگی به ترتیب ۵٪، ۱۰٪، ۱۵٪، ۲۰٪ و ۲۵٪ در نظر گرفته شده است. نتایج به‌دست‌آمده از این آزمایشات در جدول (۱) و نمودارهای شکل (۵) و (٦) مشخص است:

جدول (۱): نتایج آزمایش

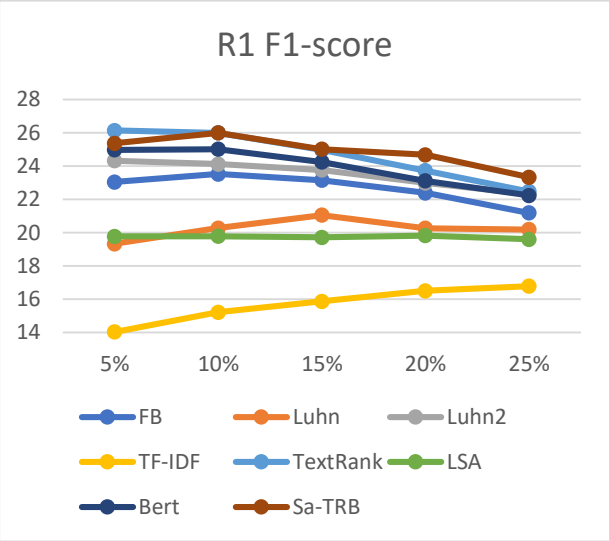
CR	ROUGE	FB	Luhn	Luhn2	TF-IDF	TextRank	LSA	Bert	Sa-TRB
5%	Recall (R1)	30.36	21.17	31.13	16.36	33.28	26.43	31.48	31.01
	Precision(R1)	19.00	18.71	20.44	12.94	22.00	16.17	21.11	22.57
	F1-Score (R1)	23.05	19.34	24.33	14.04	26.14	19.78	24.98	25.38
	Rouge2	7.13	4.62	7.44	2.65	8.66	4.21	7.73	8.70
	Rouge_L	21.25	17.35	21.96	12.67	23.91	17.57	22.73	23.06
	Average	17.14	13.77	17.91	9.79	19.57	13.85	18.48	19.05
10%	Recall (R1)	35.49	25.02	35.47	20.83	37.17	30.36	35.09	35.14
	Precision(R1)	18.27	18.07	19.27	12.72	20.77	15.32	20.03	21.82
	F1-Score (R1)	23.53	20.28	24.13	15.22	26.00	19.79	25.02	26.00
	Rouge2	7.32	4.84	7.40	2.84	8.56	4.52	7.79	8.85
	Rouge_L	21.78	18.34	21.79	13.90	23.96	17.75	22.87	23.82
	Average	17.54	14.49	17.77	10.65	19.51	14.02	18.56	19.56
15%	Recall (R1)	40.95	30.40	41.12	26.24	42.28	35.48	38.69	40.52
	Precision(R1)	16.85	17.20	17.72	12.12	18.48	14.39	18.29	19.14
	F1-Score (R1)	23.15	21.06	23.77	15.88	24.97	19.72	24.25	25.02
	Rouge2	7.18	5.41	7.77	3.40	8.00	4.46	7.49	8.13
	Rouge_L	21.48	19.12	21.82	14.49	23.11	17.94	22.11	23.09
	Average	17.27	15.20	17.79	11.26	18.69	14.04	17.95	18.75
20%	Recall (R1)	45.72	34.03	45.96	31.71	46.93	41.13	41.03	36.05
	Precision(R1)	15.59	15.48	16.29	11.94	16.70	13.82	16.81	20.04
	F1-Score (R1)	22.40	20.27	22.99	16.51	23.73	19.83	23.12	24.68
	Rouge2	7.01	5.24	7.67	3.49	7.56	4.78	6.99	8.01
	Rouge_L	20.93	18.55	21.36	15.28	21.98	18.13	21.41	22.61
	Average	16.78	14.69	17.34	11.76	17.76	14.25	17.17	18.43
25%	Recall (R1)	48.89	38.62	50.14	36.78	50.06	45.63	44.11	39.47
	Precision(R1)	14.11	14.60	15.19	11.53	15.20	13.13	15.50	17.49
	F1-Score (R1)	21.19	20.19	22.33	16.79	22.48	19.61	22.24	23.34
	Rouge2	6.54	5.32	7.38	3.59	7.15	5.04	6.58	7.17
	Rouge_L	19.90	18.65	20.93	15.50	20.94	18.22	20.51	21.35
	Average	15.88	14.72	16.88	11.96	16.86	14.29	16.44	17.29

