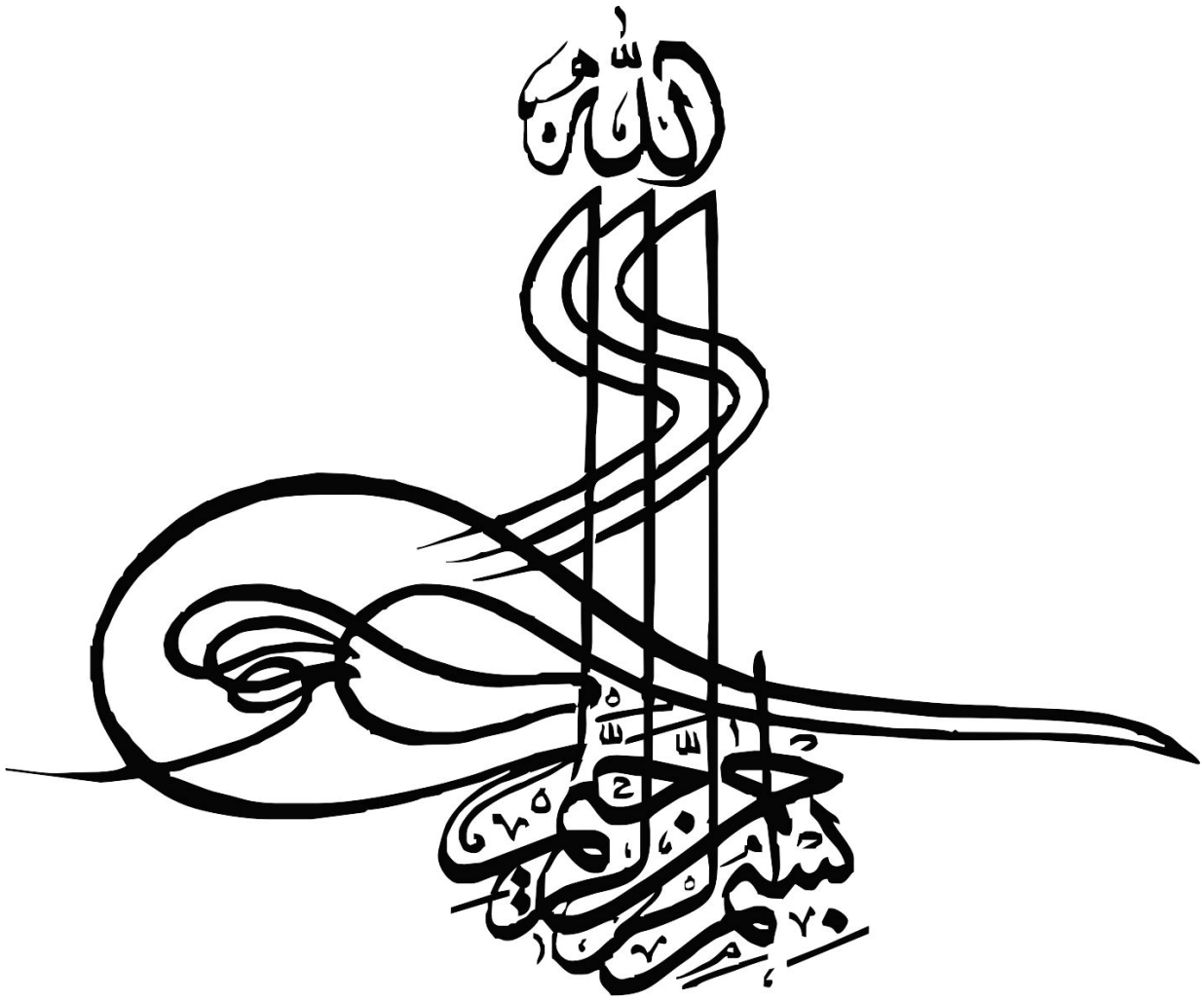

بسم الله الرحمن الرحيم

دانشکده: کامپیوتر و فن آوری اطلاعات

گروه: هوش مصنوعی

پایان‌نامه دوره‌ی کارشناسی ارشد مهندسی کامپیوتر - هوش مصنوعی

انتخاب بهینه ی کلمات کلیدی برای موتور های جستجو

حمزه هدهدکیان

استاد راهنما:

دکتر مرتضی زاهدی

استاد مشاور:

دکتر حمید حسن پور

تیرماه 1390

تقدیم به

مهربانی مادرم

آرامش پدرم

صبوری همسرم

با تشکر

از اساتید دانشکده مهندسی کامپیوتر، که بدون راهنمایی های بی دریغ ایشان رسیدن به پایان ممکن نبود

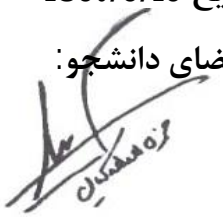
تعهدنامه

اینجانب **حمزه هدهدکیان** دانشجوی دوره کارشناسی ارشد رشته **هوش مصنوعی** دانشکده **کامپیوتر و فناوری اطلاعات** دانشگاه **صنعتی شاهرود** نویسنده پایان‌نامه تحت عنوان **انتخاب بهینه کلمات کلیدی برای موتور های جستجو** تحت راهنمایی‌های **دکتر مرتضی زاهدی** متعهد می‌شوم:

- تحقیقات در این پایان‌نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهش‌های محققان دیگر به مرجع مورد استفاده استناد شده است.
- مطالب مندرج در پایان‌نامه تا کنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.
- کلیه حقوق این اثر متعلق به دانشگاه صنعتی شاهرود می‌باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود» و یا « **Shahrood University of Technology** » به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان‌نامه تاثیرگذار بوده‌اند در مقالات مستخرج از پایان‌نامه رعایت می‌گردد.
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که از موجودات زنده (یا بافت‌های آنها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری، ضوابط و اصول اخلاقی رعایت شده است.

تاریخ 1390/9/19

امضای دانشجو:



مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه‌های رایانه‌ای، نرم‌افزارها و تجهیزات ساخته شده) متعلق به دانشگاه صنعتی شاهرود می‌باشد. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود.

دانشگاه صنعتی شاهرود

دانشکده :مهندسی کامپیوتر و فناوری اطلاعات

پایان نامه ارشد آقای حمزه صمدکیان

تحت عنوان:

انتخاب بهینه ی کلمات کلیدی برای موتورهای جستجو

در تاریخ۱۷/۱۲/۹۰ توسط کمیته تخصصی زیر جهت اخذ مدرک کارشناسی ارشد مورد ارزیابی و با

درجه عالی مورد پذیرش قرار گرفت.

اساتید راهنما	امضاء	اساتید مشاور	امضاء
نام و نام خانوادگی: دکتر مرتضی زاهدی		نام و نام خانوادگی: دکتر حمید حسن پور	
نام و نام خانوادگی:		نام و نام خانوادگی:	

اساتید داور	امضاء	نماینده تحصیلات تکمیلی	امضاء
نام و نام خانوادگی: دکتر علی اکبر پویان		نام و نام خانوادگی: مهندس علی دوست محمدی	
نام و نام خانوادگی: دکتر حسین مروی			
نام و نام خانوادگی:			
نام و نام خانوادگی:			

چکیده

به طور کلی، روش‌های ارائه شده جهت استخراج خودکار کلمات کلیدی، سعی در بدست آوردن نتایج بهتر در معیارهایی مانند بازخوانی و دقت دارند. اگرچه این معیارها، میزان کارایی روش استخراج کلمات کلیدی در نقش یک انسان را نشان می‌دهد اما با توجه به نقش غیرقابل انکار موتورهای جستجو در دنیای امروز، به نظر می‌رسد که در انتخاب کلمات کلیدی علاوه بر توجه به معیارهای رایج بازیابی اطلاعات باید به افزایش میزان دسترسی پذیری متن توسط موتورهای جستجو نیز توجه ویژه ای شود.

در این تحقیق روشی جدید برای استخراج خودکار کلمات کلیدی ارائه شده است که همزمان با افزایش دسترسی پذیری متن، امتیاز مناسبی در معیارهای بازخوانی و دقت نیز کسب می کند. روش ارائه برای استخراج کلمات کلیدی از دو تابع امتیاز دهی استفاده می کند: تابع امتیاز دهی به کلمات کلیدی و تابع ارزیابی میزان دسترسی پذیری. تابع اول سعی در بالا بردن بازخوانی و دقت دارد در حالیکه تابع دوم در طول فرایند آموزش با استفاده از الگوریتم ژنتیک و بازخوردهای موتورهای جستجو به بهینه سازی ضرایب خصوصیات درتابع اول می‌پردازد.

