چهاردهمین کنفرانس ملی سالانه انجمن کامپیوتر ایران
دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) ایران، تهران – ۲۰ و ۲۱ اسفندماه ۱۳۸۷

سیستم خلاصه‌ساز خودکار متن‌های فارسی

مجید بهره‌پور
گروه تحقیقاتی سیستم‌های فراگیرنده، دانشگاه Twente، هلند
M_Bahrepour@ieee.org

الهام مهدی‌پور
گروه کامپیوتر مؤسسه آموزش عالی خاوران مشهد
elham.mahdipour@gmail.com

آزاده کامل
دانشگاه آزاد اسلامی مشهد
azadeh_kamel@hotmail.com

ملیحه امیری
مؤسسه آموزش عالی خاوران مشهد
malihe_amiri@hotmail.com

آیدا طهماسبی
مؤسسه آموزش عالی خاوران مشهد
aida.tahmaseby@yahoo.com

محمدرضا اکبرزاده‌توتونچی
دانشگاه فردوسی مشهد-گروه برق و کامپیوتر
akbarzadeh@ieee.org

چکیده: امروزه با رشد سریع اطلاعات و داده‌ها، یافتن اطلاعات مناسب و کارا از اهمیت خاصی برخوردار است. هدف خلاصه‌سازی خودکار متن، فراهم کردن خلاصه‌ای از محتویات مطابق با اطلاعات مورد نیاز کاربر است. در این مقاله، نگارندگان ابتدا مفاهیم خلاصه‌سازی و انواع آن، سپس سیستم‌های خلاصه‌ساز موجود، و در نهایت روش خلاصه‌سازی خودکار متن‌های فارسی پیشنهادی را بررسی نموده‌اند. روش پیشنهادی، ترکیبی از روش‌های مبتنی بر گراف، TF-IDF و الگوریتم ژنتیک (Genetic Algorithm) است. در این روش کلمات قبل از امتیازدهی جملات، ریشه‌یابی می‌شوند. پس از امتیازدهی، جملات خلاصه با استفاده از الگوریتم ژنتیک (GA) انتخاب می‌شوند. تابع برازندگی الگوریتم ژنتیک مبتنی بر سه فاکتور شباهت با عنوان، قابلیت خوانایی و پیوستگی است. ارزیابی خلاصه‌های حاصل از پیاده‌سازی سیستم پیشنهادی در انتهای مقاله آورده شده است.

واژه های کلیدی: الگوریتم ژنتیک، تابع برازندگی، خلاصه‌سازی، Genetic Algorithm، TF-IDF.

۱- مقدمه

امروزه مردم در محل کار و زندگی خود به اطلاعات بیشتری نیاز دارند، استفاده از اینترنت دستیابی به اطلاعات را آسان می‌سازد؛ بنابراین در عصر دیجیتال، خلاصه‌سازی خودکار متن نقش قابل توجهی را ایفا می‌کند. خلاصه‌سازی خودکار متن، همان کوتاه کردن متن‌ها توسط کامپیوتر، درعین حفظ نکات مهم متن اصلی می‌باشد[1]. تولید خودکار خلاصه به وسیله ماشین دارای مزایای کنترل اندازه خلاصه، پیش‌بینی محتوا و ارتباط آسان عنصر متن در خلاصه با موقعیت آن در متن اصلی می‌باشد[2]. تحقیق بر روی خلاصه‌سازی خودکار متن تاریخچه بسیار طولانی دارد که می‌تواند به حداقل ۴۰ سال قبل برگردد، از زمانی که اولین سیستم IBM ساخته شد[3]. با بررسی اجمالی رویکردهای مختلف می‌توان سه مرحله اصلی در سیستم‌های خلاصه‌ساز تشخیص داد: مرحله شناسایی عناوین، تفسیر و تولید خلاصه[۱]. مرحله شناسایی، شامل فیلتر ورودی و حفظ مهمترین موضوعات است. در مرحله تفسیر، دو یا چند موضوع استخراج شده به یک یا چند مفهوم آمیخته می‌شوند. مرحله تولید، جملات استخراج شده را مجدداً فرمول‌بندی می‌کند و مفاهیم را می‌آمیزد و سپس خلاصه مناسب را تولید می‌کند[4].

