A New Persian Text Summarization Approach Based on Natural Language Processing and Graph Similarity

Tayyebeh Hosseinikhah
M.Sc. in Industrial Engineering Amirkabir University of Technology;
hoseinikhah@aut.ac.ir

Abbas Ahmadi
Assistant Professor; Department of Industrial Engineering and Management Systems; Amirkabir University of Technology;
Corresponding Author abbas.ahmadi@aut.ac.ir

Azadeh Mohebi
Assistant Professor; Faculty of Information Technology;
Iranian Research Institute for Information Science and Technology (IranDoc) mohebi@irandoc.ac.ir

Iranian Journal of Information Processing and Management

Iranian Research Institute for Science and Technology
ISSN 2251-8223
eISSN 2251-8231
Indexed by SCOPUS, ISC, & LISTA
Vol. 33 | No. 2 | pp. 885-914
Winter 2018

Received: 23, Apr. 2015 **Accepted: 08, Jan. 2017**

Abstract: A significant amount of available information is stored in textual databases which contain a large collection of documents from different sources (such as news, articles, books, emails and web pages). The increasing visibility and importance of this class of information motivates us to work on having better automatic evaluation tools for textual resources.

The automatic summarization of text is one of the ways to prevent the waste of users' time. The extractive text summarization consists of the extraction of the more important sentences with the purpose of shortening input text while maintaining the topics covered and the subjects discussed.

In this paper, we have tried to improve the accuracy of the extracted summaries by combining natural language processing and text mining techniques. By modifying the mentioned algorithms and sentence scoring measures, accuracy is increased as compared to the previously used techniques.

Part of speech tagging is used for calculating coefficient of words' importance. Using this approach will in turn help us with picking the more meaningful words and phrases that will result in better accuracy of the system.

Graph similarity's methods are used to select sentences. Changing weight of the selected sentences in each step leads to solve the redundancy problem.

Standard evaluation measures such as “Precision” and “Recall” are used to evaluate results based on a Persian corpus.

Keywords: Extractive Summarization, Natural Language Processing, Text Mining, Part of Speech Tagging, Similarity Graph

بهبود خلاصه‌سازی خودکار متون فارسی با استفاده از روش‌های پردازش زبان طبیعی و گراف شباهت

طیبه حسینی‌خواه
کارشناس ارشد مهندسی صنایع؛
دانشگاه صنعتی امیرکبیر hoseinikhah@aut.ac.ir

عباس احمدی
دکتری مهندسی صنایع؛ استادیار؛
دانشگاه صنعتی امیرکبیر؛
پدیدآور رابط abbas.ahmadi@aut.ac.ir

آزاده محبی
دکتری مهندسی طراحی سیستم‌ها؛ استادیار؛
پژوهشگاه علوم و فناوری اطلاعات ایران (ایرانداک)؛
mohebi@irandoc.ac.ir

پژوهش‌نامهٔ پردازش و مدیریت اطلاعات

دریافت: ۱۳۹۴/۰۲/۰۳ | پذیرش: ۱۳۹۵/۱۰/۱۹ | مقاله برای اصلاح به مدت ۹ ماه نزد پدیدآوران بوده است.

فصلنامه | علمی پژوهشی
پژوهشگاه علوم و فنّاوری اطلاعات ایران
شاپا (چاپی) ۸۲۲۳-۲۲۵۱
شاپا (الکترونیکی) ۸۲۳۱-۲۲۵۱
نمایه در LISTA، ISC، و SCOPUS
jipm.irandoc.ac.ir
دورهٔ ۳۳ | شمارهٔ ۲ | ص‌ص ۸۸۵-۹۱۴
زمستان ۱۳۹۶

چکیده: پایگاه داده‌های متنی شامل مجموعهٔ بزرگی از اسناد و منابع مختلف (مانند مقالات خبری، کتاب‌ها، ایمیل‌ها و صفحات وب) است. افزایش چشمگیر این نوع اطلاعات وجود ابزارهایی برای ارزیابی خودکار منابع متنی را بیش از هر زمان دیگری آشکار می‌سازد. در این میان خلاصه‌سازی خودکار متون یکی از راه‌کارهایی است که از اتلاف وقت کاربران می‌کاهد. خلاصه‌سازی استخراجی به معنای انتخاب مهم‌ترین جملات یک متن با هدف کوتاه‌نمودن آن است به شکلی که اطلاعات مهم متن ورودی را دربر داشته باشد. در این پژوهش با به‌کارگیری و ترکیب روش‌های پردازش زبان طبیعی دقت خلاصه‌های استخراجی بهبود می‌یابد و روشی برای اصلاح الگوریتم‌ها و معیارهای امتیازدهی به جملات ارائه می‌شود. در روش پیشنهادی برای امتیازدهی به کلمات از برچسب نقش دستوری کلمات در جمله به‌عنوان ضریب اهمیت کلمات استفاده می‌شود که در نتیجه، با انتخاب بهتر کلمات و جملاتی که بار محتوایی بیشتری دارند، دقت خلاصه‌سازی افزایش می‌یابد. علاوه بر آن، برای انتخاب جملات مناسب از متن از روش‌های مبتنی بر گراف شباهت استفاده می‌شود، به‌گونه‌ای که با تغییر وزن جملات انتخاب‌شده در پیمایش گراف، در هر گام چالش افزونگی اطلاعات برطرف می‌گردد. در نهایت، نتایج به‌دست‌آمده با معیارهای استانداردی مانند «بازخوانی» و

«دقت» و بر روی یک پیکرهٔ متنی استاندارد فارسی نیز ارزیابی می‌شود.

کلیدواژه‌ها: خلاصه‌سازی استخراجی، پردازش زبان طبیعی، برچسب‌گذاری دستوری کلمات، گراف شباهت

۱. مقدمه

رشد چشمگیر دسترسی به اطلاعات و استفادهٔ روزافزون از اینترنت باعث گردیده که حجم اطلاعات، اسناد و مقالات برخط به‌صورتی فزاینده افزایش یابد و در نتیجه، دسترسی به اطلاعات مفید و در عین حال خلاصه از میان انبوه اطلاعات دشوار گردد. در این میان، وجود ابزارهایی که به‌صورت خودکار اطلاعات مفید را از میان منابع مختلف در کمترین زمان استخراج کند، بیش از هر زمان دیگری احساس می‌شود. این است که با توجه به حجم زیاد اطلاعات و منابع متعدد، وجود ابزارها و روش‌هایی برای خلاصه‌سازی متون ضروری است. خلاصه‌سازی به فرایند تجمیع اطلاعات مفید یک یا چند منبع اطلاعاتی به‌منظور کوتاه‌کردن متن یا پاسخگویی به درخواست کاربر گفته می‌شود (Mani & Maybury 1999). خلاصه‌سازی خودکار به معنای استفاده از ابزارهای ماشینی و مبتنی بر کامپیوتر برای تولید یک خلاصهٔ مفید و معتبر است و یکی از مسائل مشکل و پرچالش در زمینهٔ پردازش زبان طبیعی به حساب می‌آید، از آن جهت که هنوز کیفیت خلاصه‌های تولیدشدهٔ ماشینی به اندازهٔ خلاصه‌های انسانی نیست (Nenkova & McKeown 2012). سیستم‌های خلاصه‌ساز از منظر منابع به خلاصه‌سازهای تک‌سندی یا خلاصه‌سازهای چندسندی، و از منظر نوع خلاصه به خلاصهٔ استخراجی[1] و خلاصهٔ چکیده‌ای[2] تقسیم بندی می‌شوند. در خلاصه‌سازی استخراجی جملات مهم متن ورودی عیناً در خلاصه می‌آید. این روش در مقایسه با نقطهٔ مقابل آن، یعنی خلاصه‌سازی چکیده‌ای، از پیچیدگی کمتری برخوردار است. خلاصه‌سازی چکیده‌ای با تغییر در ساختار جملات سعی در تولید جملات جدید برای خلاصه دارد (Hovy 2003). روش‌های خلاصه‌سازی خودکار به ساختار زبانی متون نگارش‌شده در آن زبان وابسته هستند و روش‌های پردازش زبان نقش اصلی را در ایجاد خلاصه‌ساز خودکار ایفا می‌کنند. البته، برای روش‌های خلاصه‌ساز استخراجی

1. extractive
2. abstractive

برخی از روش‌های موجود برای یک زبان را می‌توان با تغییراتی محدود برای زبانی دیگر نیز به‌کار برد. ولی در خلاصه‌سازی چکیده‌ای لازم است که کاملاً از اصول نگارش زبانی استفاده شود. تاکنون ابزارها و سیستم‌های متعددی برای تولید خلاصه‌ها در زبان انگلیسی ایجاد شده و پژوهش‌های قابل توجهی در این زمینه صورت گرفته است (Gambhir & Gupta 2016; Das & Martins 2007) که حجم بسیاری از آن‌ها در زمینهٔ خلاصه‌ساز استخراجی است. در زبان فارسی نیز سیستم‌های خلاصه‌ساز استخراجی متنوعی ایجاد شده است که سیستم «فارسی‌سام»[1] جزو اولین این سیستم‌هاست (Hassel & Mazdak 2004). یکی از مهم‌ترین چالش‌ها برای یک سیستم خلاصه‌ساز استخراجی، مرحلهٔ پیش‌پردازش است که در آن متن مورد نظر برای استخراج جملات خلاصه، عمدتاً بر اساس عملگرهای پردازش زبان طبیعی نظیر ریشه‌یابی، حذف کلمات توقف، برچسب‌زنی نقش کلمات، و تعیین کلمات کلیدی پردازش می‌شود. پس از آن، به هر جمله در متن امتیازی تعلق می‌گیرد و در نهایت، جملات با امتیاز بالا انتخاب می‌شوند. از آنجا که کلمات کلیدی نقش به‌سزایی در تعیین امتیاز جملات ایفا می‌کنند، مرحلهٔ پیش‌پردازش اهمیت ویژه‌ای را در خلاصه‌ساز استخراجی دارد. در این مقاله روشی ارائه می‌شود که بر اساس آن نقش کلمات نیز در مرحلهٔ پیش‌پردازش به‌عنوان یک ویژگی مهم برای انتخاب کلمات کلیدی و در نهایت، جملات مناسب در نظر گرفته خواهد شد.