۵- بحث و بررسی

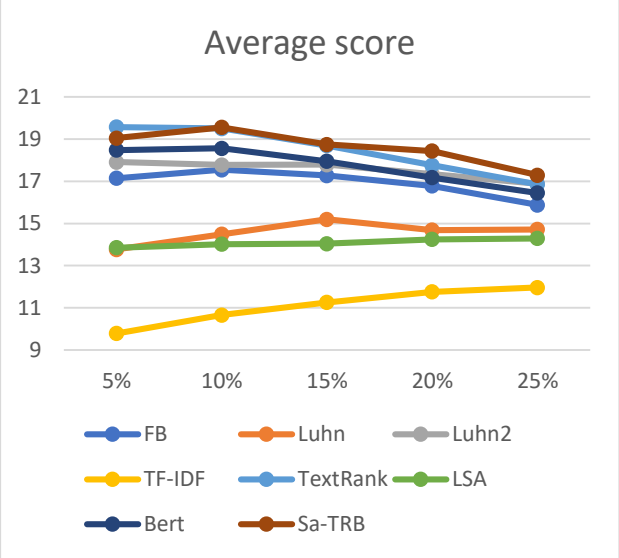
پس از به دست آمدن نتایج خلاصه‌سازی الگوریتم‌ها روی ۱۰۰ سند مختلف از دیتاست اکنون وقت آن است که به بررسی نتایج بپردازیم.

در بخش قبل الگوریتم‌ها روی دیتاست cnn_dailymail با نرخ فشردگی برابر ۵، ۱۰، ۱۵، ۲۰، ۲۵ درصد آزمایش شد و نتایج به‌دست‌آمده مشخص گردید. نتایج نشان دهنده این موضوع است که هنگامی‌که نیاز است خلاصه کوتاهی از متن داشته باشیم الگوریتم TR و الگوریتم‌های مشابه بهتر عمل کرده‌اند، اما هنگامی‌که از یک آستانه نرخ فشردگی عبور می‌کنیم، الگوریتم‌های برت و الگوریتم‌های مشابه به تدریج نتایج بهتری را نمایش داده‌اند. روش ارائه‌شده Sa-TRB با آگاهی از این منطق هنگامی‌که نیاز به خلاصه کوتاه است مانند TR عمل می‌کند و به دلیل استفاده از یک رویکرد برای افزایش دقت که به کمک دیگر الگوریتم‌های این سیستم انجام می‌گیرد، اگر چه هنگامی‌که نرخ فشردگی ۵ و ۱۰ درصد باشد عملکرد ضعیف‌تری نسبت به TR دارد اما پس از آن هنگامی‌که نرخ فشردگی ۱۵، ۲۰ و ۲۵ درصد است، عملکرد بهتری را نمایش می‌دهد. دلیل این بهتر شدن این است که روش پیشنهادی همیشه دقت بیشتر اما فراخوانی کمتری نسبت به TR دارد، اما هنگام افزایش نرخ فشردگی خلاصه از یک حد، نسبت بهتر بودن دقت روش پیشنهادی به روش TR بیشتر از کاهش فراخوانی شده و درنتیجه امتیاز بیشتری به دست می‌آورد. همچنین هنگامی‌که نیاز به یک خلاصه بلندتر داریم الگوریتم Bert که در این زمینه بهتر عمل می‌کند به کمک الگوریتم پیشنهادی آمده تا با محوریت این الگوریتم جملاتی از متن انتخاب گردد که علاوه بر حضور در خلاصه نهایی الگوریتم Bert، در اجتماع سایر الگوریتم‌ها نیز

حضور داشته باشد، تا روش پیشنهادی Sa-TRB بهترین نتایج را به دست آورد.