ساختار مقاله حاضر به شکل زیر است: در بخش۲ مفاهیم خلاصه‌سازی و انواع آن مورد بررسی قرار می‌گیرد؛ بخش۳ سیستم‌های خلاصه‌ساز موجود معرفی می‌گردد؛ در بخش۴ سیستم خلاصه‌ساز پیشنهادی بررسی می‌شود؛ در بخش۵ سیستم پیشنهادی مورد ارزیابی قرار می‌گیرد و در بخش۶ در مورد نتایج بدست آمده بحث می‌شود.

۲- انواع خلاصه‌سازی

برای دسته‌بندی سیستم‌های خلاصه‌ساز، هیچ مسیر واضحی وجود ندارد و این سیستم‌ها معمولاً در آن واحد به چندین دسته متمایل هستند. برخی از محققین خلاصه‌سازی را از سه جنبه‌ی اصلی طبقه‌بندی نمودند: نوع منبع ورودی، هدف خلاصه‌سازی، خلاصه تولید شده.

۲-۱ نوع منبع ورودی

منظور از منبع، نمایش گروه زیادی از قالب‌های ورودی و همچنین نقاط شروع در اطلاعات که احتمال خلاصه شدن دارند، است. ویژگی‌های اصلی منبع ورودی که باعث تفاوت خلاصه‌سازها می‌شود عبارتند از: تعداد اسناد، زبان، میزان عمومیت، طول سند.

۲-۲ هدف خلاصه‌سازی

منظور از هدف، کاربرد نهایی از خلاصه تهیه شده است. برای تعیین هدف خلاصه‌سازی باید کاربرد خلاصه، هدف خلاصه و کاربر در نظر گرفته شوند.

۲-۳ خلاصه تولید شده (خروجی)

برای مشخص کردن قالب خروجی خلاصه و اطلاعاتی که در آن قرار دارد از شاخص‌های زیر استفاده می‌شود: *اشتقاق*: خلاصه‌ها به دو نوع چکیده و مستخرج تقسیم‌بندی می‌شوند. تولید خلاصه از نوع چکیده نیازمند روش‌های NLP، تجزیه معنایی و... است که هنوز در حال مطالعه و پیشرفت است. برای خلاصه مستخرج، از روش‌های آماری استفاده می‌شود. خروجی یک مجموعه از جملات مهم متن است که

عیناً از متن اصلی استخراج می‌شوند. خلاصه‌سازی مستخرج پیاده‌سازی ساده‌تری دارد اما سه مشکل عمده دارد: ۱) یافتن جملات مهم برای استفاده در خلاصه. ۲) تولید یک خلاصه پیوسته. ۳) حذف تمام افزونگی‌ها در خلاصه. *قالب*: قالب خروجی ممکن است متن، جدول، خطوط زمان، نمودار، تصویر و... باشد. *جانبداری*: خروجی خلاصه می‌تواند با جانبداری از نظر خاص یا بی‌طرف، ثابت یا متغیر باشد.

٤-٢ روش‌های خلاصه‌سازی

روش‌های بکار رفته در خلاصه‌سازی متن را می‌توان به دو دسته‌ی کلی تقسیم کرد؛ دسته اول روش‌هایی که از اطلاعات آماری متن برای تعیین اهمیت جملات استفاده می‌کنند؛ دسته دیگر که روابط بین بخش‌های مختلف متن، مفاهیم و معانی عبارات را نیز مورد توجه قرار می‌دهند.

برای خلاصه‌سازی متن، لازم است جملات وزن‌دهی شوند. امتیازدهی جملات به روش‌های مختلفی انجام می‌شود از جمله: روش‌های مبتنی بر بازیابی اطلاعات (IR) [4]، روش‌های TF-IDF، محاسبه مفهوم مبتنی بر دانش، روش‌های LSA [6]، روش‌های مبتنی بر گراف [7]، روش‌های زنجیره لغوی، تکنیک‌های یادگیری ماشین [8]. در این میان روش TF-IDF که در این مقاله استفاده شده است، بررسی می‌گردد.