از طرف دیگر، در بین روش‌های خلاصه‌ساز استخراجی، روش‌های مبتنی بر گراف نتایج اثربخشی ارائه نموده‌اند (Erkan & Radev 2004; Nenkova & McKeown 2012) و تعدادی از پژوهش‌های خلاصه‌ساز استخراجی در زبان فارسی نیز روی به‌کارگیری این‌گونه روش‌ها تمرکز داشته‌اند (Shakeri et al. 2012). لیکن، یکی از چالش‌های اصلی در این روش‌ها، روش پیمایش گراف برای انتخاب جملات مناسب است، به‌گونه‌ای که هر جمله در عین حال که لازم است منعکس‌کنندهٔ محتوای متن باشد و کلمات کلیدی مناسب را دربر داشته باشد، باید کمترین اشتراک را با جملات قبلی انتخاب‌شده نیز داشته باشد. در این پژوهش رویکردی برای برطرف‌نمودن این چالش ارائه می‌شود.

ساختار مقاله به این ترتیب است: در بخش دوم، مفاهیم اصلی خلاصه‌سازی و انواع خلاصه به‌طور مختصر ارائه می‌شود. پس از آن، ادبیات موضوع بر اساس روش‌های

1. FarsiSum

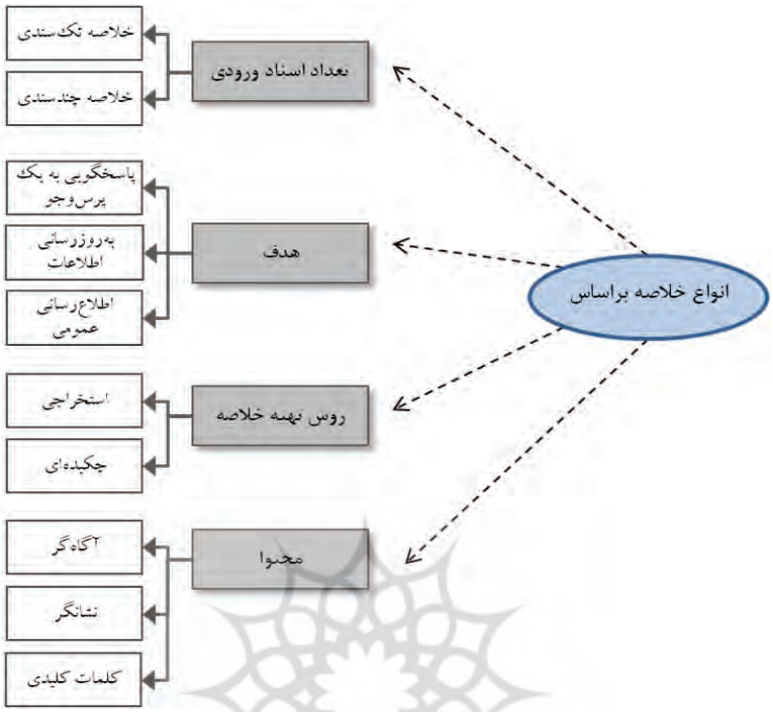
خلاصه‌ساز استخراجی در زبان انگلیسی و فارسی بررسی می‌شود. در بخش چهارم، روش پیشنهادی تشریح می‌گردد و در بخش پنجم، نتایج پیاده‌سازی روش روی یک مجموعه دادهٔ استاندارد ارائه می‌شود. در نهایت، نتیجهٔ حاصل از این پژوهش و تحقیقات پیشنهادی آتی تشریح می‌گردد.

۲. مفاهیم خلاصه‌سازی

در این بخش مروری کوتاه بر مفاهیم اصلی خلاصه‌سازی شامل انواع خلاصه و رویکردهای اصلی برای خلاصه‌سازی ارائه می‌شود.

۲-۱. انواع خلاصه

خلاصه‌ها را از جهات مختلف می‌توان دسته‌بندی نمود. نوع خلاصهٔ تهیه‌شده بسته به تعداد اسناد یا متون در دست، هدف از خلاصه‌سازی، روش مورد استفاده برای تهیهٔ خلاصه، و نوع محتوایی که خلاصه باید دربر داشته باشد، می‌تواند متفاوت باشد (Nenkova & McKeown 2012). شکل ۱، دسته‌بندی انواع روش‌های خلاصه‌سازی را بر اساس نوع خلاصه بر مبنای دسته‌بندی «ننکووا و مک‌کوئن» نشان می‌دهد.



شکل ۱. دسته‌بندی انواع خلاصه (Nenkova & McKeown 2012)

خلاصهٔ تولیدشده می‌تواند حاصل پردازش یک سند یا چندین سند باشد که در اصطلاح به آن خلاصه‌های تک‌سندی یا چندسندی می‌گویند.

هدف از تهیهٔ خلاصه می‌تواند در نوع خلاصهٔ تولیدشده مؤثر باشد. بر اساس تقسیم‌بندی که Nenkova & McKeown(2012) در مقالهٔ مروری خود در زمینهٔ روش‌های خلاصه‌سازی متون ارائه کردند، یک خلاصه می‌تواند در راستای پاسخگویی به یک عبارت پرس‌وجو[1] تولید شود. در این حالت مطالبی در خلاصه گنجانده خواهد شد که با عبارت پرس‌وجو مرتبط‌تر است (Mohamed & Rajasekaran 2006). علاوه بر آن، خلاصه می‌تواند تنها با هدف به‌روزرسانی یک منبع اطلاعاتی تولید شود، یعنی با فرض موجودبودن مجموعه‌ای از اطلاعات، خلاصه‌ای از اطلاعات به‌روزشده در منبع اضافه شود. در نهایت، یک خلاصه می‌تواند با هدف اطلاع‌رسانی عمومی دربارهٔ یک سند یا

1. query

مجموعه‌ای از اسناد تولید شود، بدون آن‌که اطلاعات خاصی در بدو امر مورد نیاز باشد (Nenkova & McKeown 2012).

روش تهیهٔ خلاصه می‌تواند بر مبنای استخراج مجموعه‌ای از جملات اصلی باشد که در متون آمده است، یعنی عین جملات متون مرجع در متن خلاصه قید شود که به آن خلاصهٔ استخراجی می‌گوییم. در نوعی دیگر که عمدتاً چالش‌برانگیزتر است، جملات متون مرجع عیناً آورده نمی‌شود، بلکه جملات جدیدی حاوی خلاصهٔ متن به‌صورت خودکار تولید می‌شود. این حالت، خلاصهٔ چکیده‌ای خوانده می‌شود که عمدتاً روش‌های تولید زبان طبیعی[1] نیز مستقیماً در آن لحاظ خواهد شد. در خلاصهٔ چکیده‌ای متن خلاصه لزوماً تکرار بعضی از جملات متن نیست، بلکه هدف درک متن و تولید جملات جدید و موجز است. به‌عنوان مثال، در متن «ساعت حرکت قطار ۱۰ شب بود، اما مریم دیر به ایستگاه رسید و قطار حرکت کرده بود» خلاصهٔ چکیده را می‌توان به شکل زیر بازنویسی کرد: «مریم به قطار ساعت ۱۰ شب نرسید». در این نوع خلاصه، سیستم علاوه بر درک محتوای متن باید قادر به جمله‌سازی بر اساس مطالب اصلی متن باشد، به همین دلیل نسبت به خلاصه‌سازی استخراجی از پیچیدگی بیشتری برخوردار است، اما به خلاصهٔ انسانی (خلاصه‌ای که توسط یک انسان به‌صورت دستی تولید شده است) شباهت بیشتری دارد (Al-Hashemi 2010; Diola et al. 2004; Nenkova & McKeown 2012).

در نهایت، یک خلاصه می‌تواند بر اساس محتوایی که دارد، متفاوت باشد. خلاصهٔ نشانگر[2]، معمولاً اطلاعات توصیفی دربارهٔ متن نظیر حجم، شیوهٔ نگارش و موضوع اصلی آن را دربر دارد، در حالی که خلاصهٔ آگاه‌گر[3] دربارهٔ محتوای متن اطلاعاتی را ارائه می‌کند. در برخی موارد نیز نیاز است که مجموعه‌ای از کلمات کلیدی مهم یک متن یا مجموعه‌ای از متون، به‌عنوان نمایندهٔ محتوای متن استخراج شود که در این صورت خلاصه بر مبنای کلمات کلیدی حاصل می‌شود (Nenkova & McKeown 2012b).

با توجه به وجود انواع خلاصه، در این پژوهش تمرکز ما روی خلاصهٔ استخراجی از یک سند است و هدف ما اطلاع‌رسانی عمومی و تولید متن خلاصه‌ای است که محتوای متن را به‌صورت موجز منعکس کند. در نهایت، ساختار محتوای خلاصه‌ای که تولید

1. natural language generation
2. indicative
3. informative

می‌شود، یک متن آگاه‌گر خواهد بود. بنابراین، در بخش بعدی، مراحل اصلی برای خلاصه‌سازی استخراجی ارائه می‌شود.

۲-۲. مراحل اصلی خلاصه‌سازی استخراجی

در هر روشِ خلاصه‌سازی استخراجی سه مرحلهٔ پیش‌پردازش، پردازش و انتخاب جملات وجود دارد (Nenkova & McKeown 2012). در مرحلهٔ پیش‌پردازش، روی متن ورودی، عملیات پردازش اولیهٔ متن و نرمال‌سازی آن انجام می‌شود تا برای مراحل بعدی آماده شود. در این مرحله عملگرهای پردازش زبان طبیعی نظیر برچسب‌زنی اجزای گفتار[1]، استخراج کلمات کلیدی، و ریشه‌یابی[2] اعمال می‌شود. در مرحلهٔ پردازش با استفاده از مفاهیم آماری، زبان‌شناختی یا ترکیب هر دوی آن‌ها، جملات موجود در متن ورودی امتیازدهی می‌گردد. در مرحلهٔ انتخاب جملات، بر اساس معیاری که بر مبنای نیاز کاربر در سیستم تعبیه شده، جملات مناسب انتخاب می‌گردند و در نهایت، جملات مرتب شده و نتیجه به‌عنوان خروجی سیستم خلاصه‌ساز در اختیار کاربر قرار می‌گیرد.

مرحلهٔ پیش‌پردازش شامل استخراج جملات و کلمات، تعیین نقش کلمات در جمله (برچسب‌زنی کلمات)، حذف کلمات توقف، و ریشه‌یابی است. در ادامه، با توجه به این‌که بخشی از نوآوری این پژوهش روی ارتقاء مرحلهٔ پیش‌پردازش است، برخی از مفاهیم این مرحله تشریح می‌گردد.