شکل (۵): نمودار امتیاز R1 الگوریتم‌ها



شکل (٦): نمودار امتیاز میانگین الگوریتم‌ها

در دو شکل (۵) و (٦) عملکرد روش پیشنهادی Sa-TRB در ۵ نرخ فشردگی مختلف نمایش داده‌شده است. در نمودار اول R1 F1-score الگوریتم‌ها مقایسه گردیده و دیده می‌شود که پس از عبور از ۱۰٪ نرخ فشردگی روش پیشنهادی بهتر از سایر الگوریتم‌هاست. همچنین در نمودار دوم که میانگین امتیاز R1، R2، RL است نمایش داده‌شده است. در اینجا نیز همان‌گونه که مشاهده می‌شود روش پیشنهادی باوجود نتایج نزدیک هنگام افزایش نرخ فشردگی عملکرد بهتری را نمایش می‌دهد.

٦- نتیجه‌گیری

خلاصه‌سازی متن فرایندی است که در آن یک سند متنی به خلاصه‌ای از بخش‌های آن تبدیل می‌شود به‌صورتی که در آن ارتباط موضوع و بخش‌های مهم از بین نمی‌روند. هرچقدر این اصل رعایت گردد کیفیت خلاصه نهایی افزایش خواهد یافت. دو روش خلاصه‌سازی به‌صورت استخراجی و انتزاعی برای این منظور وجود دارد. در این مقاله به بررسی تکنیک‌هایی مانند TextRank، TF-IDF، BERT، LSA و چند الگوریتم دیگر که به‌صورت گسترده برای خلاصه‌سازی استخراجی استفاده می‌شود پرداخته شد. برای مقایسه این الگوریتم‌ها از معیارهای Rouge همچون فراخوانی، دقت و F1-score استفاده شد. همچنین رویکردی ترکیبی از همین الگوریتم‌ها بانام Sa-TRB پیشنهادگردید که در بازه ۱۵ تا ۲۵ درصد آزمایش‌شده برای نرخ فشردگی خلاصه عملکرد بهتری نسبت به الگوریتم‌های موردبررسی نشان داد. این عملکرد بهتر به دلیل افزایش دقت در انتخاب مهم‌ترین جملات برای دو الگوریتم TextRank و Bert با استفاده از این منطق که جمله‌ای که در خلاصه چند الگوریتمهای مهم حضور دارد احتمالاً مهمتر است به دست آمد.

References

[1] S. Soumya, S. Kumar, R. Naseem and S. Mohan, Automatic Text Summarization, In: Das V.V., Thankachan N. (eds) , Computational Intelligence and Information Technology. CIIT 2011, Communications in Computer and Information Science (CCIS), Springer, Berlin, Heidelberg, Volume 250, pp. 787-789, 2011.

[2] R. M. Aliguliyev, N. R. Isazade and N. Idris, COSUM: Text summarization based on clustering and optimization, Expert Systems: The Journal of Knowledge Engineering, Volume 36, Issue 1, 2019.

[3] H. Jing, Sentence Reduction for Automatic Text Summarization, Sixth Applied Natural Language Processing Conference. Association for Computational Linguistics, pp. 310–315, 2000.

[4] K. Knight and D. Marcu, Summarization beyond sentence extraction: A probabilistic approach to