روش TF-IDF یک روش توزیع کلمه است. این روش معادل معیار فرکانس کلمه - معکوس فرکانس سند در مفهوم بازیابی اطلاعات (IR) است[۱]. در این روش امتیاز کلمه از فرمول زیر بدست می‌آید[5]:

$$tf_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}} \qquad isf_i = Log(\frac{N}{n_i}) \qquad (۱)$$

وزن‌های TF-ISF برای هر جمله محاسبه می‌شوند که در آن j به جمله jام و i به کلمه‌iام در جمله jام اشاره دارد. $Freq_{i,j}$ فراوانی کلمه iام در جمله jام است. $Freq_{L,j}$ ماکزیمم فراوانی کلمه L در جمله jام است. N تعداد جملات سند است و n_i تعداد جملات شامل کلمه است.

وزن هر کلمه از رابطه (۲) بدست می‌آید. (۲) $W_{i,j} = tf_{i,j} * isf_{i,j}$

۳- کارهای انجام شده برای زبان فارسی

تا کنون خلاصه‌سازهای زیادی برای زبان‌های انگلیسی، آلمانی، سوئدی و... طراحی شده است اما طبق بررسی‌های انجام شده تنها سه سیستم خلاصه‌ساز فارسی معرفی شده است. در [۲] به چکیده‌سازی از چند منبع ورودی توجه شده است. در [9] سیستم خلاصه‌ساز FarsiSum معرفی شده است که در واقع نسخه تغییر یافته یک سیستم خلاصه‌ساز متن های سوئدی به نام SweSum [10] برای زبان فارسی است. در [۱] از ترکیب روش‌های زنجیره لغوی و مبتنی بر گراف استفاده شده است. در این روش ارتباط بین جملات بر اساس زنجیره لغوی، شباهت بین جملات بر اساس مفهوم آن‌ها، محاسبه می‌شود.

سیستم‌های خلاصه‌ساز فوق دارای یکسری معایب هستند، از جمله این معایب می‌توان به موارد زیر اشاره کرد: سیستم خلاصه‌ساز FarsiSum فاقد راه‌حل‌های خاص زبان در پروسه جداسازی نشانه، و الگوریتم‌های ارزیابی است. سیستم خلاصه‌ساز معرفی شده در [۱] نیز دارای اشکالاتی است از جمله عدم بررسی ضمایر موجود در متن و جایگزین نمودن مرجع ضمیر با آن، وجود افزونگی در خلاصه، ابهام در کلمات اشاره و آخرین مورد اینکه در این سیستم از علامت ":" به عنوان جداساز جمله استفاده شده است در حالی که جملاتی که بیان کننده اظهار نظر یا بیان کننده مواردی هستند مانند "وی گفت:" یا نباید در خلاصه وارد شوند و یا باید به همراه جمله بعد از خود بیایند.