همچنین در این پروژه با بهره گیری از کلمات برجسته ساز، فرایند پس‌پردازشی ارائه شده که با گزینش نهایی کلمات کلیدی از میان لیست کلمات کاندید، منجر به بهبود کارایی روش در معیار دقت می شود.

آزمایشات نشان می دهد که با به کارگیری تعداد تکرار مناسب در طی فرایند آموزش و ایجاد موازنه منطقی در کسب هریک از سه معیار یاد شده می توان به نتایج مطلوبی در هر سه معیار دست یافت.

واژگان کلیدی: استخراج خودکار کلمات کلیدی، بهینه‌سازی نتایج موتورهای جستجو، کلمات پرتکرار فارسی، الگوریتم ژنتیک، کلمات برجسته ساز فارسی، بهبود معیار دقت، افزایش دسترسی پذیری

لیست مقالات مستخرج از این پایان نامه

[1] Hamzeh Hodhodkian, Morteza Zahedi,"**Improving Precision in Automatic Keyword Extraction Using Attention Attraction Strings**", Arabian Journal for Science and Engineering, July 2011

[2] Hamzeh Hodhodkian, Morteza Zahedi," **An Efficient Approach for Keyword Selection; Improving Accessibility of Web Contents by General Search Engine**", International Journal of Web & Semantic Technology (IJWesT), July 2011

فهرست شکل ها

فهرست جداول

فصل اول:

مفاهیم اولیه و تعاریف پایه

1-1 مقدمه

در این فصل به دنبال تعریف جایگاه پروژه انجام شده در تحقیقات امروزی و دلایل نیاز به آن هستیم. در ابتدای فصل با تعریف متن کاوی و بازیابی اطلاعات سعی در مشخص کردن جایگاه روش‌های استخراج کلمات کلیدی داریم. سپس با تعریف مفهوم کلمات کلیدی و ارائه ساختار کلی روش های استخراج کلمات کلیدی به بررسی تئوری های مرتبط با این فرایند و اشکالات و چالش های پیش رو با آن می-پردازیم.

1-2 استخراج کلمات کلیدی زیر وظیفه ای از متن کاوی

متون دیجیتال از دید یک دستگاه کامپیوتر رشته‌ای ساده از کاراکتر هاست در حالیکه در یک ساختار انسانی این رشته کاراکترها حاوی ارتباطات و معانی فراوانی در خود هستند. آنچه که مسلم است با توجه به حجم تولید اطلاعات در دنیای کنونی، نیاز شدیدی به خودکار سازی فرایند پردازش متون توسط کامپیوترها احساس می شود و به همین علت نیازمند روش‌ها و الگوریتم‌هایی برای پیش پردازش یا پردازش این متون جهت تبدیل آن ها به الگوهایی مناسب با ساختار های کامپیوتری هستیم.

مفهوم متن کاوی بصورت "*یافتن قوانین جذاب (به معنای مخفی در ذات متن، که بصورت بالقوه مفید هستند) در حجم وسیعی از پایگاه داده متنی*" تعریف می شود که می تواند به "*پیدا کردن مفهوم یا چیکده اطلاعات موجود در یک سند از روی متن ظاهری آن*" نیز گسترش یافته و تعبیر شود[1].

پروسه‌های متن کاوی را می‌توان در سه شاخه، بازیابی اطلاعات، استخراج اطلاعات و کشف دانش بررسی کرد.

بازیابی اطلاعات اصولا با بازیابی مستندات و مدارک مرتبط است. به این معنا که در این روش دانشی کشف نمی‌شود بلکه تنها بسته‌ای از کلمات از میان مجموعه‌ای از مستندات بیرون کشیده می‌شود که با توجه به نیاز مطرح شده از سوی کاربر، بیشترین ارتباط را با مفهوم درخواست یا جستجو شده داشته باشد.

اما برای تعریف **استخراج اطلاعات**، در نظر بگیرید که موسسه‌ای بسیار موفق دارید و از اینرو تعداد زیادی ایمیل در روز دریافت می‌کنید. شما می‌خواهید سابقه‌ی این ایمیل‌ها را ثبت کنید. اینکه چه کسانی آن‌ها را فرستاده اند، تاریخ فرستادن چه روزی بوده است، عنوان و متن آن چه بوده است و نظایر آن. این امر با بیرون کشیدن این اطلاعات از تک تک ایمیل‌ها و پر کردن یک پایگاه داده از این اطلاعات میسر است. می‌توانید برای این کار با تعریف یا کشف یک قالب از داده‌هایی که به آن سرو کار دارید به این هدف برسید. مثلا می‌شود برنامه‌ای داشت که به طور اتوماتیک به دنبال کلمه‌ی "عنوان" در سند بگردد و آنچه را که بعد از آن آمده است را به عنوان یک فیلد در پایگاه داده پر کند. هرچند یافتن این قالب در داده‌های غیر ساخته یافته‌ی دیگر ممکن است به هیچ وجه ساده نباشد، اما زمانی که شما این کار را به پایان بردید، با داده‌های کاملا ساخت یافته مواجهید که از درون ایمیل‌ها بدست آمده است. اما از سویی دیگر این نتیجه باز هم همان اطلاعاتی است که داشته اید. به این معنی که هیچ چیز جدیدی از آنچه دارید کشف نشده است و همان را که قبلا می‌دانستید هنوز هم می‌دانید.

برای روشن شدن مفهوم **کشف دانش** همان مثال قبل را در نظر بگیرد با این تفاوت که اینبار به دو مفهوم Aو B که از مجموعه برخورد کرده ایم که رابطه ای منطقی دارند. مثلا اینکه "مناطق بارانی" و "کشت برنج" دارای رابطه‌ای به این صورتند که *کشت برنج به مناطق بارانی نیاز دارد* $(A \to B)$. به علاوه فرض کنید که مفهوم B نیز با مفهوم C به همین شکل دارای ارتباط است. مثلا "مناطق شمال ایران" و

"مناطق بارانی" این رابطه ر ا با هم دارند که *مناطق شمال ایران منطقه‌ی بارانی هستند* $(B \to C)$. سیستمی که کمی باهوش‌تر باشد می‌تواند از این اطلاعات استخراج شده که در میان اطلاعات ما بوده اند. اطلاعاتی **نو** بدست آورد به این معنی که به رغم عدم وجود این رابطه بطور مستقیم سیستم کشف می-کند که "مناطق شمال ایران" قابل "کشت برنج" است $(A \to C)$. این مثال نشان می دهد که کشف دانش به چه معناست.

با وجود آنکه هدف اصلی آنالیز متون فهم محتوای متنی و حتی کشف دانش است اما این مسئله هنوز در دنیای الگوریتم‌ها و کامپیوترهای امروزه‌ای مسئله‌ای پیچیده به نظر می‌رسد که ما را بر آن می‌دارد که در *سطوح پایین تر* به دنبال *زیر وظایفی* باشیم که البته هریک از این زیر وظایف به خودی خود ارزش زیادی دارند.

یکی از این زیر وظایف *استخراج کلمات کلیدی* است که نقش بسزایی در عملی شدن سایر فرایندها دارد.

در ادامه به تعریف کلمات کلیدی می‌پردازیم و سپس بعد از ارائه مختصری از خصوصیات و مشکلات زبان فارسی به بحث های تئوریک در مورد مفاهیم استخراج کلمات کلیدی خواهیم پرداخت. و در پایان فصل با ارائه یک شکل کلی از فرایند استخراج کلمات کلیدی به تشریح ساختار پایان نامه می‌پردازیم.

3-1 کلمات کلیدی

1-3-1 تعریف کلمات کلیدی

به *مجموعه ای از کلمات یا عبارات کوتاه که بیان کننده منظور اصلی یک متن باشند* کلمات کلیدی آن متن گفته می‌شود[2] .

مجموعه کلمات کلیدی در حالت ایده آل در حین آنکه مفهوم اصلی متن را به طور کامل توضیح می دهد باید بتواند متن را از سایر متون به صورت منحصر به فرد جدا کند.