[17] E. Lloret and M. Palomar, Text summarisation in progress: a literature review, Artif. Intell. Rev. 37 (1), pp. 1–41, 2012.
[18] R.A. García-Hernández, R. Montiel, Y. Ledeneva, E. Rendón and A. Gelbukh, Cruz, R. Text Summarization by Sentence Extraction Using Unsupervised Learning., In Proceedings of the Mexican International Conference on Artificial Intelligence, Atizapán de Zaragoza, Mexico, 27–31 October 2008; Springer: Berlin/Heidelberg, Germany, 2008.
[19] M. Fiszman, D. Demner-Fushman, H. Kilicoglu and T.C. Rindflesch, Automatic summarization of MEDLINE citations for evidence-based medical treatment: A topicoriented evaluation, J. Biomed. Inform. 42 (5), pp. 801–813, 2009.
[20] H. Zhang, M. Fiszman, D. Shin, C.M. Miller, G. Rosemblat and T.C. Rindflesch, Degree centrality for semantic abstraction summarization of therapeutic studies, J. Biomed. Inform. 44 (5), pp. 830–838, 2011.
[21] H. Christian, M.P. Agus and D. Suhartono, Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF)., ComTech: Computer, Mathematics and Engineering Applications, 7(4), pp. 285-294, 2016.
[22] J. Ramos, Using tf-idf to determine word relevance in document queries., In Proceedings of the first instructional conference on machine learning, Vol. 242, pp.133-142, December 2003
[23] P. Bafna, D. Pramod, and A.Vaidya, Document clustering: TF-IDF approach., In 2016 International Conference on Electrical, Electronics, and Optimization Techniques IEEE (ICEEOT), pp. 61-66, March 2016.
[24] R. Mihalcea and P. Tarau, Textrank: Bringing order into text., In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, pp. 25–26, July 2004.
[25] P. Zha, X.Xu and M. Zuo, An Efficient Improved Strategy for the PageRank Algorithm., In Proceedings of the 2011 International Conference on Management and Service Science, Bangkok, Thailand, pp. 7–9, May 2011.
[26] N. Moratanch and S.A Chitrakala, survey on extractive text summarization. In Proceedings of the 2017 International Conference on Computer, Communication and Signal Processing (ICCCSP), Chennai, India, pp. 10–11, January 2017.
[27] R. Mihalcea and P. Tarau, Textrank: Bringing order into text., In Proceedings of the 2004 sentence compression, Artificial Intelligence, Volume 139, Issue 1, pp. 91-107, 2002.
[5] I. Mani, Automatic summarization, John Benjamin's Publishing Company, Amsterdam/Philadelphia, 2001.
[6] P. Kouris, G. Alexandridis and A. Stafylopatis, Abstractive Text Summarization Based on Deep Learning and Semantic Content Generalization, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp. 5082–5092, 2019.
[7] A. See, P. J. Liu and Ch. D. Manning, Get To The Point: Summarization with Pointer-Generator Networks, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1073-1083, 2017.
[8] S. Singhal and A. Bhattacharya, Abstractive Text Summarization, pp. 1-11, 2015.
[9] J.N. Madhuri and R. Ganesh Kumar, Extractive Text Summarization Using Sentence Ranking, 2019 International Conference on Data Science and Communication (IconDSC), IEEE, pp. 1-3, 2019.
[10] M. Gambhir and V. Gupta, Recent automatic text summarization techniques: a survey, Artificial Intelligence Review, Volume 47, Issue 1, pp. 1–66, 2017.
[11] S. Ghodratnama, A. Beheshti, M. Zakershahrak and F. Sobhanmanesh, Extractive document summarization based on dynamic feature space mapping, IEEE Access 2020, 8. [CrossRef], pp. 139084–139095, 2020.
[12] H. P. Luhn, The automatic creation of literature abstracts, IBM Journal of research and development, 2(2), pp. 159-165, 1958.
[13] H. P. Luhn, A statistical approach to mechanized encoding and searching of literary information, IBM J. Res. Dev.1957.1.[CrossRef] pp. 309–317, 1957.
[14] R. Mishra, J. Bian, M. Fiszman, C.R. Weir, S. Jonnalagadda, J. Mostafa and et al., Text summarization in the biomedical domain: A systematic review of recent research, J.Biomed. Inform. 52, pp. 457–467, 2014.
[15] S. Afantenos, V. Karkaletsis and P. Stamatopoulos, Summarization from medical documents: a survey, Artif. Intell. Med. 33 (2) pp. 157–177, 2005.
[16] V. Gupta and G.S. Lehal, A survey of text summarization extractive techniques, J. Emerg. Technol. Web Intell. 2, pp. 258–268, 2010.