٤- معرفی سیستم خلاصه‌ساز فارسی پیشنهادی

در سیستم خلاصه‌ساز پیشنهادی سعی شده است معایب سیستم‌های موجود برطرف گردد. سیستم خلاصه‌ساز متن‌های فارسی پیشنهادی، خلاصه‌ای مستخرج تولید می‌کند. ایده بکار رفته در این خلاصه‌ساز، ترکیبی از روش‌های مبتنی بر گراف و الگوریتم ژنتیک است که شباهت‌هایی با [5] دارد. در [5] سیستم خلاصه‌سازی برای متن‌های انگلیسی معرفی شده است. این سیستم پس از وزن‌دهی جملات و تشکیل ماتریس شباهت برای جملات سند، گراف جهت‌داری بوجود می‌آورد. پس از آن سه فاکتور شباهت با عنوان، پیوستگی و قابلیت خوانایی برای جملات موجود در خلاصه محاسبه می‌شود. در مرحله بعد با استفاده از تابع برازندگی تعریف شده در [5] و الگوریتم ژنتیک، جملات خلاصه انتخاب می‌شوند. در [5] نگارندگان، جمعیت اولیه الگوریتم ژنتیک را به تعداد کل جملات متن در نظر گرفتند، چنانچه جمله در خلاصه باشد مقدار آن کروموزم، یک و اگر جمله در خلاصه نباشد مقدار کروموزم صفر است. دراین مطالعه، مرجع [5] به صورت زیر توسعه داده شده است: ۱) استفاده از الگوریتم ریشه‌یابی کلمات و تشخیص کلمات مترادف. ۲) تغییر ساختار برای پذیرفتن متن‌های فارسی. ۳) ارائه یک نسخه ساده‌تر و کارآمدتر برای کد کردن ژن‌ها در الگوریتم ژنتیک که پیچیدگی زمانی و مکانی الگوریتم اجرایی را کاهش می‌دهد. همچنین در سیستم خلاصه‌ساز پیشنهادی مشکلاتی همچون عدم قانونمندی زبان فارسی، تعدد معانی در زبان فارسی، وجود کلمات مرکب و دو بخشی در فارسی و وجود کلمات استثناء در زبان فارسی حل شده است. نوآوری تکنیک‌های بکار رفته در سیستم خلاصه‌ساز پیشنهادی عبارتند از: ۱) در نظر گرفتن جمعیت اولیه الگوریتم ژنتیک به تعداد جملات خلاصه بطوریکه هر کروموزم نشانگر شماره‌ی جمله‌ای است که در خلاصه می‌آید. در واقع می‌توان گفت نگارندگان از نسخه صحیح الگوریتم ژنتیک (Real GA) استفاده نموده‌اند. با این کار در مصرف فضای حافظه صرفه‌جویی می‌شود و سرعت الگوریتم بالا می‌رود زیرا تعداد جمعیت کمتر است. ۲) نگارندگان به جای استفاده از میانگین‌های آماری که در عملگر برش Real GA مرسوم است، در

عملگر برش از روش برش در نسخه باینری GA استفاده کرده‌اند، همچنین در این روش با احتمالی مناسب، از ژن قوی‌تر، کروموزم‌های بیشتری برداشته می‌شود. بنابراین ژن قوی‌تر نقش مؤثرتری در تولید جمعیت جدید ایفا می‌کند. عملگر جهش نیز با احتمال مناسبی بر روی ژن‌ها اعمال می‌شود. ۳) استفاده از الگوریتم ریشه‌یابی و استفاده از جدول استثنائات برای حل مشکل وجود کلمات استثناء در زبان فارسی مثلاً کلماتی که به علائم جمع ختم می‌شوند مانند اژدها، اطمینان و غیره. ۴) استفاده از جدول مترادف‌ها برای حل مشکل تعدد معانی در زبان فارسی. ۵) استفاده از مرحله پیش‌پردازش برای یکپارچه‌سازی کلمات دوبخشی و حل مشکل کلمات مرکب.

در روش پیشنهادی از الگوریتم ریشه‌یابی کراوتز [۳] استفاده شده است. الگوریتم کراوتز اولین بار در سال ۱۹۹۳ معرفی شد. این الگوریتم از روش‌های ریخت‌شناسی و یک فرهنگ لغت برای آزمودن ریشه‌های یافت شده، استفاده می‌کند. در [۳] نگارندگان با استفاده از الگوریتم ریشه‌یابی کراوتز، آن را برای زبان فارسی پیاده‌سازی نموده‌اند. نگارندگان در این مقاله برای ریشه‌یابی کلمات از الگوریتم پیشنهادی در [۳] بهره برده‌اند.

مراحل الگوریتم سیستم خلاصه‌ساز پیشنهادی مانند زیر است: در مرحله اول با استفاده از بانک‌های اطلاعاتی کلمات مترادف، خاص و زائد؛ و الگوریتم ریشه‌یابی، تمامی کلمات و افعال ریشه‌یابی می‌شوند و حروف اضافه، نشانه‌های جمع و کلمات غیر مهم حذف می‌گردند. در مرحله دوم، فرکانس کلمات با استفاده از **TF-IDF**، فرمول‌های (۱) بدست آورده می‌شوند. سپس وزن هر کلمه در جمله از فرمول (۲) محاسبه می‌شود. لازم به ذکر است که وزن عنوان و کلمات کلیدی کاربر از فرمول (۳) محاسبه می‌شود. (۳)

$$W_{i,j} = (0.5 + \frac{0.5 * freq_{i,q}}{\max_l freq_{l,q}}) * isf_i$$

که در آن **q** نشانگر جمله عنوان یا جمله شامل کلمه کلیدی کاربر است. مرحله سوم، ساخت ماتریس شباهت با استفاده از فرمول‌های (۴) و (۵) است که در (۴) شباهت جمله با عنوان و در (۵) شباهت دو جمله با یکدیگر محاسبه می‌شود.