۲-۲-۱. برچسب‌زنی کلمات

یکی از کارهای اساسی در پردازش زبان طبیعی، برچسب‌زنی اجزای گفتار است. برچسب‌زنی، تعیین مقوله‌های دستوری برای هر نماد در متن است. در برچسب‌زنی از حوزهٔ ساخت واژه و نحو زبان برای تعیین مقوله‌های دستوری استفاده می‌شود. برچسب‌زنی اجزای کلام در واقع، به معنای انتخاب مناسب‌ترین مقولهٔ دستوری به هر کلمه یا ابهام‌زدایی از برچسب کلمه است. بسیاری از واژه‌ها در متون بیش از یک برچسب دستوری دارند، به‌عنوان مثال، کلمهٔ «تند» در زبان فارسی می‌تواند صفت یا قید باشد (Azimizadeh, Arab, & Quchani 2008).

روش‌های برچسب‌زنی دستوری را می‌توان به سه رویکرد اصلی تقسیم نمود:

1. part of speech (POS) tagging
2. stemming

رویکرد مبتنی بر قواعد، رویکرد آماری مبتنی بر مدل‌های مارکوف، و رویکرد مبتنی بر ماکزیمم آنتروپی[1] (Güngör 2010).

رویکرد مبتنی بر قواعد: در این رویکرد، مجموعه‌ای از قواعد برای برچسب‌گذاری واژگان استخراج می‌شود. این مجموعه قواعد می‌تواند به‌صورت دستی بر مبنای قواعد دستوری و زبانی ایجاد شود، یا بر اساس روش‌های یادگیری ماشینی استخراج گردد. در روش استخراج دستی لازم است که واژه‌نامه‌ای که قواعد دستوری بر اساس آن‌ها تهیه می‌شود، کامل بوده و به‌طور دائم پویا شود، زیرا اگر کلمه‌ای در واژه‌نامه نباشد قادر به برچسب‌گذاری آن نخواهیم بود. روش‌های یادگیری ماشینی می‌تواند محدودیت‌های روش‌های استخراج دستی قوانین را تا حد زیادی برطرف سازد. در روش یادگیری مبتنی بر تغییر[2]، که اولین‌بار توسط «بریل» ارائه شد، قواعد برچسب‌گذاری با استفاده از الگوریتم‌های یادگیری ایجاد می‌شود. در این روش از مجموعهٔ دادهٔ آموزشی متشکل از یک پیکره به همراه الگوهای برچسب‌گذاری واژگان آن برای یادگیری قواعد برچسب‌گذاری استفاده می‌شود (Brill 1995). ابتدا بر اساس یک‌سری قواعد اولیه متن دادهٔ آموزشی برچسب‌گذاری شده، سپس حین فرایند یادگیری و با توجه به الگوها و قواعد موجود در داده‌های آموزشی، به‌تدریج قواعد اولیه تغییر پیدا می‌کنند یا حتی قواعد جدیدی ایجاد می‌شوند (Güngör 2010). خروجی این روش مجموعه‌ای از قواعد برچسب‌گذاری واژگان در متن است. در روش یادگیری مبتنی بر تغییر معمولاً زمان محاسباتی زیادی برای پردازش و یادگیری قواعد نیاز است.

رویکرد آماری مبتنی بر مدل‌های مارکوف: با توجه به امکان دسترسی به حجم زیادی از پیکره‌های زبانی مختلف، روش‌های آماری نتایج اثربخشی را در زمینهٔ برچسب‌گذاری واژگان ارائه کرده‌اند و بسیاری معتقد هستند که این روش‌ها می‌توانند نتایج بهتری نسبت به رویکرد مبتنی بر قواعد ارائه کنند (Güngör 2010). این روش‌ها معمولاً بر مبنای مدل‌های «مارکوف»[3] شکل گرفته‌اند و از بین این مدل‌ها، «الگوی پنهان مارکوف»[4] بیشترین استفاده را دارد. در برچسب‌گذاری با «الگوی پنهان مارکوف» دنبالهٔ برچسب‌ها در

1. maximum entropy
2. transformation-based learning
3. Markov
4. Hidden Markov Model (HMM)

یک متن را به‌عنوان یک زنجیرهٔ «مارکوف» در نظر می‌گیرند (Wicaksono & Purwarianti 2010). در این میان، یکی از روش‌هایی که بر روی زبان فارسی موفق بوده، روش HunPOS است، که بر اساس پیاده‌سازی دوبارهٔ روش TNT[1] ساخته شده و قابلیت تنظیمات مختلف برای زبان‌های گوناگون را به کاربر می‌دهد و برای زبان فارسی ۹۶/۹ درصد درست عمل می‌کند (Seraji 2011). در روش TNT از مدل «مارکوف درجه دو» استفاده می‌شود (Brants 2000).

رویکرد ماکزیمم آنتروپی: در مدل‌های «مارکوف» برای تعیین برچسب یک واژه، وابستگی آن به برچسب سایر واژگان در جمله در نظر گرفته نمی‌شود. در رویکرد ماکزیمم آنتروپی که توسط «راتناپارکی»[2] در سال ۱۹۹۶ مطرح شد، این وابستگی تا حدی لحاظ می‌شود. در این رویکرد امکان لحاظ کردن مجموعه‌ای از ویژگی‌ها که بیانگر زمینهٔ[3] متن نیز هستند و وابستگی برچسب یک واژه به سایر واژگان در جمله را نشان می‌دهند، در یک مدل احتمالی فراهم می‌شود (Güngör 2010).

۲-۲-۲. ریشه‌یابی فارسی

از جمله کارهای انجام‌شده در زمینهٔ ریشه‌یابی کلمات فارسی می‌توان به ریشه‌یاب «بُن» (Tashakori et al. 2002) اشاره کرد. این ریشه‌یاب شبیه ریشه‌یاب «پورتر»[4] عمل کرده و بر اساس قواعدی اقدام به حذف پسوندها و پیشوندها می‌کند.

ریشه‌یاب دیگری که در زبان فارسی نتایج قابل قبولی ارائه کرده، ریشه‌یاب تهیه‌شده توسط Taghva, Beckley & Sadeh (2005) است. این الگوریتم نیز شبیه ریشه‌یاب «پورتر» عمل می‌کند، اما تفاوت‌هایی نیز دارد. برای مثال، الگوریتم ریشه‌یاب «پورتر» به‌منظور تخمین محتوای اطلاعات، الگوی حروف صدادار و بی‌صدا را تشخیص می‌دهد. اما در فارسی بسیاری از حروف صدادار نوشته نمی‌شوند، بنابراین ریشه‌یاب فارسی از طول رشته برای تعریف کران پایین محتوای ریشه استفاده می‌کند که در حال حاضر، حداقل طول ریشه ۳ حرف است. این محدودیت در بعضی موارد باعث خطا می‌گردد؛ به‌ویژه زمانی که یک زیررشته که قسمتی از یک کلمهٔ کوتاه است، به اشتباه به‌عنوان

1. Trigrams 'n' Tags
2. Ratnaparkhi
3. context
4. Porter

یک پسوند در نظر گرفته شود. تفاوت دیگر این دو الگوریتم آن است که این الگوریتم بر خلاف الگوریتم «پورتر» پیشوندها را هم شناسایی می‌کند.

۲-۳. چالش‌های مرحلهٔ پیش‌پردازش در زبان فارسی

ویژگی‌هایی که زبان فارسی در بخش نگارش کلمات دارد، مرحلهٔ پیش‌پردازش را در همهٔ کاربردهای پردازش زبان طبیعی مانند دسته‌بندی و خلاصه‌سازی با چالش مواجه می‌کند. عمدهٔ ویژگی‌های نگارشی به رسم‌الخط زبان فارسی برمی‌گردد که برخی از آن‌ها عبارت‌اند از:

- ◇ صورت‌های مختلف نوشتاری برای بعضی از حروف مانند «ی» و «ک»؛
- ◇ استفاده از «ا» و «آ» به جای هم؛
- ◇ عدم رعایت فاصله‌گذاری‌ها (استفاده از فاصله به جای نیم‌فاصله)؛
- ◇ استفاده از کلمات انگلیسی هم به‌صورت اصل کلمه و هم به‌صورت برگردان فارسی مانند کلمه «گوگل» و «google»؛
- ◇ صورت‌های مختلف نوشتاری برای پیشوندهای افعالی و اسمی مانند «آن‌ها» و «آنان» یا «می‌توانم» و «می‌توانیم»؛
- ◇ تنوع نوشتار کلماتی که به «ه» ختم می‌شوند، مثل «مجموعه اسناد» و «مجموعهٔ اسناد»؛
- ◇ تنوع نوشتاری در برخی از کلمات مانند «اتاق» و «اطاق».

چالش‌های معنایی اغلب در همهٔ زبان‌های رایج دنیا وجود دارد. زبان فارسی نیز از این قاعده مستثنی نیست. از جملهٔ این مشکلات می‌توان به ابهام معنایی کلمات اشاره کرد. به‌عنوان مثال، کلمهٔ «شیر» دارای سه معنای متفاوت «شیر جنگل»، «شیر خوراکی» و «شیر آب» است. یا کلمهٔ «عکس» دارای دو معنای «تصویر» و «خلاف» است. برخی از ابهام‌های معنایی به‌دلیل نبودِ اطلاعات آوایی است. به‌عنوان مثال، کلمه «مُرد» و کلمهٔ «مَرد» به یک شکل «مرد» نوشته می‌شود.

از آنجا که زبان فارسی از معدود زبان‌هایی است که در آن حروف تشکیل‌دهندهٔ کلمات به هم چسبیده است، همین مسئله خود باعث پیچیدگی‌هایی در پردازش زبان فارسی می‌شود. به‌تبع آن خلاصه‌سازی متون فارسی به‌دلیل ویژگی‌های نگارشی و گرامری خاص این زبان پیچیده‌تر از سایر زبان‌هاست. به‌طور خاص، در زبان فارسی استثناهای فراوانی وجود دارد که برخی از آن‌ها عبارت‌اند از:

- ◇ ابهام در قواعد تولید بن مضارع مانند «گفت» از بن «گو»؛
- ◇ وجود افعال مرکب مانند «برداشتن»؛
- ◇ متصل‌شدن بعضی افعال با اسم مانند «بیدارند» به جای «بیدار هستند».