مسئله تعیین کلمات کلیدی به دو شکل تعریف می شود:

1. منتصب کردن کلمات کلیدی[1]: در این حالت به یک متن تعدادی از کلمات کلیدی که از یک دامنه محدود انتخاب شده اند منتصب می شود. به عنوان مثال سایت خبری تابناک بخش های خبری ویژه ای برای مسئله فلسطین در نظر گرفته است. حال هر خبری که در آن ارتباطی با فلسطین وجود داشته باشد یکی از کلمات کلیدی که به این خبر منتصب خواهد شد کلمه فلسطین خواهد بود. این دامنه محدود منجر می شود که مسئله انتصاب کلمات کلیدی به طور کامل به یک مسئله کلاسبندی تبدیل شود که در آن یک متن به عنوان یک نمونه عضوی از یک یا چند کلاس از کلاس‌های ایجاد شده توسط کلمات کلیدی محدود خواهد شد.
2. استخراج کلمات کلیدی[2]: انتخاب (یا پیشنهاد) کلماتی از یک متن که نشان دهنده مفهوم کلی آن باشد این پروسه بیشتر شبیه به یک رتبه بندی است که رتبه بالاتر به معنای قدرت بیشتر عبارت در تشریح متن مورد بررسی است.

1-3-2 کلمات کلیدی و فرایندهای متن کاوی

کلمات کلیدی کوچکترین واحدی نحوی هستند که می توانند معنای سند را با خود حمل کنند از اینرو در بسیاری از فرایندهای خودکار کاربردی متن کاوی از قبیل شاخص‌گذاری[3]، خلاصه‌سازی[4]،کلاسه‌بندی[5]،

[1] Keywords assignment
[2] Keywords extraction
[3] Indexing
[4] summarization
[5] classification

خوشه بندی[1]، فیلترینگ[2]، تشخیص و ردیابی موضوع[3] و نیز مصورسازی اطلاعات[4] پایه تحقیقات بر کلمات کلیدی بنا نهاده می شود که باعث می شود استخراج کلمات کلیدی هسته ی تکنولوژیکی تمامی فرایندهای متن کاوی به شمار آید.[3]

از آنجا که به طور معمول در بسیاری از اسناد کلمات کلیدی مشخص نشده و نیز انجام این فرایند با استفاده از نیروی انسانی زمانبر بوده و هزینه زیاد و خطاهای احتمالی در پی‌دارد در نتیجه تحقیقات زیادی برای اتوماتیک سازی روش های استخراج کلمات کلیدی انجام گرفته است. اما از آنجا که زبان فارسی ویژگی های منحصر به فرد خود را داراست می طلبد که به مقوله استخراج کلمات کلیدی در متون فارسی نگاه ویژه ای معطوف شود.

3-3-1 کلمات کلیدی و موتور های جستجو

موتورهای جستجو در دنیای امروز نقش غیرقابل انکاری در ارائه دانش به کاربران دارند. در واقع حتی گاهی این مسئله مطرح می شود که اگر یک مطلب توسط یک موتور جستجو دیده نشود شاید هرگز دیده نشود. همچنین بررسی ها نشان می دهد نحوه رتبه بندی این موتورها بسته به کلمات کلیدی به کار برده شده در مطلبی است که موتور جستجو آن را به عنوان پاسخ یک عبارت جستجو [5] ارئه می‌دهد. در نتیجه منطقی به نظر می رسد که در انتخاب کلمات کلیدی یک متن، غیر از توجه به این که چقدر کلمات انتخابی به تعریف مفهوم اصلی متن می پردازند ، به این مسئله نیز توجه کنیم که این کلمات از نظر یک موتور جستجو چه قدر اهمیت دارند.

[1] Clustering
[2] filtering
[3] Topic detection and tracking
[4] Information visualization
[5] Query

به همین علت فرایند استخراج کلمات کلیدی ارائه شده در این پروژه به میزان اهمیت یک کلمه از نظر موتور جستجو نیز توجه کرده و فرایند ارزیابی کلمات کلیدی استخراج شده را بر پایه نتایج موتورهای جستجو قرار داده است.

4-1 چالش های موجود در پردازش متون فارسی

زبان فارسی که در گستره‌ی جغرافیایی ایران، افغانستان و برخی نقاط تاجیکستان مورد استفاده قرار می-گیرد یک زبان خاورمیانه ایست که در دامنه لغات و حتی گاهی قواعدش متاثر از زبان های دیگر مثل عربی، ترکی ، و بعضی زبان های اروپایی مانند انگلیسی و فرانسوی است. از این رو آنالیز ریخت شناسی این زبان به ارتباط با زبانی ها زیادی نیاز دارد[4]

زبان فارسی از دیدگاه ویژگی های زبانی یک زبان پیوندی و ضمیر انداز است و در شکل رسمی آن ترتیب فاعل – مفعول – فعل برای جملاتش قید می شود این در حالیست که استثنائات مجازی که به خاطر فرایندهایی مثل نامکانی ، بهم ریختگی ، حرکت جهت برجسته سازی ، تاخیر، شکافت و شبه شکافت و غیره بوجود می آیند این زبان را به یک زبان بدون ترتیب نزدیک کرده است[5].

به خاطر تفاوت های ویژه طبیعت این زبان، کار کردن بر روی سیستم های بازیابی اطلاعات در زبان فارسی نیازمند توجهات ویژه ای است که در زبان های دیگر نظیر انگلیسی به این مسئله نیاز نیست[6].

تحقیقاتی که در زمینه پردازش و درک متون فارسی صورت می گیرد هم با مشکلاتی که در سایر زبان ها وجود دارد سروکار دارد هم با مشکلاتی که مخصوص زبان فارسی است. برخی از این مشکلات ناشی از تحقیقات کم در رابطه شناخت قواعد زبان فارسی و برخی دیگر به علت محدودیت های سیستم های هوش مصنوعی است [5] که بطور خلاصه در فهرست زیر ذکر شده است:

1. عدم وجود منابع زبانی مناسب و کافی برای زبان فارسی مانند
   - واژه های تک زبانه و چند زبانه محاسباتی
   - واژه های معنایی و متصل به هستان شناسی[1]
   - هستان شناسی های لغوی
   - هستان شناسی جامع عمومی و تخصصی
   - مجموعه متون[2] عمومی و تخصصی ساده یا برچسب خورده (با برچسب های اجزاء کلام، کسره ی اضافه، نقش های موضوعی، مفاهیم و روابط مفهومی و غیره)
   - مجموعه مدون قوانین ساخت واژه ای و دستوری پوشا
2. ساختارهای متفاوت فایلی اسناد فارسی مانند نسخه های گوناگون Microsoft word ، زرنگار، Pe2 و غیره
3. استانداردهای گوناگون دخیل در کدگذاری اسناد مختلف فارسی: این استانداردها و جزئیات آن ها به حدی متنوع و متفاوت است که بحث در مورد آن ها خسته کننده می نماید. اما به طور عمومی با وجود اشکالاتی که استاندارد[3] نرم افزار هایی مثل Wordو VisualStudio وجود دارد استاندار کدگذاری این نرم افزارها در حال رایج شدن است.
4. عدم وجود استانداردهای شیوه ی نگارش این مسئله منجر به مشکلاتی از قبیل:
   - مشکل تشخیص مرز کلمات به خاطر شیوه های گوناگون ممکن در نوشتار
   - ایجاد تفاوت در صورت های کلمات به خاطر عدم وجود صورت های ممکن برای انواع پسوندها و پیشوندها

[1] Anthology

[2] از مجموعه متون به عنوان ترجمه کلمه corps از عبارت پیکره نیز استفاده می شود.

[3] بزرگترین تفاوت استاندارد این نرم افزارها با استاندارد 3342 موسسه استاندارد ایران رعایت نکردن ترتیب برای چهار حرف مخصوص زبان فارسی («پ»، «ژ»، «گ»، «چ» می باشد.)