$$sim(s_j, q) = \frac{\sum_{i=1}^{t} w_{i,j} * w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} * \sqrt{\sum_{i=1}^{t} w_{i,q}^2}} \quad (۴)$$

$$sim(s_m, s_n) = \frac{\sum_{i=1}^{t} w_{i,m} * w_{i,n}}{\sqrt{\sum_{i=1}^{t} w_{i,m}^2} * \sqrt{\sum_{i=1}^{t} w_{i,n}^2}} \quad (۵)$$

پس از ساخت ماتریس شباهت گراف وزن‌داری تشکیل داده می‌شود که در آن هر لبه برای دو بردار متصل به هم، میزان شباهت دو جمله با یکدیگر (۶) است. (۶)

$$\forall (s_i, s_j) \in E, W(s_i, s_j) = sim(s_i, s_j)$$
$$\forall i < N : sim(s_i, s_i) = 0, \forall i, j < N : sim(s_j, s_i) = 0$$

یک خلاصه خوب معمولاً شامل جملات مشابه با عنوان است. هرگاه شباهت جملات به یکدیگر محاسبه شد، می‌توانید فاکتور شباهت با عنوان را تعریف کنید. یک روش ساده، محاسبه میانگین شباهت جملات موجود در خلاصه تقسیم بر ماکزیمم میانگین است. میانگین شباهت با عنوان در خلاصه (S) با TR نمایش داده می‌شود. با استفاده از TR، فاکتور شباهت با عنوان (TRF) مانند (۷) محاسبه می‌شود. بطوریکه ماکزیمم از میان همه خلاصه‌های ممکن به طول S محاسبه می‌شود. برای پیدا کردن ماکزیمم، میانگین شباهت جمله با عنوان را برای S جمله بگیرید. TRF شباهت خلاصه ایجاد شده با عنوان سند را نشان می‌دهد. در خلاصه‌هایی که جملات مرتبط با عنوان هستند، TRF نزدیک به یک است؛ اما در خلاصه‌هایی که توسط جملات دور از عنوان ساخته می‌شوند، TRF به صفر میل می‌کند.

$$TR_s = \frac{\sum_{s_j \in summary} sim(s_j, q)}{S}, TRF_s = \frac{TR}{\max_{\forall summary} (TR)} \quad (۷)$$

فاکتور پیوستگی تعیین می‌کند که جملات موجود در خلاصه در مورد اطلاعات یکسانی صحبت می‌کنند یا خیر. فاکتور ایده‌آل، میانگین وزن همه لبه‌ها در زیرگراف خلاصه تقسیم بر ماکزیمم میانگین بین همه خلاصه‌های ممکن است. C میانگین شباهت همه جملات در خلاصه است. واضح است که C میانگین وزن همه لبه‌ها در زیرگراف خلاصه (S) است. بطوریکه N_S تعداد کل یال‌ها در خلاصه است. N_S می‌تواند بسادگی محاسبه شود. گره‌های خلاصه با $S_{S_1}, S_{S_2}, \ldots, S_{S_S}$ نمایش داده می‌شوند و N_S تعداد لبه‌ها از S_{S_1} به S_{S_j} ، $1<j\leq S$ بعلاوه تعداد لبه‌ها از S_{S_j} به S_{S_2}، $2<j\leq S$ و ... است، بنابراین $N_S=(S-1)+(S-2)+\ldots=\frac{(S)*(S-1)}{2}$.

$$C_s = \frac{\sum_{\forall s_i, s_j \in summary subgraph} W(s_i, s_j)}{N_s}, CF_s = \frac{\log(C*9+1)}{\log(M*9+1)}, M = \max_{i,j \leq N} sim_{i,j} \quad (۸)$$