۳. مروری بر پیشینهٔ پژوهش

ابتدایی‌ترین کارها در زمینهٔ خلاصه‌سازی، توسط «لون» و «دانینگ» انجام شده که در آن از آستانهٔ فراوانی برای شناسایی کلمات کلیدی استفاده شده و در عین حال، کلمات پرتکرار نیز که عموماً حروف ربط و اضافه هستند، حذف می‌شوند (Luhn 1957; Dunning 1993). در زمینهٔ خلاصه‌ساز از یک متن، بخش بزرگی از کارهای انجام‌شده توسط «بارزیلی و الحداد» بر روی استفاده از زنجیره‌های لغوی (Taghva, Beckley & Sadeh 2005) به‌منظور ساخت زنجیره‌های کارآمد و رفع ابهام از کلمات برای خلاصه‌سازی بوده است (Barzilay & Elhadad 1999). در این روش از مشابهت معنایی لغات برای استخراج جملات بااهمیت استفاده می‌شود (Guo-shun 2011; Nadkarni, Ohno-Machado & Chapman 2011). به‌طور مثال، تعداد رخداد کلماتی مثل "car"، "wheel"، "seat" و "passenger" با هم در نظر گرفته می‌شود، اگرچه به تنهایی پرتکرار نیستند. این رویکرد کاملاً به شبکهٔ واژگان WordNet متکی است (Song, Han, & Rim 2004; Miller et al. 1990) که یک مجموعهٔ دستی جمع‌آوری شده است و هر کلمه را به همراه مترادف‌ها و متضادهای آن نشان می‌دهد.

در زمینهٔ خلاصه‌ساز استخراجی، سیستم‌های متنوعی برای خلاصه‌سازی در زبان انگلیسی و سایر زبان‌ها ارائه‌شده که مهم‌ترین آن‌ها سیستم SumBasic (Vanderwende et al. 2007)، خلاصه‌ساز SweSum (Dalianis 2000)، و خلاصه‌ساز MEAD (Radev et al. 2006) است. در سیستم SumBasic، ابتدا برای کلمات یک تابع توزیع احتمال بر اساس فراوانی آن‌ها در مجموعهٔ داده در نظر گرفته می‌شود. سپس، برای هر جمله یک امتیاز محاسبه می‌شود که این امتیاز حاصل میانگین احتمال کلمات آن جمله است. سیستم SweSum، توسط «مؤسسهٔ فناوری رویال سوئد» طراحی شده که بر اساس اطلاعات آماری و زبان‌شناختی متن، خلاصه را استخراج می‌کند. این سیستم در حال حاضر زبان‌های انگلیسی، دانمارکی، نروژی، سوئدی و فارسی را پشتیبانی می‌کند. خلاصه‌ساز MEAD سیستمی است که در آن از چند روش خلاصه‌سازی استفاده شده است و در حال حاضر زبان‌های انگلیسی و چینی را پشتیبانی می‌کند (Radev et al. 2004).

در زمینهٔ خلاصه‌ساز برای زبان فارسی پژوهش‌های متعددی انجام شده است. مهم‌ترین آن‌ها، سیستم FarsiSum، نسخهٔ فارسی سیستم SweSum است که بر اساس الگوریتم اصلی سیستم SweSum ایجاد شده است (Hassel & Mazdak 2004). این سیستم به‌طور خاص برای زبان فارسی تعبیه نشده و ویژگی‌های زبانیِ زبان فارسی در طراحی آن لحاظ نشده است. در راستای لحاظ کردن معانی واژگان در سیستم خلاص‌ساز استخراجی، «رمضانی و فیضی درخشی» یک روش مبتنی بر هستان‌شناسی را بر اساس مجموعهٔ واژگان فارس‌نت[1] برای خلاصه‌سازی فارسی ارائه کرده‌اند (Ramezani & Feizi-Derakhshi 2015).

اکثر سیستم‌های خلاصه‌ساز متون فارسی که به‌صورت تجاری یا تحقیقاتی موجود هستند، بر مبنای روش‌های استخراج کلمات پرتکرار و حذف کلمات توقف ایجاد شده‌اند. ساختار زبان فارسی و چالش‌های آن در این سیستم‌ها کمتر مورد توجه قرار گرفته است. به‌عنوان مثال، در سیستم خلاصه‌ساز «ایجاز» سعی شده که علاوه بر ویژگی‌های مبتنی بر فراوانی کلمات، از ویژگی‌های دیگری نظیر میزان شباهت با زمینه، تأثیر کلمات توقف، تأثیر طول جمله، موقعیت جمله در متن، برای انتخاب جملات برتر استفاده شود (پورمعصومی و همکاران ۱۳۹۳). در این پژوهش از روش‌های مبتنی بر گراف برای استخراج خلاصهٔ چند سند استفاده می‌شود. روش‌های مبتنی بر گراف، که نتایج خوبی را در خلاصه‌سازی ارائه نموده‌اند، از الگوریتم‌های PageRank الهام گرفته‌اند که در آن متن ورودی در قالب یک گراف نمایش داده می‌شود (Erkan & Radev 2004; Mihalcea Rada 2004). گره‌ها یا رئوس گراف نشان‌دهندهٔ جملات هستند و وزن یال بین دو جمله بیانگر میزان شباهت دو جمله است. برای زبان فارسی نیز پژوهش‌هایی برای به‌کارگیری روش‌های مبتنی بر گراف ارائه شده، مانند پژوهش شاکری و همکاران (۱۳۹۰)، زمانی‌فر و همکاران (۲۰۰۸) (Zamanifar et al. 2008)، وکریمی و شمس‌فر (۱۳۸۵). هر یک از این پژوهش‌ها روش‌های کارآمدی را برای استخراج خلاصه پیشنهاد کرده‌اند، لیکن در هیچ‌یک از آن‌ها بر نقش دستوری کلمات به‌عنوان یک ویژگی برای انتخاب جملات برتر متن مرجع تأکید نشده است.

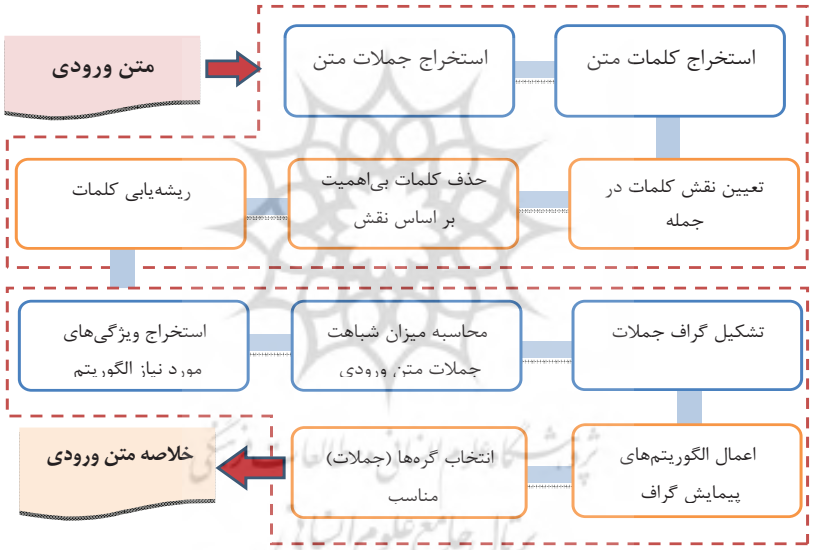
در این پژوهش روشی ارائه می‌شود که از طریق آن می‌توان نقش دستوری کلمات را نیز در روش‌های مبتنی بر گراف لحاظ نمود. علاوه بر آن یکی از چالش‌های اصلی

1. FarsNet

که همچنان در روش‌های مبتنی بر گراف وجود دارد، کاهش افزونگی اطلاعات در جملات انتخاب شده است، به این مفهوم که جملاتی انتخاب شوند که در ضمن این‌که اطلاعات اصلی متن مرجع را دربر دارند، اما کمترین اشتراک اطلاعاتی را با یکدیگر داشته باشند. در این پژوهش رویکردی ارائه می‌شود که این چالش تا حدی مرتفع گردد.

٤. روش پژوهش

در این پژوهش از روش مبتنی بر گراف استفاده شده است. بنابراین، جملات، رئوس گراف را تشکیل می‌دهند و یال بین رئوس نیز میزان شباهت بین جملات است. شکل ۲، معماری روش پیشنهادی را نشان می‌دهد.



شکل ۲. معماری روش پیشنهادی جهت خلاصه‌سازی استخراجی

مراحل نشان داده‌شده در شکل را می‌توان به دو بخش کلی تقسیم نمود:

- مرحلهٔ پیش‌پردازش: شامل مراحل استخراج جملات و کلمات، تعیین نقش کلمات در جمله، حذف کلمات توقف و ریشه‌یابی کلمات؛
- مرحلهٔ پردازش: شامل مراحل استخراج ویژگی‌های کلمات هر متن، محاسبهٔ میزان شباهت همهٔ جملات یک متن به هم، ساخت گراف برای هر متن ورودی، اعمال.

الگوریتم‌های پیمایش گراف و در نهایت، انتخاب گره‌های (جملات) مهم و نمایش آن در خروجی.

در ادامه، مراحل ساختار پیشنهادی به تفکیک گام‌های آن شرح داده می‌شود.

٤-١. مرحلهٔ پیش‌پردازش

در این مرحله لازم است داده‌های مسئله، که همان متن ورودی است، به شکلی که در مرحلهٔ پردازش نیاز است تبدیل شود. بر اساس روش پیشنهادی، در مرحلهٔ پردازش برای هر متن ورودی باید نقش هر کلمه در جمله و ریشهٔ آن نیز موجود باشد. بنابراین، در مرحلهٔ پیش‌پردازش باید این مراحل انجام شود: تعیین نقش کلمات در جمله، حذف کلمات توقف بر اساس نقش آن‌ها، و ریشه‌یابی کلمات.

٤-١-١. تعیین نقش کلمات در جمله

برای تعیین نقش دستوری کلمات در جمله (مانند اسم، صفت، فعل)، از الگوریتم HunPos که برای زبان فارسی دارای دقت بالایی است، استفاده شده است. این الگوریتم نسخهٔ اصلاح‌شدهٔ الگوریتم «تی‌ان‌تی» است که در زبان انگلیسی از آن استفاده می‌شود (Brants 2000).

٤-١-٢. حذف کلمات توقف بر اساس نقش آن‌ها

کلمات توقف[1] (یا سیاههٔ غیرمجاز) کلماتی هستند که در جملات مسئولیت ارتباط اجزای مختلف جمله را بر عهده دارند و به همین دلیل، بسیار مورد استفاده قرار می‌گیرند، اما فاقد بار معنایی بوده و عموماً در فعالیت‌هایی که در حوزهٔ پردازش زبان طبیعی انجام می‌شود، در مرحلهٔ پیش‌پردازش حذف می‌شوند. معمولاً برای حذف کلمات توقف از فهرستی از کلمات توقف که از پیش تهیه‌شده، استفاده می‌شود.