conference on empirical methods in natural language processing, pp. 404-411, July 2004.
[28] S. Brin and L. Page, The anatomy of a large-scale hypertextual web search engine., 1998.
[29] C. Mallick, A. K. Das, M. Dutta and A. Sarkar, Graphbased text summarization using modified TextRank., In Soft Computing in Data Analytics. Springer,Singapore, pp. 137-146, 2019.
[30] F. Barrios, F. López, L. Argerich, and R. Wachenchauzer, Variations of the similarity function of textrank for automated summarization., arXiv preprint arXiv:1602.03606, 2016.
[31] L. Yao, Z. Pengzhou and Z. Chi, Research on News Keyword Extraction Technology Based on TF-IDF and TextRank., In 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS). IEEE Computer Society , pp. 452-455, June 2019.
[32] R.K. Roul, J.K. Sahoo and K. Arora, Modified TF-IDF term weighting strategies for text categorization., In 2017 14th IEEE India Council International Conference (INDICON), pp. 1-6, December 2017.
[33] Y.L. Chang and J.T. Chien, Latent Dirichlet learning for document summarization., In 2009 IEEE international conference on acoustics, speech and signal processing, pp. 1689-1692, April 2009.
[34] R. Arora and B. Ravindran, Latent Dirichlet allocation based multi-document summarization., In Proceedings of the second workshop on Analytics for noisy unstructured text data, pp. 91-97, July 2008.
[35] R. Kumar and K. Raghuveer, Legal document summarization using latent dirichlet allocation., International Journal of Computer Science Telecommunications. 3, pp. 114-117, 2012.
[36] A.C. Onwutalobi, Using Lexical Chains for Efficient Text Summarization., Available online: https://ssrn.com/abstract=3378072, accessed on 16 May 2009.
[37] J.L. Neto, A.A. Freitas and C.A.A. Kaestner, Automatic text summarization using a machine learning approach., In Proceedings of the Brazilian Symposium on Artificial Intelligence, Porto de Galinhas, Recife, Brazil, 11–14 November 2002, Springer: Berlin/Heidelberg, Germany, 2002.
[38] H.J. Jain, M.S. Bewoor and S.H. Patil, Context sensitive text summarization using k means clustering algorithm., Int. J. Soft Comput.Eng, 2,pp. 301–304, 2012.
[39] L.Yao, Z.Pengzhou and Z.Chi, Research on News Keyword Extraction Technology Based on TF-IDF and TextRank., In 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS), pp. 452-455, June 2019.
[40] F. Barrios, F.López, L. Argerich, and R. Wachenchauzer, Variations of the similarity function of textrank for automated summarization., arXiv preprint arXiv:1602.03606, 2016.
[41] S. R.Manalu and A.M.Sundjaja, Review assessment support in Open Journal System using TextRank., JPhCS, 801(1), 012074, 2017.

Journal of Applied And Basic Machine Intelligence Research (ABMIR)

Applied and Basic Machine Intelligence Research

Vol. 1, No. 2, Autumn & Winter 2023, pp. 145-159

Extractive Automatic Text Summarization using integrated set of algorithms and Sa-TRB method

Abolfazl Sadrolsadati [1*], Mohammad-Reza Feizi-Derakhshi[2]

[1]Department of Computer Engineering, Faculty of Electrical & Computer Engineering,Pardis-aras, University of Tabriz, Tabriz, Iran

[2]Department of Computer Engineering, Faculty of Electrical & Computer Engineering,University of Tabriz, Tabriz, Iran

Article Information

Original Research Paper

Received:
2023 September 19

Accepted:
2023 December 4

Keywords:
Extractive text summarization, NLP, Luhn, TF-IDF, TextRank, BERT, LSA, Sa-TRB, CR, ROUGE

Corresponding Author*:
Abolfazl.Sadrolsadati@Gmail.com

Abstract

Extractive summarization of text is an essential technique in natural language processing, which helps to produce compact versions of text by extracting the most important sentences. Since the task of shortening and summarizing a text document is time-consuming and exhausting, an automatic system for creating these short versions of the text seems necessary. In extractive summarization, sentences that contain useful and relevant information are usually selected for the final summary. In order to identify these sentences, there are different algorithms, the performance and summary created by each one is different based on the type and scope of the text and the size of the required summary. In this article, a method called Sa-TRB is presented, which is derived from two algorithms, TextRank and BERT, and in addition to using these two methods, it also uses the common sentences created by other algorithms to achieve high accuracy in selection. Have final summary sentences. The most important criterion for evaluating the performance of algorithms is the quality of their final summary, so the more the final summary created by these algorithms is similar to the summary created by humans, the better the quality of the created summary is. ROUGE criteria have been used to obtain the size of this similarity. Finally, by conducting experiments on the cnn-dailymail dataset with different sizes of summaries, it is shown that the proposed method, by increasing the size of the required summaries, despite the decrease in the recall criterion, has accuracy, score and, as a result, higher quality of the final summaries. So, in the last two tests, the score of the proposed method has reached 24.68 and 23.34%, which is almost one percent better than the best tested methods.

: 10.22034/ABMIR.2023.20650.1035

E-ISSN: 2821-2037