5. حذف اطلاعات گویشی مانند مصوت ها که باعث ایجاد مشکلاتی از قبیل:
   - اشکال در تشخیص کسره اضافه: کسره اضافه بزرگترین عامل تشخیص مرز گروه های اسمی است که متاسفانه با حذف مصوت ها، حذف شده و ایجاد مشکل می کند.
   - مسأله ی ابهام در معانی در کلماتی مثل رَوَد و رود می شود.
6. افعال و کلمات مرکب یا کلماتی که هم آیی دارند
7. مسأله ی هم نگاره ها در کلماتی مثل «شیر»
8. معناشناسی و مشکلات تحلیل معنایی

1-5 فرایند کلی استخراج خودکار کلمات کلیدی

به طور کلی تمامی روش های استخراج کلمات کلیدی را می توان بر ساختاری شبیه به شکل 1-1 انطباق داد. (مسیر های نقطه چین به معنای اختیاری بودن ادامه مسیر می باشد)

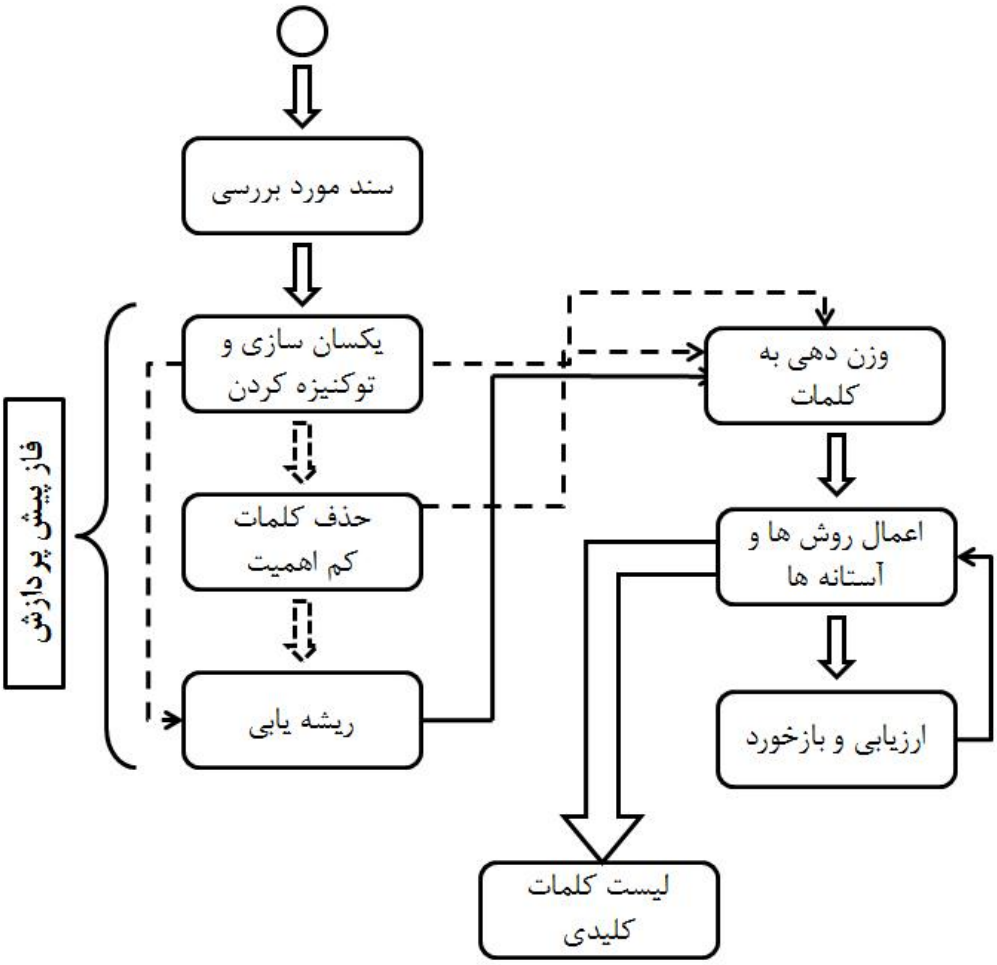
برای استخراج مجموعه کلمات کلیدی ابتدا سند مورد بررسی با استفاده از قواعد احتمال یا علائم نگارشی و یا سایر روش ها به بسته های کلمات یا توکن ها تبدیل می شود سپس کلماتی که معنای خاصی ندارند و بیشتر نقش دستوری دارند یا به خاطر عمومی بودن و تکرار زیادشان تمایزی بین این متن با متون دیگر ایجاد نمی کنند حذف می شوند[1].

البته در بعضی زبان ها مثل زبان فارسی که نوع رمزگذاری ها تفاوت دارد و یا حتی چند شکل درست برای یک ترکیب می‌تواند وجود داشته باشد پیش از هر کاری بهتر است فرایندی به نام یکسان سازی[2]

[1] کلمات مذکور را کلمات پرتکرار می نامند که در فصل دوم و چهارم به طور مفصل مورد بحث قرار می گیرند.

[2] Unification

اجرا شود. فرایند یکسان سازی، حالات متفاوت عبارات یا رمزگذاری های متفاوت را به یک نوع یکسان تبدیل می کند.



شکل 1-1 شمای کلی سیستم استخراج کلمات کلیدی

ریشه یابی[1] در واقع تبدیل کردن مجموعه کلمات با ریشه یکسان به یک کلمه یکسان است. در ریشه یابی هیچ تقیدی برای ریشه قائل نیستیم و اینکه یک ریشه طبق تعاریف رسمی دستور زبان بدست آید یا نه اهمیتی ندارد.

[1] stemming

همانطور که در شکل مشخص است استفاده از مراحل پیش پردازش بسته به نوع روش به کار برده شده است. در واقع در برخی روش ها ریشه یابی انجام نمی شود یا ممکن است برای دقت بالاتر کلمات پرتکرار نیز حذف نشوند.

پس از آنکه لیست کلمات استخراج شد به کلمات وزن داده می شود. روش های وزن دهی به کلمات یکی از بزرگترین متمایز کننده های روش های مختلف محسوب می شود. با اعمال یک آستانه مثلا یک امتیاز خاص یا یک تعداد کلمه خاص لیستی از کلمات کلیدی بدست می آید. به طور معمول این کلمات کلیدی با استفاده از معیارهایی که هر روش تعریف می کند مورد ارزیابی قرار می گیرد تا لیست نهایی استخراج شود. در روش هایی که از یادگیری ماشینی کمک می گیرند معمولا بازخوردی از ارزیابی برای بهینه‌سازی و ارتقا امتیاز کلمات نیز گرفته می شود.

1-5-1 تئوری های مسئله

دمسئله استخراج کلمات کلیدی بر پایه تئوری لان[1] و قانون ZIPF بنا نهاده شده است که به مرور این دو تئوری می پردازیم

تئوری ZIPF: این تئوری بیان می دارد که اگر تعداد وقوع یک کلمه F بوده و در لیست صعودی مرتبه شده کلمات بر حسب تعداد، رتبه r را داشته باشد آنگاه

$$F*r=K$$

[1] Lane

که K یک عدد ثابت است. یعنی اگر کلمه ای مثل «را» رتبه 1 را در لیست تکرار کلمات با 2000 تکرار داشته باشد توقع داریم که کلمه ای مثل «می شود» که رتبه 20 را در لیست تکرار کلمات دارد 100 بار تکرار شده باشد. یک نتیجه واضح این قانون اینست که تعداد کمی از کلمات عمومی اند و زیاد تکرار می-شوند، در حالیکه که تعداد متوسطی از کلمات با تعداد مناسبی تکرار می شوند و تعداد زیادی از کلمات با تکرار بسیار کم وجود دارد.

تئوری لان

این تئوری بیان می دارد که تعداد وقوع یک کلمه در یک متن میزان اهمیت کلمه را نشان می دهد. همچنین میزان اهمیت یک جمله را می توان براساس وضعیت نسبی اهمیت کلمات آن جمله تعیین کرد.