در زبان فارسی نیز فهرستی از کلمات توقف وجود دارد، مانند: «چون، بالاخره، با، من، ایشان، بله، یا، برای، ...»، اما این فهرست کامل نیست. به همین دلیل در این مقاله کلمات توقف بر اساس نقش آن‌ها در جمله شناسایی و حذف می‌شوند. در جدول ۱، نمونه‌ای از کلمات توقف شناسایی‌شده با استفاده از نقش دستوری آن‌ها آمده است.

1. stop list

جدول ۱. نمونه‌ای از کلمات توقف[1]

نمونهٔ کلمات	برچسب دستوری
از، در، به، برای	حرف اضافه (PREP)[1]
و، اما، یا	حرف ربط (CONJ)[2]
خود، من، ما	ضمیر (PRO)[3]

٤-١-٣. ریشه‌یابی کلمات

در این پژوهش برای ریشه‌یابی کلمات از الگوریتم «کراوتز»[1] (Jivani & others 2011) استفاده شده است. این الگوریتم پسوند و پیشوند کلمات را بررسی کرده و در هر مرحله با حذف پسوند یا پیشوند یافت‌شده، ریشه به‌دست‌آمده را بررسی کرده و در صورتی که در فرهنگ لغت موجود باشد، عمل ریشه‌یابی متوقف می‌شود. در نسخهٔ اصلاح‌شده عمل ریشه‌یابی با توجه به نقش کلمه در جمله انجام می‌شود.

٤-٢. مرحلهٔ پردازش

خروجی مرحلهٔ پیش‌پردازش در نهایت، یک جدول حاوی این اطلاعات است: شمارهٔ متن، شمارهٔ جمله، شمارهٔ کلمه، کلمه، نقش دستوری کلمه در جمله و ریشهٔ کلمه. علاوه بر آن، در کنار هر متن موجود در مجموعهٔ ارزیابی، یک عنوان برای متن نیز وجود دارد. تمام عملیاتی که در مرحلهٔ پیش‌پردازش برای جملات داخل متن انجام شده برای عنوان هر متن نیز انجام می‌شود.

٤-٢-١. استخراج ویژگی‌ها

ویژگی‌های استخراج‌شده برای هر متن موارد زیر است:

◇ کلمات کلیدی هر متن: برای استخراج کلمات کلیدی از معیار TF.IDF[4] استفاده شده است. بدین شکل که برای هر متن داده‌شده از میان کلماتی که در انتهای مرحلهٔ پیش‌پردازش به‌دست آمده، ۵ درصد آن‌هایی که دارای بیشترین مقدار TF.IDF هستند،

1. 1. preposition
2. conjunction
3. pronoun
4. Term Frequency .Inverse Document Frequency

به‌عنوان کلمات کلیدی متن استخراج می‌شود. مقدار TF.IDF هر کلمه به شکل زیر محاسبه می‌گردد:

$$TF_{w,d} = d \text{ تعداد تکرار کلمه } w \text{ در متن} \quad (۱)$$

$$IDF_w = log\frac{N}{n_w} \quad (۲)$$

که در آن N تعداد کل اسناد و n_w تعداد اسنادی که شامل کلمهٔ w است.

$$TF.IDF_{w,d} = TF_{w,d} \times IDF_w \quad (۳)$$

◇ ضریب تأثیر نقش‌های دستوری مختلف: بدیهی است که کلمات متفاوت در یک متن ضرایب تأثیر متفاوتی در رساندن مفهوم کل متن دارند. همان‌طور که در مرحلهٔ پیش‌پردازش کلمات توقف حذف شدند، در واقع ضریب تأثیر نقش‌های دستوری کلمات توقف صفر در نظر گرفته شده است. به همین صورت، می‌توان برای سایر نقش‌های دستوری نیز ضرایب تأثیر آن‌ها را بر اساس داده های آموزشی محاسبه کرد.

برای این منظور فرض کنید که D مجموعهٔ کل متون و D_i مجموعه متونی است که در کلمات کلیدی خود نقش دستوری Tag_i را دارند. به‌عبارت دیگر:

$$D_i = \{d | d \in D, \exists w \in Keywords(d): Tag(w) = Tag_i\} \quad (۴)$$

در تعریف بالا فرض کنید Keywords(d) مجموعهٔ کلمات کلیدی متن d است. همچنین فرض کنید K_i مجموعهٔ کلمات کلیدی با برچسب Tag_i از مجموعهٔ متون D_i باشد:

$$K_i = \{w | \forall d_i \in D_i : w \in Keywords(d_i) \wedge Tag(w) = Tag_i\} \quad (۵)$$

و در نهایت، ضریب تأثیر هر یک از نقش‌های دستوری از رابطهٔ زیر محاسبه می‌شود:

$$Coeff\ (tag_i) = \frac{|K_i|}{|\cup_{d_i \in D_i} Keywords(d_i)|} \quad (۶)$$

به این ترتیب، برای تمام برچسب‌های دستوری می‌توان ضریب تأثیر آن را از طریق مجموعهٔ دادهٔ آموزشی محاسبه کرد. این ضرایب یک بار محاسبه و در طول اجرای کل برنامه از آن‌ها استفاده می‌شود.

٤-۲-۲. ساخت بردار ویژگی برای هر جمله

لازمهٔ ساخت گراف جملات برای هر متن این است که هر جمله را به شکل

کمیت‌های عددی نمایش دهیم. برای این منظور، از چهار ویژگی برای هر جمله استفاده شده که عبارت‌اند از:

◇ میزان شباهت هر جمله با عنوان متن:

$$F1 = Sim(S_i, Title) = \frac{\sum\{TF.IDF\ (w_j)\times(Avg(Coeff(Tag(w_j\ in\ S_i), Coeff\left(Tag(w_j\ in\ Title)\right)|w_j \in S_i \wedge w_j \in Title\}}{\sum\{TF.IDF\ (w_j)\times(Avg(Coeff(Tag(w_j\ in\ S_i), Coeff\left(Tag(w_j\ in\ Title)\right)|w_j \in S_i \vee w_j \in Title\}} \quad (۷)$$

◇ میزان شباهت هر جمله با کلمات کلیدی:

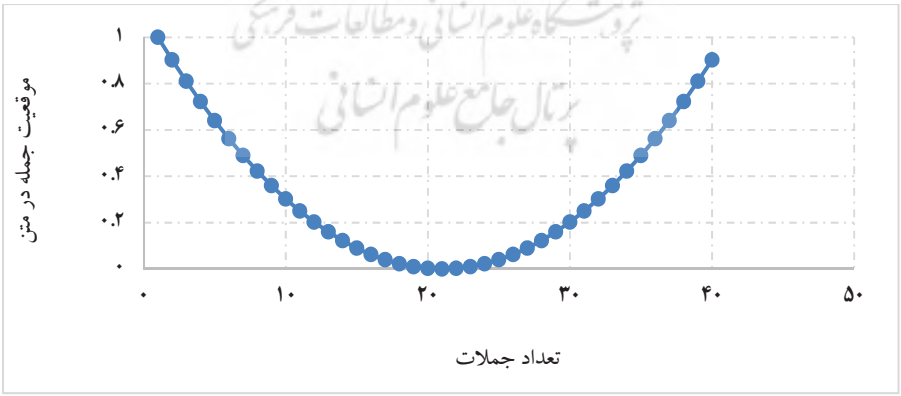
$$F2 = Sim(S_i, Keywords\ ) = \frac{\sum\{TF.IDF\ (w_j)\times(Avg(Coeff(Tag(w_j\ in\ S_i), Coeff\left(Tag(w_j\ in\ Keywords\ )\right)|w_j \in S_i \wedge w_j \in Keywords\ \}}{\sum\{TF.IDF\ (w_j)\times(Avg(Coeff(Tag(w_j\ in\ S_i), Coeff\left(Tag(w_j\ in\ Keywords\ )\right)|w_j \in S_i \vee w_j \in Keywords\ \}} \quad (۸)$$

◇ میزان شباهت دو جمله با هم:

$$F3 = Sim(S_i, S_k) = \frac{\sum\{TF.IDF\ (w_j)\times(Avg(Coeff(Tag(w_j\ in\ S_i), Coeff\left(Tag(w_j\ in\ S_k)\right)|w_j \in S_i \wedge w_j \in S_k\}}{\sum\{TF.IDF\ (w_j)\times(Avg(Coeff(Tag(w_j\ in\ S_i), Coeff\left(Tag(w_j\ in\ S_k)\right)|w_j \in S_i\ \}} \quad (۹)$$

$$F4 = Position(S_i) = \left(\frac{Index(S_i) - (1/2)\times N}{(1/2)\times N}\right)^2 \quad (۱۰)$$

که در آن Index(S_i) نشان‌دهندهٔ این است که جملهٔ S_i، جملهٔ چندم متن ورودی است. جملهٔ اول دارای موقعیت صفر و جملهٔ آخر دارای موقعیت N-1 است. نمودار این تابع برای یک متن با ۴۰ جمله در شکل ۳، رسم شده است. البته، شکل کلی تابع به ازاء هر تعداد جمله نیز به همین شکل است.



شکل ۳. تابع امتیاز موقعیت جمله

در غالب متن‌ها، به‌ویژه در متن‌های خبری، جملات اول، سرخط خبر، و جملات پایانی، نتیجه‌گیری از متن بوده و بیشتر از سایر جملات در خلاصه‌ها ظاهر می‌شود. به همین دلیل، تابع موقعیت جمله طوری در نظر گرفته شده است که به جملات ابتدایی و انتهایی امتیاز بالاتری می‌دهد.