CF باید نشان دهد که جملات خلاصه چگونه به هم وابسته هستند و به صورت (۸) تعریف می‌شود. M ماکزیمم وزن در گراف است یعنی M ماکزیمم شباهت جملات است. اگر بیشتر جملات خلاصه مشابه با عنوان باشند، CF رشد می‌کند و اگر جملات خلاصه دور از عنوان باشند، CF به صفر میل می‌کند. محاسبه فاکتور قابلیت خوانایی دشوار است. یک سند خوانا دارای جملات مرتبط است. اولین جمله و دومین جمله با میزان شباهت بالایی به یکدیگر مرتبط می‌شوند، به همین شکل جمله دوم و سوم و... . در حقیقت یک خلاصه خوانا یک زنجیره از جملات می‌سازد. فاکتور خوانایی خلاصه‌ای به طول **S** با $\mathbf{R_S}$ نمایش داده می‌شود. بنابراین فاکتور قابلیت خوانایی خلاصه **(RF)** به صورت (۹) محاسبه می‌شود. (۹)

$$R_s = \sum_{0 \leq i < S} W(s_i, s_{i+1}), RF_s = \frac{R_s}{\max_{\forall i} R_i}$$

به خاطر داشته باشید که فرض بر آن است که طول خلاصه ثابت است و ماکزیمم در بین همه خلاصه‌های ممکن به طول خلاصه محاسبه می‌شود. پیدا کردن این ماکزیمم در مرتبه چندجمله‌ای امکان‌پذیر است. طول خلاصه **S** نام داشت، بنابراین هدف پیدا کردن خلاصه‌ای با قابلیت خوانایی بیشتر است که برابر با پیدا کردن مسیری به طول **S** با ماکزیمم وزن در گراف سند است. پس از محاسبه این فاکتورها برای جملاتی که در خلاصه می‌آیند، با استفاده از تابع برازندگی (۱۰) میزان برازندگی

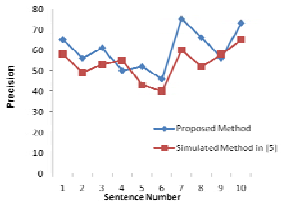
خلاصه محاسبه می‌شود[5]. تابع برازندگی این مسأله قابل انعطاف است زیرا پارامترهایی دارد که توسط کاربر تعیین می‌شوند. یک نفر ممکن است خلاصه‌ای با قابلیت خوانایی بالا بخواهد در حالی که کس دیگری ممکن است خلاصه‌ای با جملات مشابه به عنوان بخواهد و قابلیت خوانایی برایش مهم نباشد. برای داشتن چنین تابع برازندگی، یک تابع که میانگین وزن‌دار سه فاکتور می‌باشد، طراحی شده است.

$$F=\frac{\alpha*TRF+\beta*CF+\gamma*RF}{\alpha+\beta+\gamma} \qquad 0\le\alpha\le1,0\le\beta\le1-\alpha,\gamma=1-\alpha-\beta \quad (۱۰)$$

که در آن ضرایب α، β و γ اعدادی هستند که توسط کاربر تعیین می‌شوند و میزان تأثیرگذاری فاکتورهای شباهت با عنوان، پیوستگی و قابلیت خوانایی را تعیین می‌کنند. مقادیر TRF، RF و CF بین صفر و یک هستند، بنابراین نتیجه ترکیب آن‌ها عددی حقیقی بین صفر و یک است. پس از اعمال الگوریتم ژنتیک و رسیدن به برازندگی ایده‌آل، جملات موجود در جمعیت ایده‌آل، به ترتیب حضور در متن اصلی در خلاصه نمایش داده می‌شوند.