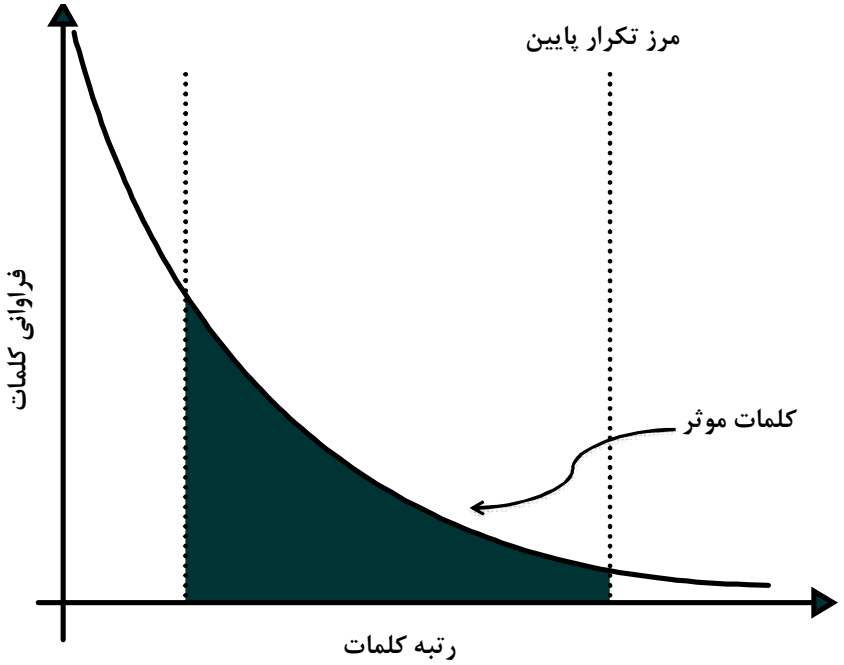
بر اساس این تئوری کلمات تکرار شونده را می توان برای استخراج کلمات و جملات تشریح کننده سند به کار گرفت.

در نمودار شکل 1-2 f تعداد رخداد کلمات مختلف و r رتبه آن است و به نوعی نشان دهنده قانون ZIPF می‌باشد.(رابطه 1-1)

لان با اعمال یک آستانه گیر بالا و پایین بر روی این منحنی، توانست کلمات کم اهمیت را جدا کند.کلماتی که از آستانه ی بالا تجاوز کرده باشند، به عنوان کلمات بی ارزش و پر تکرار درنظر گرفته می‌شوند، و کلماتی که ازآستانه پایین کم تر باشند، کلمات نادر و کمیاب نامیده می شوند که در محتویات مقاله یا سند شرکت چندانی ندارند.

$$F * r \xrightarrow{\approx} cons\tan t \qquad 1\text{-}1$$

همچنین لان با استفاده از این فرض که کلمات پر تکرار بین این دو مرز بیشترین تفکیک را بین سند موجود و سایر اسناد ایجاد می کند به این نتیجه رسید که ارزش کلمات از میانه به سمت مرزها کمتر شده تا به صفر می رسد. البته آستانه‌های تعریف شده با استفاده از سعی و خطا بدست آمده و هیچ پیش گویی نمی توان در آن مورد انجام داد. اما جالب اینجاست که این ایده، پایه اکثر کارهای بعدی بازیابی اطلاعات شده است همانطور که خود لان نیز از آن برای ایجاد یک چکیده نویس خودکار استفاده کرده است.



شکل 1-2 نمودار تعداد رخداد کلمات بر اساس رتبه آن ها (تئوری لان)

فصل دوم:

مروری بر کارهای گذشته

2-1 مقدمه

همانطور که در فصل اول بصورتی کوتاه بررسی کردیم، استخراج کلمات کلیدی شامل دو بخش پیش پردازش و انتخاب کلمات است.

روش های ارائه شده برای انتخاب کلمات کلیدی را می توان از جنبه های مختلفی تقسیم بندی کرد:

به عنوان مثال مانند تمامی روش های هوش مصنوعی می توان برای فرایند انتخاب خودکار کلمات کلیدی حالت با ناظر و بدون ناظر در نظر گرفت به این معنا که اگر برای هر سند مورد بررسی یک مجموعه کلمات کلیدی مشخص کنیم و سیستم را بر اساس این کلمات مشخص شده آموزش دهیم سیستم یک سیستم یادگیری با ناظر خواهد بود و در غیر اینصورت سیستم یک سیستم یادگیری بدون ناظر بشمار می آید.

همچنین می توان روش‌ها را برپایه نوع وزن دهی به آماری و غیر آماری تقسیم کرد.

اما تقسیم بندی از نظر نوع تکنیک که در بخش 2-4 به آن می پردازیم جامع ترین نوع تقسیم بندی در بسیاری از جهات محسوب می شود

در این فصل به ارائه خلاصه ای از کارهای انجام شده مرتبط با تحقیق می پردازیم این فصل شامل سه بخش پایگاه های داده، فرایندهای پیش پردازش و روش های استخراج کلمات کلیدی است. اگرچه تاکید ما در این فصل بر روش های استفاده شده بر روی متون فارسی است اما مروری بر تحقیقاتی که نتایج یا بخشی از روش های آن ها نیز در پروژه آمده است نیز صورت گرفته است.

2-2 پایگاه های داده

وجود یک مجموعه متنی استاندارد پیش نیاز هر گونه تحقیق بر روی ساختارهای بازیابی اطلاعات محسوب می‌باشد. این در حالیست که کارهایی که اخیرا در بدست آوردن اطلاعات از زبان فارسی شده معمولا از مشکل فقدان مجموعه متون استاندارد رنج می برد در این بخش از فصل دوم به کارهای صورت گرفته بر روی جمع آوری متون استاندارد به عنوان پایگاه داده ای برای استفاده در تحقیقات بازیابی اطلاعات می‌پردازیم.

پایگاه داده مطرح شده در [7] حاوی 25 مگابایت از اسناد مجلس است و به علت موضوعیت تک بعدی و سایز کم می تواند نتایج کار را تحت الشعاع قرار دهد.

مجموعه متنی شیراز [8] شامل 10 مگابایت متون دو زبانی است که برای تست ترجمه ماشینی در دانشگاه نیو مکزیکو طراحی شده است.

پایگاه داده FLDB[1] ساخته شده از منتخبی از متون فارسی امروزی رسمی و غیر رسمی است و یک سری ورودی های دیکشنری و لیستی شامل حدود 3 میلیون عدد از کلمات است[9]. اگرچه بزرگی FLDB در متون خوب ساختیافته فارسی خوب است اما به اندازه کافی اطلاعات مناسب برای سیستم های بازیابی اطلاعات ندارد

یکی دیگری از مجموعه سند های فارسی محک است[10]. این مجموعه برای ارزیابی روش های استخراج اطلاعات ایجاد می شود اما دارای 3007 سند است که آن را برای سیستم های بزرگ استخراج اطلاعات نامناسب می سازد.

[1] Farsi Language Data Base

همشهری [11] پایگاه داده ایست که بر اساس مقالات روزنامه با توجه به مشخصات دقیق TREC[1] درست شده است. به علاوه اطلاعات آماری در مورد اسناد و جستجوهایی مربوط به آن ها در این مقاله آمده است. این مجموعه بزرگترین مجموعه زبان فارسی جمع آوری شده تا کنون می باشد. پایگاه داده کنونی همشهری 1600000 سند دارد و به همراه 65 جستجو و نتایج تفضیلی آن برای استفاده محققین بر روی وب [2]بصورت رایگان قرار گرفته است.

جامعه آماری مناسب در موضوعات مختلف و نیز ساختار مناسب ارائه باعث شد (بصورت XML) که مجموعه همشهری برای ساخت پایگاه داده خاص این پروژه استفاده شود. که در فصل چهارم به این موضوع می پردازیم.

3-2 فرایندهای پیش پردازش

همانطور که در شکل 1-1 نشان داده شده معمولا پیش از اعمال روش های استخراج کلمات کلیدی فرایندهایی برای پیش پردازش متن و ایجاد زمینه برای کسب نتایج بهتر صورت می گیرد.

به طور کلی انواع پیش پردازش را به چهار دسته تقسیم بندی کرد:

1. یکسان سازی
2. توکنیزه سازی (تشخیص مرز کلمات و جملات)
3. حذف کلمات پر تکرار

[1] TREC(Text REtrieval Confrence)کنفرانس بین المللی بازیابی اطلاعات و پردازش زبان های طبیعی که به منظور ارزیابی سیستم های بازیابی اطلاعات برگزار می شود.

[2] http://ece.ut.ac.ir/DBRG/hamshahri/

4. ریشه یابی

کارایی روش های پیش پردازش می تواند به دو صورت مطلق و در خلال فرایند کلی بررسی شود

سنجش کارایی پیش پردازش بصورت مطلق: در این حالت یک فرایند با ارائه تابع مناسب و قابل درک برای سیستم های انسانی نتایج پیش پردازش را مورد بررسی قرار می دهد به عنوان مثال فرض کنید یک سیستم پیش پردازش بدون استفاده از فهرست ثابت کلمات پرتکرار به شناسای و حذف پویای آنان می‌پردازد در این صورت نسبت تعداد کلماتی که یک کارشناس آن‌ها را در متن بی‌ارزش و پرتکرار می‌داند و از متن حذف نشده باشد به تعداد کلماتی که حذف شده اند می تواند نسبت خطا را تعیین کند.