۴-۱-۱. محاسبهٔ میزان شباهت جملات بر اساس بردار ویژگی‌های جمله

پس از محاسبهٔ چهار ویژگی ذکرشده برای هر جمله لازم است میزان شباهت دو جمله از یک متن نسبت به هم محاسبه شود. برای این منظور از ضرب داخلی بردار ویژگی دو جمله، میزان شباهت هر دو جمله به‌دست می‌آید:

$$\forall i,j = 1,2,\dots,N \; i \neq j \quad S_i = (F_{1i}, F_{2i}, F_{3i}, F_{4i})^T \Rightarrow$$
$$Similarity(S_i, S_j) = \overrightarrow{S_i}.\overrightarrow{S_j} = F_{1i} \times F_{1j} + F_{2i} \times F_{2j} + F_{3i} \times F_{3j} + F_{4i} \times F_{4j} \qquad (۱۱)$$

۴-۳. تشکیل گراف جملات

بعد از محاسبهٔ میزان شباهت هر دو جملهٔ یک سند با هم، گراف مشابه ترسیم می‌شود. در این گراف گره‌ها (رئوس) جملات متن ورودی هستند و یال (ارتباط بین دو گره) بیانگر میزان شباهت بین دو گره (جمله) آن است. گراف حاصل یک گراف کامل، همبند و غیرجهت‌دار است که به‌صورت زیر تعریف می‌شود:

$$G = (V, E) \;:\; V = \{S_1, S_1, \dots, S_N\} \; E = \{(S_1, S_2), (S_1, S_3), \dots, (S_2, S_3), \dots\} \rightarrow$$
$$|V| = N \quad , \quad |E| = \frac{N(N-1)}{2} \qquad (۱۲)$$

۴-۳-۱. روش پیشنهادی برای پیمایش گراف بر اساس بیشترین میزان شباهت بیرونی و کمترین شباهت درونی

یک روش سادهٔ پیمایش گراف به این شکل است که برای هر گره یا جمله جمع یال‌های خروجی آن محاسبه و سپس گره‌ها را به ترتیبِ نزولی این مجموع مرتب کرده و به تعدادی که بر اساس نرخ فشردگی[1] (با r نشان می‌دهیم و به‌عنوان ورودی از جانب کاربر تعیین می‌شود) جملات بالای لیست انتخاب گردند. این روش در نتایج به نام Simple Graph ذکر شده است. در این روش جملات انتخاب‌شده ممکن است به هم شباهت زیادی داشته باشند و در نتیجه، خلاصه دارای محتوای تکراری بوده و اطلاعات

1. compression rate

کاملی از متن را به ما ندهد. برای حل این مشکل الگوریتم را طوری تغییر می‌دهیم که بعد از انتخاب یک جمله، جملات بعدی به‌گونه‌ای انتخاب شوند که شباهت کمتری به جملات انتخاب‌شدهٔ قبلی داشته باشند. به‌عبارت دیگر، بعد از انتخاب یک گره ارتباط آن گره با سایر گره‌ها به شکل زیر تضعیف می‌شود:

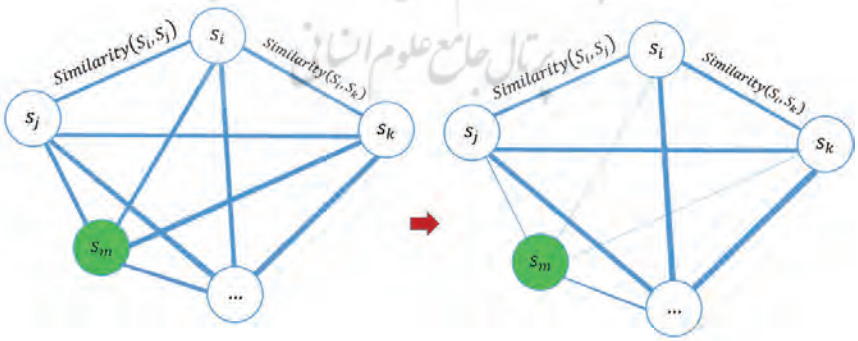
$$if\ S_i \in S \rightarrow \forall j |\ S_j \in V - S$$

$$Similarity(S_i, S_j) = \overrightarrow{S_i}.\overrightarrow{S_j} - F_{3i} \times F_{3j} = F_{1i} \times F_{1j} + F_{2i} \times F_{2j} + F_{4i} \times F_{4j} \quad (۱۳)$$

یعنی ویژگی میزان شباهت دو جمله از بردار شباهت حذف می‌شود. به همین دلیل، شباهت این جمله با سایر جملات باعث انتخاب جملات بعدی نخواهد شد. در نتیجه، جملاتی انتخاب خواهند شد که نه‌تنها شبیه به هم نیستند، بلکه اطلاعات محتوایی بیشتری دارند.

همان‌طور که در شکل ۴، مشاهده می‌شود، بعد از انتخاب گره S_m تمامی اتصالات این گره (تمام یال‌های متصل به گره تضعیف شده‌اند؛ یعنی از مقدار شباهت آن‌ها کاسته شده است. شبه‌کد[1] این روش به شکل زیر است:

$S = \emptyset$
$While\ |S| \leq (1 - r) \times N$
{
$\forall S_i \in V - S \quad Weight_i = \sum_j Similarity(S_i, S_j) \quad j = 1,2, \dots, |V|\ \ i \neq j$
$S = S \cup \{S_i | Weight_i = max_{\forall j: S_j \in V - S} Weight_j\}$
$\forall S_j \in V - S \quad Similarity(S_i, S_j) = F_{1i} \times F_{1j} + F_{2i} \times F_{2j} + F_{4i} \times F_{4j}$
}



شکل ٤. تأثیر انتخاب گره در اتصالات آن گره به‌صورت شماتیک

1. Pseudo code

۵. پیاده‌سازی و ارزیابی روش پیشنهادی

۵-۱. روش پیشنهادی بر روی پیکرهٔ استاندارد «پاسخ» پیاده‌سازی شده که برای ارزیابی خلاصه‌سازی خودکار در زبان فارسی به‌کار می‌رود و توسط آزمایشگاه فناوری وب «دانشگاه فردوسی مشهد» و با همکاری «سازمان فناوری اطلاعات ایران» تولید گردیده است (Moghaddas et al. 2013). این پیکره مشتمل بر دو مجموعهٔ تک‌سندی و چندسندی است که در این مقاله تنها از پیکره تک‌سندی استفاده شده است. پیکرهٔ تک‌سندی ۱۰۰ موضوع مختلف از انواع گونه‌های خبری را شامل می‌شود. جدول ۲، شامل اطلاعاتی در خصوص پیکرهٔ مورد استفاده است.

جدول ۲. مشخصات تک‌سندی پیکرهٔ مورد استفاده

ویژگی	مقدار
تعداد کل خبرها (موضوعات خبری)	۱۰۰
تعداد کل جملات	۴۳۵۴
تعداد کل کلمات	۷۲۳۱۸
تعداد کلمات توقف	۲۵۷۹۴

پس از تعیین نقش کلمات در جمله برچسب‌های دستوری که بار معنایی ندارند، انتخاب و کلماتی که یکی از این نقش‌ها را داراست، حذف می‌شوند و در گام آخر مرحلهٔ پیش‌پردازش کلمات بر اساس الگوریتم «کراوتز» ریشه‌یابی می‌شوند.

در اولین مرحلهٔ پردازش با محاسبهٔ ضرایب تأثیر هر یک از نقش‌های دستوری، اهمیت کلمات موجود در متن بر اساس نقش آن‌ها در جمله مشخص می‌شود که این میزان اهمیت در گام‌های بعدی در محاسبهٔ امتیاز جملات تأثیرگذار خواهد شد.

نتیجهٔ محاسبهٔ ضرایب تأثیر نقش‌های دستوری در جدول ۳، آمده است. با توجه به این جدول، «اسامی»[1] در رساندن محتوای کلمهٔ بیشتر از سایر برچسب‌های دستوری تأثیرگذار است. در پردازش‌های زبانی نیز معمولاً کلماتی که نقش اسم را دارند، بهتر

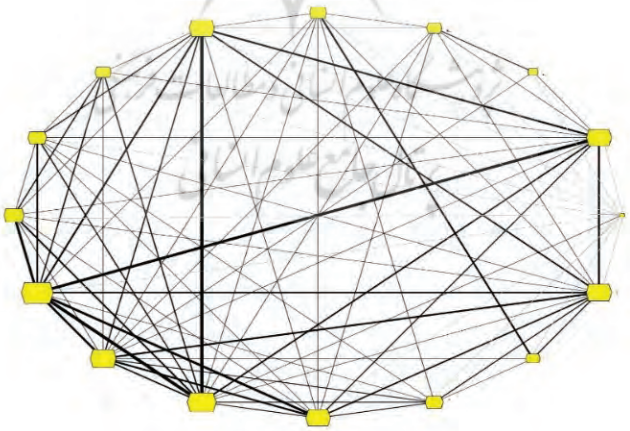
1. noun

می‌توانند بیانگر محتوای یک جمله باشند.

جدول ۳. ضریب تأثیر نقش‌های دستوری مختلف در کلمات کلیدی

برچسب	ضریب تأثیر
صفت (ADJ)	۰/۱۵۵
اسم (NOUN)	۰/۷۵۶
قید (ADV)	۰/۰۶۳
حرف تعریف(IDEN)	۰/۰۴۸
فعل (VERB)	۰/۰۸۸
عدد (NUM)	۰/۰۵۹

در گام بعدی برای هر جمله در متن ورودی چهار ویژگی میزان شباهت به عنوان، میزان شباهت به کلمات کلیدی، و میزان شباهت به جملهٔ دیگر و موقعیت جمله در متن محاسبه و برای تشکیل گراف میزان شباهت هر دو جمله با هم بر اساس ضرب داخلی بردارهای ویژگی دو جمله محاسبه و به‌عنوان وزن یال در نظر گرفته شد. شکل ۵، یک نمونهٔ گراف تولیدشده از یک متن با ۱۶ جمله است. یال‌های قوی‌تر با ضخامت بیشتر ترسیم شده است.



شکل ۵. شماتیک یک نمونهٔ گراف تولیدشده

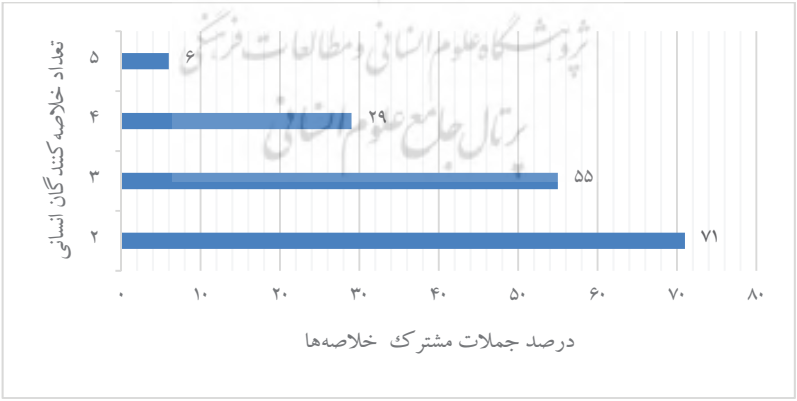
۵-۲. ارزیابی روش پیشنهادی

هر کدام از متون موجود در پیکره دارای چند خلاصه چکیده‌ای و استخراجی هستند که توسط کارشناسان تولید شده‌اند. جدول ۴، وضعیت اطلاعات خلاصه‌های انسانی را نشان می‌دهد.