۵- ارزیابی سیستم خلاصه‌ساز پیشنهادی

برای ارزیابی سیستم از ده متن در مقوله‌های علمی و خبری استفاده شده است. این متن‌ها توسط افراد مختلف با نسبت فشردگی‌های مختلف خلاصه شده و با خروجی سیستم مقایسه شد. از آنجایی که در روش پیشنهادی از الگوریتم GA برای خروجی خلاصه استفاده می‌شود و ضرایب α، β و γ، همچنین نسبت فشرده‌سازی خلاصه توسط کاربر تعیین می‌شوند، خلاصه تولید شده متناسب با نیاز کاربر است. با مقایسه خلاصه سیستم و خلاصه‌های دستی مشاهده شد که به طور میانگین در ۶۰/۲۳ درصد موارد جملات خلاصه مانند جملات خلاصه‌های دستی انتخاب شده‌اند؛ این در حالی است که در الگوریتم معرفی شده در [5] که توسط نگارندگان مقاله حاضر برای زبان فارسی پیاده‌سازی شد، میانگین شباهت جملات خلاصه شده توسط سیستم به جملات خلاصه انسانی ۵۳٪ می‌باشد و خلاصه‌ساز پیشنهادی ۷٪ بهتر عمل کرده است. نتایج این مقایسه در شکل ۱ آورده شده است.



شکل ۱: ارزیابی سیستم خلاصه‌ساز فارسی پیشنهادی و سیستم [5]

۶- نتیجه گیری

در این مقاله یک سیستم خلاصه‌ساز متن‌های فارسی بر اساس ترکیبی از روش‌های مبتنی بر گراف و الگوریتم GA معرفی شد. نتایج ارزیابی نشان می‌دهد که جملات خلاصه کاملاً متناسب با کلمات کلیدی کاربر و درصد فشرده‌سازی هستند. بدیهی است بدلیل استفاده از الگوریتم ژنتیک، همچنین با کلمات کلیدی متفاوت، خلاصه‌های مختلفی ایجاد می‌گردد. در ایجاد خلاصه‌ی بهینه، ضرایب α، β و γ که توسط کاربر تعیین می‌شوند و میزان تأثیرگذاری فاکتورهای شباهت با عنوان، پیوستگی و قابلیت خوانایی را مشخص می‌کنند، نقش اساسی ایفا می‌نمایند.

مراجع

[۱] کریمی، زهره. شمس‌فرد، مهرنوش. **"خلاصه‌سازی خودکار متون فارسی"**، دوازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، صفحه۱۲۸۶، ۱۳۸۵.

[۲] شهابی، امیرشهاب. **"چکیده‌سازی متون زبان فارسی"**، دومین کنفرانس بین‌المللی علوم‌شناختی، صفحه۵۶، تهران، ۱۳۸۱.

[۳] حسامی‌فرد، رضا. قاسم‌ثانی، غلامرضا. **"طراحی یک الگوریتم ریشه‌یابی برای زبان فارسی"**، یازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، ۱۳۸۴.

[1] Hassel,M.,*"Resource Lean and Portable Automatic TextSummarization"*,2007,Stockholm,Sweden.p.144.

[2] Visser,W.T.,Wieling M.B.,*"Sentence-based Summarization of Scientific Documents"*, M.S.Project, University of Groningen.

[3] HongyanJing,*"Summary Generation through Intelligence cutting and pasting of the Input Document"*, Department of Computer Science.

[4] Jen-Yuan Yeh,H.-R.K,Wei-Pang Yang,I-HengMeng, "Text Summarization using a trainable summarizer and laten semantic analysis",*Elsevier*,2005.

[5] Qazvinian,Vahed.,SharifHassnabadi,Leila., Halavati, Ramin.,*"Summarizing Text With a Genetic Algorithm-Based Sentence Extraction"*, Department of Computer Engineering.

[6] Lagzian,S.,*"Text Mining"*,2007,Azad University of Mashhad.

[7] Ohtake,K.,Okamoto,D.,Kodama,M.,Masuyama,S.,*"Yet another summarization system with two modules using empirical knowledge"*,In Proceeding of NTCIR Workshop2 Meeting,2001.

[8] Neto,L.,Freitas,A.,Kaestner,C.,*"In GBittencourt and GL Ramalho"*,editors,Proc.16th Brazilian Symp.on Artificial Intelligence(SBIA-2002),Lecture Notes in Artificial Intelligence 2507,pp205-215,2002.

[9] Mazdak,N.,Hassel,M.,*"FarsiSum-a Persian Text Summarizer"*,Master thesis,Department of Linguistics, Stockholm University.

[10] Dalianis,H.,*"SweSum-A Text Summarizer for Swedish, Technical Report"*,TRITANA-p0015, IPLab-174,NADA,KTH,October2000.