سنجش کارایی پیش پردازش در خلال فرایند کلی: معمولا زمانی که تابع ارزیابی مناسبی برای یک فرایند پیش پردازش قابل تعریف نباشد می توان میزان کارایی پیش پردازش را براساس نتایج فرایند کلی در حضور و عدم حضور روش بررسی کرد. به عنوان مثال فرض کنید در حالت قبل بحث داشتن کارشناس برای کلمات بی‌ارزش مسئله را به داوری انسانی مرتبط می‌کند که هم خطاپذیر و هم هزینه‌بر است در این صورت میزان موفقیت روش با استفاده از فرایند پیش پردازش نسبت به میزان موفقیت روش در عدم استفاده از فرایند پیش پردازش می تواند ملاک خوبی برای سنجش کارایی روش باشد.

2-3-1 یکسان سازی

در زبان فارسی به علت شرایط خاصی که وجود دارد گاهی با وجود آنکه دو واژه کاملا یکسان هستند، به علت نوع رمزگذاری نویسه ها یا تعدد حالت های نوشتاری ممکن، از دید یک پردازشگر متن دو واژه

متفاوت تلقی می شوند. به عنوان مثال، پیشوند «می» و پسوند «ها» در ابتدا و انتهای واژه‌ها، ممکن است به سه صورت مختلف دیده شوند:

چسبان	جدا با فاصله	جدا بدون فاصله
کتابها	کتاب ها	کتاب‌ها
میرود	می رود	می‌رود

و یا واژه‌ها «مسئول» و «مجموعهٔ» به صورت‌های زیر در اسناد مختلف دیده می شوند:

مسئول	مسؤول	مسوول
مجموعه ی	مجموعهٔ	مجموعه

یا حروف «ی» و «ک» که در متون مختلف ممکن است داری شکل یکسان اما رمزگذاری متفاوت باشند. برای جلوگیری از بروز این مشکل، نیازمند یکسان سازی متون مورد پردازش هستیم.

به عنوان یک مثال از یکسان سازی، در مورد پسوند «ها» می‌توان با تبدیل شکل‌های «جدا با فاصله» و «جدا بدون فاصله» به شکل چسبان تمامی اشکال ها را به حالتی یکسان تبدیل کرد.

پروسه یکسان سازی دارای دو بخش است:

2-3-1-1 یکسان سازی تکنیکی

این پروسه شامل یکسان سازی نوع رمزگذاری بر اساس یک استاندارد مشخص و همچنین یکسان سازی با یک ساختار فایلی یکسان است. انواع استاندارد برای رمز گذاری نویسه ها وجود دارد که برخی از آن ها

توسط نرم افزارهایی کاربردی انتخاب شده و برخی دیگر توسط موسسه استاندارد. اگرچه انتخاب موسسه استاندارد از نظر رسمی بودن اعتبار خوبی دارد اما،در واقع نمی توان گفت کدام استاندارد بر آن یکی ترجیح دارد چراکه به هرحال با وجود تمامی مشکلات استاندارد انتخاب شده توسط نرم افزارهای معتبر کارایی این نرم افزارها در کفه ی ارزیابی سنگینی زیادی نشان می دهد.

اما در بحث یکسان سازی ساختار فایل باید گفت با وجود قالب های گوناگون رایج مورد استفاده در زبان فارسی بهترین قالب نمایش XHTML است. این قالب به علت آنکه از سوی وب مورد پشتیبانی قرار می-گیرد و همچنین مستقل از سیستم عامل است مناسب ترین قالب به شمار می رود. اما به هرحال برای یکسان سازی باید برنامه ای برای تبدیل قالب های دیگر به XHTML ، یا پیدا کرد یا نوشت.

2-3-1-2 یکسان سازی بر اساس دانش زبان

این یکسان سازی شامل چگونگی نوشتن‌هاست . این مسئله که «ها» علامت جمع به کلمه بچسبد یا نه و یا آنکه "ی" در پایان کلماتی که به «ه» ختم می شوند چگونه باید بیاید در این یکسان سازی حل می-شود. اگرچه استفاده از قواعد استاندارد و رایج برای یکسان سازی ارجح است اما در صورتیکه سیستم یکسان سازی شما به عنوان یک بخش داخلی از یک کل به شمار می رود می توانید از هر قاعده ای استفاده کنید[1].

در مرجع [12] مجموعه‌ای نسبتا جامع و موثر از تبدیل ها برای یکسان سازی متون فارسی ارایه شده است. نکته‌ای که در یکسان سازی باید دانست آنست که هرچه فرایند یکسان سازی جامع تر و مفصل تر باشد نتیجه کار بهتر است و استثنائات بیشتر تحت پوشش قرار می گیرند اما در مقابل پیچیدگی زمانی مسئله بالاتر می رود.

[1] فرایند یکسان سازی در صورت استفاده از قواعد استاندارد نرمال سازی (normalization) نیز نامیده می شود.

2-3-2 تشخیص مرز واژه ها و جملات یا توکنیزه سازی

توکنیزه سازی یا تقسیم کردن رشته های کاراکتری متن به کلمات یکی از فرایندهای پیش پردازش اصلی در اکثر روش ها به شمار می رود. همچنین از آنجا که برخی از روش های امتیاز دهی به کلمات مورد سیستم های استخراج کلمات کلیدی، از مکان کلمه در جمله یا مکان جمله نسبت به متن نیز بهره می-گیرند فرایند تعیین مرز جملات نیز گاهی به عنوان یک فرایند پیش پردازش در نظر گرفته می شود.

2-3-2-1 تعیین مرز جملات

در بیشتر مواقع، تعیین مرز جمله‌ها از طریق بررسی علایم جداکننده انجام می‌شود. علایمی که برای تعیین مرز جمله از آنها استفاده می‌شود، عبارتند از: ".“، ”؛“، ”!“، ”؟“، ”?“ و ”:“. باید توجه داشت که جستجو برای یافتن این علایم به تنهایی کافی نیست و در برخی موارد مشکلاتی را به همراه دارد. به عنوان مثال، در زبان فارسی بعضی از واژه‌های اختصاری به صورت چند حرف که با نقطه از هم جدا شده‌اند، ظاهر می‌شوند (مانند «ه.ق.» که مخفف «هجری قمری» است). در صورتی که تعداد این حالات مشکل‌ساز در پیکره زیاد باشد، باید از روش‌های پیچیده‌تری استفاده شود، که قادر به شناسایی این موارد باشند. همچنین ممکن است در متونی که بصورت غیر رسمی نوشته شده یا قواعد نوشتاری در آن ها رعایت نشده (این امر در متون تحت وب اتفاقی دور از انتظار نیست.) باید با استفاده از ساختارهای نحوی زبان جملات را مشخص کرد. لیست زیر تعدادی از این موارد را نشان می‌دهد..

1. در آدرس های وب علائم «.» به کار می رود.
2. علائم اختصاری مانند« ق.م» به معنای قبل از میلاد که البته منجر به اشکال در تعیین مرز جملات نیز می شوند.

D.J.& J. H.Martin در کتاب خود پیشنهاد می‌کند که برای حل مسئله ابهام نقطه پایان یا نقطه درون یک کلمه از کلاسبند دودویی[1] به عنوان یک راه حل رایج استفاده شود[13]. به این ترتیب که برای تمامی نقطه‌های یک متن بردار خصوصیات تعیین و سپس با استفاده از یک سیستم یادگیری ماشین با آموزش تفاوت نقاط پایان جمله با سایر نقاط این تعیین تفاوت را به یک کلاسبند تشخیص دهنده دو کلاس مذکور[2] واگذار می‌گردد. همچنین الگوریتم ساده ای را نیز به عنوان یک راه‌حل با صحت عملکرد نسبتا مناسب برای زبان انگلیسی ارائه کرده است که در بخش فرایندهای پیش پردازش با اعمال یکسری تغییرات ساده برای زبان فارسی آن را به کارگرفته ایم.