جدول ٤. وضعیت خلاصه‌های انسانی پیکره

ویژگی	مقدار
تعداد کل خلاصه‌ها	۳۰۰
تعداد کل خلاصه‌کنندگان	۵
متوسط تعداد خلاصه‌ها برای هر موضوع خبری	۳

البته خلاصه‌هایی که توسط انسان تولید می‌شود، بیشتر وابسته به متن و فرد خلاصه‌کننده است و یک متنِ واحد توسط افراد مختلف لزوماً مانند هم خلاصه نمی‌شود. این مسئله را در این پژوهش به راحتی در شکل ۶، می‌توان دید. متوسط جملات مشترک بین گروه‌های مختلف خلاصه‌کننده‌ها در این شکل نشان داده شده است. هرچه تعداد خلاصه‌کننده‌ها بیشتر باشد، اشتراکات آن‌ها کمتر است. محور افقی درصد جملات مشترک و محور عمودی تعداد خلاصه‌کننده‌ها را نشان می‌دهد.



شکل ٦. میزان اشتراکات خلاصه‌های انسانی

یکی از معیارهای ارزیابی که در اغلب کاربردهای پردازش زبان طبیعی استفاده می‌شود،

معیار دقت[1] و بازخوانی[2] است. دقت برابر است با: نسبت تعداد جملات درستی که توسط سیستم خلاصه‌ساز انتخاب‌شده به کل جملاتی که سیستم برای خلاصه ارائه کرده است. بازخوانی برابر است با: نسبت تعداد جملات درستی که توسط سیستم خلاصه‌ساز انتخاب شده به کل جملاتی که توسط انسان برای خلاصه ارائه است. بنابراین، اگر S_h مجموعهٔ خلاصهٔ تولیدشدهٔ انسانی و S_c مجموعهٔ جملات انتخابی توسط سیستم خلاصه‌ساز خودکار باشد، آن‌گاه:

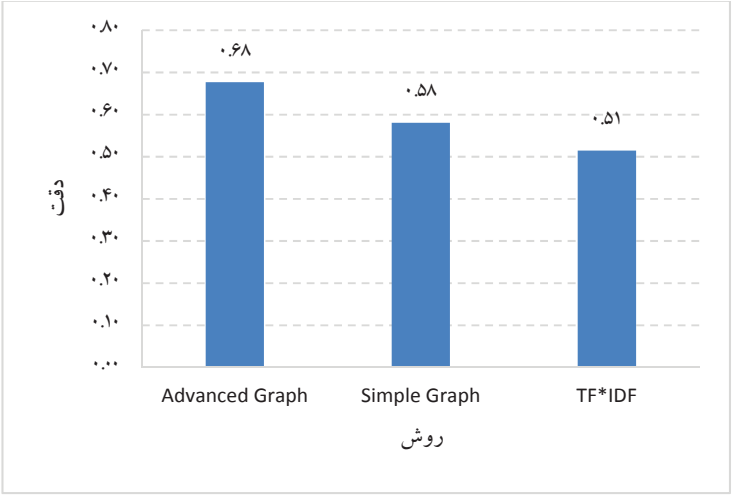
$$Precision = \frac{|S_h \cap S_c|}{|S_c|} \qquad Recall = \frac{|S_h \cap S_c|}{|S_h|} \qquad (۱۴)$$

خلاصه‌سازی خودکار روی مجموعهٔ داده‌های انتخابی بر اساس سه روش زیر اعمال شده است:

- روش TF.IDF: جمع ویژگی TF.IDF کلمات درون هر جمله تقسیم بر طول جمله و انتخاب جملات با بیشترین مقدار این ویژگی؛
- روش Simple Graph: روش مبتنی بر گراف که در آن نقش کلمات در جمله به‌عنوان ویژگی در نظر گرفته شده، ولی برای پیمایش گراف از روش ساده استفاده شده است؛
- روش Advanced Graph: روش مبتنی بر گراف که در آن نقش کلمات در جمله به‌عنوان ویژگی درنظر گرفته شده، ولی برای پیمایش گراف از روش پیشنهادی بیشترین میزان شباهت بیرونی و کمترین شباهت درونی استفاده شده است.

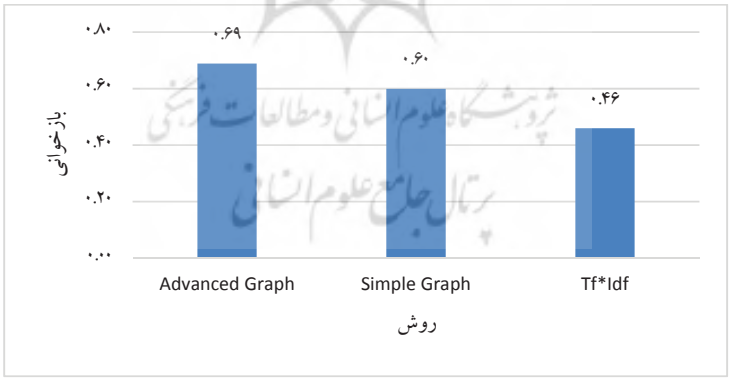
نمودار مقایسهٔ دقت حاصله از اجرای سه روش درشکل ۷، قابل مشاهده است.

1. precision
2. recall



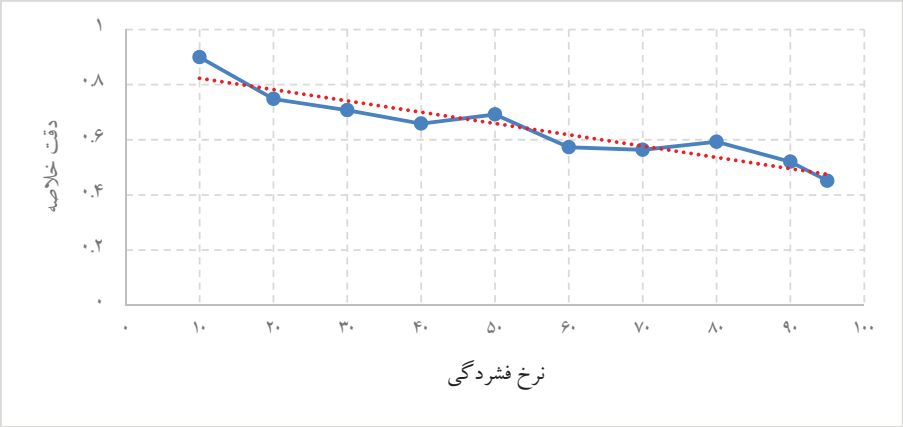
شکل ۷. مقایسهٔ معیار دقت برای روش‌های مختلف

شکل ۷ نشان می‌دهد که انتخاب ویژگی‌های عنوان‌شده در روش پیشنهادی باعث افزایش دقت خلاصه به میزان ۷ درصد شده است. همچنین، تغییر روش پیمایش گراف و تضعیف یال‌های گره انتخاب‌شده در هر مرحله باعث افزایش دقت به میزان ۱۰ درصد گردیده است. در شکل ۸ ، معیار بازخوانی روش‌های مختلف با هم مقایسه شده است.



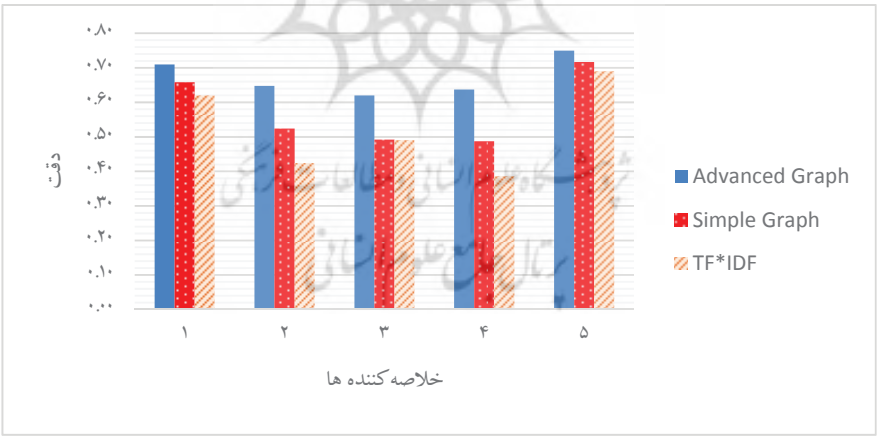
شکل ۸. مقایسهٔ معیار بازخوانی روش‌های مختلف

در شکل ۹، میزان دقت بر اساس میزان فشردگی آمده است، به‌طوری‌که هرچه حجم خلاصه بالاتر باشد (یعنی میزان فشردگی کمتر باشد)، دقت خلاصه بالاتر است که البته، این موضوع طبیعی است.



شکل ۹. رابطهٔ میزان فشردگی خلاصه و میزان دقت خلاصه

همان‌طور که در بالا اشاره شد، خلاصه‌کننده‌های انسانی یک متن واحد را مثل هم خلاصه نمی‌کنند. به همین دلیل، میزان دقت به‌دست‌آمده توسط سیستم نیز در مقایسه با خلاصه‌کننده‌های مختلف متفاوت است. این مسئله را در شکل ۱۰، نیز می‌توان مشاهده کرد.

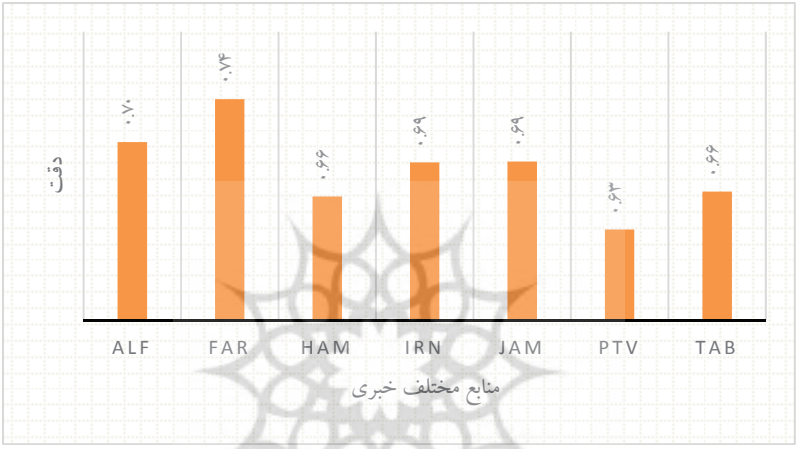


شکل ۱۰. میزان دقت خلاصه‌سازی بر اساس مقایسه با خلاصه‌های انسانی تولید شده به‌تفکیک خلاصه‌کنندگان مختلف

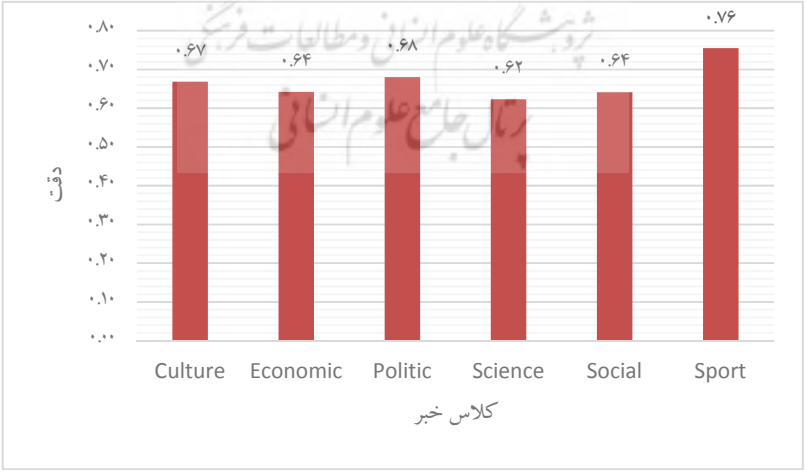
علاوه بر خلاصه‌کننده، متن ورودی و موضوع متن نیز بر کیفیت خلاصه تأثیرگذار است، چرا که منابع خبری مختلف در رعایت استانداردهای نگارشی نیز با هم متفاوت

هستند.