2-2-3-2 تعیین مرز کلمات

کلمه به عنوان کوچکترین واحد معنادار متن در اکثر تحقیقات به عنوان پایه ارزیابی ها و نتیجه گیری ها قرار می گیرد. به طور مرسوم کلمات با استفاده از علائم نوشتاری از قبیل فضای خالی، علامت پرش، علامت خط جدید، « ، »، « , »، «>»، «<»، «]»، «[»، «- »، «_» و «/ »شناسایی شده و از یکدیگر تفکیک می‌شوند. اما در زبان نوشتاری فارسی به علت ویژگی های خاصی که وجود دارد، تنها به این علائم تکیه کردن کافی نیست چراکه در این صورت ممکن است بخش هایی از یک کلمه، به عنوان کلمه ای مستقل و ترکیب چند کلمه مستقل، به عنوان یک کلمه واحد در خروجی بدست آید. لیستی از مواردی که در این تحقیق با آن مواجه شده‌ایم بصورت زیر است:

1. **کلماتی شامل حروف «ر»، «ژ»، «ز»، «د»، «ذ»، «ا» ، «و»**

   از آنجا که این حروف از سمت چپ به حرف دیگری نمی چسبند در دو حالت منجر به اشتباهاتی توسط نگارنده می شوند:

[1] Binary classifier

[2] نقاط پایان جمله و کلاس نقاط با کابردی غیر از پایان جمله

أ. حضور در انتهای کلمه: به خاطر فاصله ای که بصورت ذاتی با کلمه بعد ایجاد می کنند گاهی باعث می شونده نگارنده تایپ فاصله بین کلمات را از یاد ببرد.

به عنوان مثال عبارت: «اصرار بر جستجو» ، «اصراربرجستجو» اگرچه در هر دو حالت به راحتی خوانده می‌شود اما حالت دوم از دید یک پردازشگر متنی کلمه‌ای 12 حرفی محسوب می شود.

ب. حضور در میان کلمه: منجر به عدم تشخیص فاصله اشتباها تایپ شده میان کلمات توسط نگارنده می شوند.

به عنوان مثال: «تابناک»، «تابنا ک»

2. افعال و کلمات چند بخشی

زبان فارسی مملو از کلمات و ترکیباتیست که با وجود یکپارچگی در معنا با استفاده از علامت فاصله به چند بخش تقسیم می شوند در ادامه برخی از این نوع کلمات فهرست شده است:

أ. کلمه هایی که دارای صفت جمع مثل «ها»، «ان» و غیره هستند.

ب. افعال مضارع که از کلمه «می» استفاده می کنند.

ج. افعال مرکب از قبیل «آتش زدن»

د. کلمات چند بخشی از قبیل «ساده زیست»

ه. کلمات مرکبی که با یک دیگر هم آیی دارند: از قبیل «امام خمینی» یا «محمود فرشچیان».

3. نشانه ی دستوری / و «،»

اگرچه نشانه دستوری «/» در زبان فارسی به عنوان کلمه جانشین به کار می رود و دو کلمه را جایگزن هم معرفی می‌کند اما در موارد دیگر معنای متفاوتی نیز دارد. (مثلا به جای عبارت چه مرد چه زن در یک فرم شکل مرد/زن به کار برده می شود.) به عنوان نمونه:

أ. تاریخ در فارسی با علامت « / » مشخص می شود.

ب. در آدرس های کامپیوتری علامت «/» به کار می‌رود.

مشکل «،» در مورد اعداد در صورتی که ارقام آن ها با علامت «،» از هم جدا شود اتفاق می‌افتد مانند 999،222 در اینصورت تقسیم این عدد به دو عدد کوچتر معنای عدد را بهم می‌ریزد

برای حل مشکل واژه‌های مرکب روش‌های مختلفی پشنهاد شده است. در مرجع [14] یک روش آماری مبتنی بر پیکره برای تعیین واژه‌هایی که باهم‌آیی دارند، معرفی شده است. در این روش، با توجه به فراوانی رخداد دو واژه به صورت متوالی و هم چنین فراوانی هر کدام به تنهایی، تصمیم‌گیری می‌شود که آیا دو واژه باهم‌آیی معنی‌داری دارند یا خیر. باید توجه داشت که باهم‌آیی دو واژه لزوما به معنای آن نیست که آن دو واژه با هم یک واژه مرکب را تشکیل می‌دهند. تجربه نشان می‌دهد که در بیشتر کاربردهای پردازش متن، در نظر گرفتن واژه‌هایی که باهم‌آیی دارند به عنوان یک واژه واحد، مفید واقع می‌شود.

به طور کلی از فعالیت های انجام شده در زمینه تعیین مرز کلمات و گروه‌های کلمات می توان به مرجع [15] اشاره نمود که به تشخیص انتهای کلمات و فاصله گذاری میان آن ها می پردازد. هم چنین مرجع [16] در مطالعه ای به بررسی نحوه ی تشخیص کسره‌ی اضافه در متن با استفاده از روش های آماری مبتنی بر پیکره های زبانی پرداخته است . تشخیص کسره اضافه محذوف کمک بسیاری به حل مشکل ابهام در شناسایی مرز گروه های اسمی می نماید. از سوی دیگر در بعضی کارها این مرحله با مراحل دیگر

ادغام و یا در طی مراحل دیگر انجام می شود. مثلا در برخی تحلیل گر ساخت واژی معرفی شده و در حین تحلیل ساخت واژی، مرز کلمات نیز تعیین و فاصله های زائد درون کلمات حذف می شوند.

مقاله [17] با استفاده از یک روش آماری مبتنی بر پیکره برای شناسایی واژه‌های به هم چسبیده معرفی شده است. این روش، با استفاده از معیار احتمال شرطی متقارن تعیین می‌کند که آیا یک عنصر متنی، یک واژه‌ی کامل است یا مجموعه‌ای از واژه‌های به هم چسبیده است. روش پیشنهادی روی یک مجموعه‌ی 18000 تایی از اخبار ورزشی مورد آزمایش قرار گرفته و تعداد نمونه‌هایی که به درستی از هم جدا شدند به کل نمونه‌های جداشده نسبت قابل قبولی (بیش از 80 درصد در مورد نمونه‌های جداشده به دو بخش) را نشان داده است.

3-3-2 حذف کلمات پر تکرار

کلمات پر تکرار در یک فرایند متن کاوی به کلماتی تلقی می‌شود که در تمامی متن‌ها به کرات یافت می-شوند و معمولا معنی مستقل خاصی نداشته یا برای ایجاد قواعد دستوری به کار می‌روند. به طور مثال کلماتی مثل "را" "در" "به" یا افعالی مثل "است" و "می‌شود" یا در زبان انگلیسی کلماتی شبیه به

A, ABOVE, ACROSS, AFTER, AGAIN, AGAINST, ALL, ALMOST, ALONE

را می‌توان جزو کلمات پر تکرار محسوب کرد.

عدم حذف کلمات پر تکرار در یک متن به علت فرکانس بالای این کلمات منجر به کم اثر شدن کلمات غیرمعمول با فرکانس نسبتا خوب در متن می شود. از طرف دیگر با توجه به این نکته که کلمات مذکور فارغ از زبان مورد بررسی ، حجم بزرگی از یک متن را تشکیل می دهند حذف این کلمات تاثیر مقید زیادی بر کارایی سیستم چه از نظر پچیدگی های زمانی و چه از نظر حافظه داشته باشد[18].

از جمله کارهایی که در زمینه حذف کلمات پر تکرار شده است می توان به کار تقوا [19]که در آن با اشاره به انواع فعل های بی ارزش مثل «بود» پرداخته است یا مرجع [20] که با استفاده از روش های آماری و تحلیلی لیستی 928 کلمه ای از کلمات پر تکرار فارسی را ارائه داده است اشاره کرد. همچنین مقاله ذکر شده در مرجع [21] به این مسئله پرداخته است که حذف کلمات پر تکرار فارغ از دامنه می تواند مشکلاتی را ایجاد کند. این مشکل در مرجع [20] نیز به نوعی بررسی شده است و نشان داده شده که مثلا در تحقیقی که برروی پایگاه داده همشهری انجام شد کلمه با ارزشی مثل «ایران» نیز در صورت نگاه آماری صرف جزو کلمات پر تکرار تلقی می‌شود.