به همین ترتیب، انواع موضوعات نیز در کیفیت خلاصه تأثیرگذار است. به‌عنوان مثال، یک خبر ورزشی اغلب دارای جملات کوتاه و اسامی خاص بیشتری نسبت به یک خبر اجتماعی است و همین مسائل بر کیفیت خلاصهٔ سیستمی نیز تأثیرگذار است. شکل ۳، میزان دقت را به تفکیک منابع خبری متفاوت و شکل ۴، میزان دقت را به تفکیک موضوعات مختلف خبری نشان می‌دهد.



شکل ۱۱. مقایسهٔ دقت روش گراف اصلاح‌شده به تفکیک منابع مختلف خبری



شکل ۱۲. مقایسهٔ دقت روش گراف اصلاح‌شده به تفکیک موضوعات مختلف خبری

٦. نتیجه‌گیری

در این مقاله، روشی برای بهبود کیفیت خلاصه‌های استخراجی ارائه شده است. برای این منظور از ترکیب روش‌های پردازش زبان طبیعی مانند ریشه‌یابی، تعیین نقش دستوری کلمات و الگوریتم‌های مبتنی بر گراف استفاده شد. طی سه مرحلهٔ پیش‌پردازش، پردازش و انتخاب جملات سعی در تولید خلاصه از متون شد. وجه تمایز این پژوهش در مرحلهٔ پردازش در مقایسه با سیستم‌های مشابه این است که از نقش‌های دستوری کلمات در وزن‌دهی به کلمات و جملات استفاده شده است که همین مسئله باعث افزایش دقت در نتایج گردیده است. مجموع آزمایش‌ها نشان داد که دخالت نقش‌های دستوری در انتخاب درست و دقیق‌تر کلمات و جملات مهم متن ورودی باعث افزایش دقت خلاصه به میزان ۷ درصد گردیده است.

علاوه بر آن روش، در پیمایش گراف پیشنهاد شده است که جملات به شکلی انتخاب شوند که با وجود این‌که دارای شباهت به‌عنوان و کلمات کلیدی هستند، اما به جملات انتخاب‌شدهٔ قبلی شباهت کمتری داشته باشند. در بخش نتایج نشان داده شد که این تغییر در پیمایش گراف باعث افزایش دقت کل به میزان ۱۰ درصد شده است.

یکی از تحقیقات آتی که در جهت افزایش دقت سیستم می‌توان انجام داد، استفادهٔ بهینه از ضریب تأثیر نقش‌های دستوری به شکلی است که به تناسب کلاس متن ورودی از ضرایب مخصوص به آن کلاس استفاده شود. به‌عنوان نمونه، در متون اقتصادی ضرایب تأثیر اعداد بیشتر از متون اجتماعی است. اما در متون اجتماعی ضریب تأثیر صفات و قیود بیشتر از متون اقتصادی است.

فهرست منابع

پورمعصومی، آصف، محسن کاهانی، سیداحمد طوسی، احمد استیری، و هادی قائمی. ۱۳۹۳. ایجاز: یک سامانه عملیاتی برای خلاصه‌سازی تک‌سندی متون خبری فارسی. *پردازش علایم و داده‌ها* ۱ (۲): ۳۳-۴۸.

کریمی، زهره، و مهرنوش شمس‌فرد. ۱۳۸۵. سیستم خلاصه‌ساز خودکار متون فارسی. دوازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران. تهران.

شاکری، حسن؛ فاطمه تقویان و فاطمه بهبودی، ۱۳۹۰، یک روش جدید خلاصه‌سازی متن فارسی مبتنی بر ویژگی‌های جملات، *دومین همایش فناوری اطلاعات، حال، آینده*، مشهد، دانشگاه آزاد اسلامی واحد مشهد.

Al-Hashemi, R., 2010. Text Summarization Extraction System (TSES) Using Extracted Keywords. *Int. Arab J. e-Technol.*, 1 (4), pp. 164–168.

Barzilay, R. & Elhadad, M., 1999. Using lexical chains for text summarization. *Advances in automatic text summarization*, pp. 111–121.

Dalianis, H., 2000. *SweSum-A Text Summarizer for Swedish*, Available at: https://people.dsv.su.se/~hercules/papers/Textsumsummary.html.

Diola, A.M. et al., 2004. Automatic Text Summarization. In *Proceedings of the 2 nd National Natural Language Processing Research Symposium*.

Dunning, T., 1993. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19 (1), pp. 61–74.

Erkan G. & Radev, D., 2004. LexRank: Graph-based Lexical Centrality As Salience in Text Summarization. *Journal of Artificial Intelligence*, 22 (1), pp. 457–479.

Güngör, T., 2010. Part-of-Speech Tagging. In N. Indurkhya & F. Damerau, eds. *Handbook of natural language processing*. CRC Press, pp. 205–235.

Guo-shun, W., 2011. Dynamic pages sequencing strategy. In *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*. pp. 591–593.

Hovy, E., 2003. Text Summarization. In R. Mitkov, ed. *The Oxford Handbook of Computational Linguistics*. Oxford University Press, pp. 583–598.

Jivani, A.G. & others, 2011. A comparative study of stemming algorithms. *Int. J. Comp. Tech. Appl*, 2 (6), pp.1930–1938.

Luhn, H.P., 1957. A Statistical Approach to Mechanized Encoding and Searching of Literary Information. *IBM Journal of Research and Development*, 1 (4), pp. 309–317.

Mani, I. & Maybury, M.T., 1999. *Advances in automatic text summarization*, MIT press.

Miller, G.A. et al., 1990. WordNet: An on-line lexical database. *International Journal of Lexicography*, 3, pp. 235–244.

Moghaddas, B.B. et al., 2013. Pasokh: A standard corpus for the evaluation of Persian text summarizers. In *ICCKE 2013*. Mashhad: IEEE, pp. 471–475.

Nadkarni, P.M., Ohno-Machado, L. & Chapman, W.W., 2011. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18 (5), pp. 544–551. Available at: http://jamia.oxfordjournals.org/content/18/5/544.

Nenkova, A. & McKeown, K., 2012. A Survey of Text Summarization Techniques. In C. C. Aggarwal & C. Zhai, eds. *Mining Text Data*. Boston, MA: Springer US, pp. 43–76.

Radev, D. et al., 2006. MEAD Documentation v3.10. *University of Michigan*, pp.1–64.

Radev, D.R. et al., 2004. MEAD-A Platform for Multidocument Multilingual Text Summarization. In *LREC*.

Shakeri, H. et al., 2012. A New Graph-Based Algorithm for Persian Text Summarization. In J. J. (Jong Hyuk) Park et al., eds. *Computer Science and Convergence: CSA 2011 {&} WCC 2011 Proceedings*. Dordrecht: Springer Netherlands, pp. 21–30.

Song, Y.-I., Han, K.-S. & Rim, H.-C., 2004. A term weighting method based on lexical chain for automatic summarization. In *International Conference on Intelligent Text Processing and Computational Linguistics*. pp. 636–639.

Taghva, K., Beckley, R. & Sadeh, M., 2005. A Stemming Algorithm for the Farsi Language. In *ITCC (1)*. pp. 158–162.

Tashakori, M., Meybodi, M. & Oroumchian, F., 2002. Bon: The persian stemmer. In *EurAsia-ICT 2002: Information and Communication Technology*. Springer, pp. 487–494.

Vanderwende, L. et al., 2007. Beyond SumBasic: Task-focused summarization with sentence

simplification and lexical expansion. *Information Processing and Management*, 43 (6), pp.1606–1618.

Zamanifar, A., Minaei-Bidgoli, B. & Sharifi, M., 2008. A new hybrid farsi text summarization technique based on term co-occurrence and conceptual property of the text. In *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008. SNPD'08. Ninth ACIS International Conference on*. pp. 635–639.

طیبه حسینی‌خواه

متولد سال ۱۳۵۷، دارای مدرک کارشناسی ارشد مهندسی صنایع گرایش مهندسی سیستم‌های اقتصادی-اجتماعی از دانشگاه صنعتی امیرکبیر تهران است. ایشان هم‌اکنون در حوزه برنامه‌ریزی، در شرکت ملی نفت ایران مشغول به کار است.

پژوهش در حوزه داده‌کاوی و هوش تجاری از جمله علایق پژوهشی وی است.

عباس احمدی

متولد سال ۱۳۵۷، دارای مدرک تحصیلی دکتری در رشته مهندسی صنایع- مهندسی طراحی سیستم‌ها از دانشگاه واترلو کانادا است. ایشان هم‌اکنون استادیار دانشکده مهندسی صنایع و سیستم‌های مدیریت دانشگاه صنعتی امیرکبیر تهران است.

هوش تجاری، سیستم‌های هوشمند، داده‌کاوی، بازیابی اطلاعات و دانش و سیستم‌های سلامت از جمله علایق پژوهشی وی است.

آزاده محبی

متولد سال ۱۳۵۷، دارای مدرک دکتری در رشته مهندسی طراحی سیستم‌ها از دانشگاه واترلو کانادا است. ایشان هم‌اکنون استادیار پژوهشکده فناوری اطلاعات پژوهشگاه علوم و فناوری اطلاعات ایران (ایرانداک) است.

داده‌کاوی، سیستم‌های هوشمند، بازشناسی الگو، متن‌کاوی و بازیابی اطلاعات از جمله علایق پژوهشی وی است.