در پایان شایان ذکر است که البته گاهی بعضی استثنائات در مسئله حذف کلمات کلیدی مشکلاتی را به وجود می‌آورد. مثلا تا چند سال قبل اگر جمله مشهور شکسپیر "be or not to be" را در موتورهای جستجو وارد می‌کردید نتیجه جستجو این بود "لغات استفاده شده شما به علت تکرار زیاد، توسط موتور جستجو حذف شده است" و این نشان می‌دهد که برخورد فارغ از معنی با کلمات پر تکرار می‌تواند مشکل ساز باشد اگرچه در اکثر تحقیقات این طور استثنائات قربانی افزودن بر سرعت و کمتر شدن پیچیدگی خواهند شد اما در این پروژه همانگونه که در بخش 3-5 خواهیم دید با اضافه کردن این قید که «حذف کلمات پرتکرار به شرط عدم حضور در یک توالی بیش از سه کلمه ای از کلمات پرتکرار، قابل انجام است.» از استثنائاتی به این شکل تا حد زیادی جلوگیری شده است.

2-3-4 ریشه یابی کلمات

ریشه یابی یکی از کارهای رایج در استخراج کلمات کلیدی است چرا که باعث انتخاب کلماتی می‌شود که به اشکال متفاوت تکرار زیادی داشته اند. هدف از ریشه یابی حذف اضافات از کلمه و رسیدن به ریشه ای

یکسان برای کلمات هم ریشه است. اکثر ریشه یاب هایی که برای فارسی پیشنهاد شده از روش های مبتنی بر حذف پسوندها و پیشوندها استفاده می کنند. در این قسمت ابتدا به بررسی مشکلات ریشه یابی در زبان فارسی و همچنین راه حل های محتمل برای آنان می پردازیم سپس به برخی از کارهایی که تاکنون در زمینه ریشه یابی انجام شده اند اشاره می کنیم.

2-3-4-1 دشواری های ریشه یابی در زبان فارسی

1. مشکلات ناشی از خطاهای یکسان سازی

- مشکلات ناشی از مسائلی که استاندارد یکسانی ندارند مانند «ها»ی نشانه جمع یا «ی» بعد از «ه» (مانند کتیبه ی)
- مشکلات ناشی از فاصله گذاری ها که در تعیین مرز جملات و کلمات نیز به آن اشاره شد.

واضح است که برای حل این مشکلات باید هرچه بیشتر یکسان سازی را دقیقتر و جامع تر انجام داد.

2. کلمات مرکب: در زمینه کلمات مرکب (به ویژه فعل مرکب که تاثیر بیشتری در فرایندهای ریشه-یابی دارد.) با وجود کوشش های فراوان هنوز ابهامات و ناهماهنگی های فراوانی وجود دارد که به نظر می رسد برای رفع آن لازم است فهرست کاملی از افعال مرکب به همراه شکل پیشنهادی آن از سوی فرهنگستان ارائه شود.
3. دگرگونی در کلمه ها هنگام پیوند:

برخی از کلمات فارسی در هنگام ادغام با پسوندها و پیشوندها دچار تغییر شکل می شوند. در زمان ریشه یابی لازم است که برای این حالات (یا حداقل برای حالاتی که بیشتر دیده می شوند) نیز راه حلی ارائه شود یا بصورت لیستی از استثنائات در برنامه ریشه یاب منظور شوند.

جدول 2-1 نمونه ای از این دگرگونی ها را نمایش می دهد.

جدول 2-1 نمونه ای از دگرگونی کلمات در هنگام پیوند

زنده + ان ← زندگان	زنده + م ← زنده ام
گو+م ←گویم	ن + افتاد ←نیفتاد
ن + آزما ← نیازما	زنده+ها← زنده ها

4. کلمات زبان های بیگانه با ساختار ظاهری شبیه به کلمات فارسی

در مجموع واژگان فارسی کلمات زیادی از زبان های دیگر وجود دارد که ساختار ظاهری آنان شبیه به کلمات فارسی است. مثلا کلمه رئالیست در یک نگاه رشته ای صرف می تواند بصورت ساختار «رئالی + ست» دیده شود که در این صورت با یک ریشه یابی اشتباه حروف «ست» که بخش از اصل کلمه هستند به عنوان پسوند حذف خواهند شد.

5. شناسایی ریشه فعل ها

یکی از ابتدایی‌ترین مشکلات در زمینه شناسایی ریشه فعل ها خود مسئله شناسایی فعل است. در واقع در زبان فارسی فرایند تعیین بخش های سخن[1] فرایند مشکلی است. به عنوان مثال قواعدی مثل «فعل شناسه می گیرد» نمی تواند پاسخ گوی تشخیص نیاز به تشخیص فعل ها از اسامی باشد چراکه گاهی اسامی و صفات نیز شناسه می گیرند. به عنوان مثال کلمه «خوشحالیم» به جای «خوشحال هستیم» به کار گرفته می شود.

[1] Part of speech

«در فعل های ساده پس از حذف «َن» از مصدر، بن ماضی باقی می مند و از جهت تغییری که از بن ماضی به مضارع انجام می گیرد، آن‌ها را می توان در هشت گروه جای داد. جدول 2-2 تغییرهای فعل ها را درگروه های هشت گانه نشان می دهد.»[22]

تعداد کمی از فعل های فارسی از قاعده های این هشت گروه پیروی نمی کنند . بن گذشته و غیرگذشته این فعلها جداگانه نوشته شدند. مرجع [23] با نقل جدول 2-2 ذکر می کند که این روش بر مجموعه بزرگی از کلمات به کارگرفته شده و بن گذشته و غیر گذشته بیش تر فعل های ساده ی فارسی به خوبی شناسایی گردیده اند.

جدول 2-2 هشت گروه فعل های فارسی

گروه	حروف پایانی بن ماضی +پسوند	حروف پایانی بن ماضی	نحوه ساخت بن مضارع	مثال	
1	یدن	ـید	پس از حذف «ـید» باقیمانده بن مضارع است	نال + ید + ـَن	نالیدن نال نالید
2	دن	د	پس از حذف «د» باقیمانده بن مضارع است	خور+ د ـَ +ن	خوردن خور خورد

3	ودن	ـ ود	پس از حذف «و»، «و» به «ا» تبدیل می شود	آزمو+د ـَ + ن	آزمودن آزما آزمود
4	ادن	ـ اد	پس از حذف «اد» باقیمانده بن مضارع خواهد بود	افت + اد ـَ +ن	افتادن افت افتاد
5	ختن	ـ خت	«ت» حذف و «خ» تبدیل به «ز» می شود	ساخ + ت ـَ +ن	ساختن ساز ساخت
6	ستن	ـ ست	پس از حذف «ست» باقیمانده بن مضارع است	آرا + ست + ـَ+ ن	آراستن آرا آراست
7	شتن	ـ شت	پس از حذف «ت»، «ش» تبدیل به «ر» می شود	کاش + ت ـَ + ن	کاشتن کار کاش

8	فتن	ـ فت	پس از حذف «ت»، «ف» تبدیل به «ب» می شود	تاف+ ت ـَ + ن	تافتن تاب تافت

2-3-4-2 روش های ریشه یابی

روش های ریشه یابی را می توان به دو دسته کلی تقسیم کرد:

روش های ساختاری که با توجه به ساختار کلمات و قواعد زبان کار می کنند. این روش ها با فرمالیزه-کردن قواعد زبان امکان استفاده از ماشین برای تشخیص ریشه یک کلمه در زبان را فراهم می کنند. معمولا در این روش ها یک ماشین پذیرنده متناهی[1] ساخته می شود و وندهای مختلف مسیرهای موجود در این ماشین را ایجاد می کنند. دو اشکال بزرگ برای این روش ها وجود دارد

1. روش های ساختاری نیازمند دانش کافی در مورد ساختار زبان است.
2. روش های ساختاری در صورت نیاز به تغییر نیاز به کد نویسی مجدد دارند و معمولا در این روش ها نگهداری و به گسترش برنامه هزینه بر خواهد بود

روش های غیر ساختاری: در این روش ها یک مجموعه بزرگ از کلمات با انواع ساخت ها جمع آوری می-شود و واضح است که عملکرد ریشه یاب وابستگی مستقیم به جامعیت این مجموعه کلمات دارد. در این روش با استفاده از تحلیل های آماری وندهایی که در کلمات تکرار شده اند شناسایی می شوند. در این روش ها با استفاده از دو مرحله آموزش و ریشه یابی با استفاده از مرحله آموزش سعی می شود از برخورد

[1] DFA: Deterministic Finite Accepters

مستقیم با قواعد زبانی جلوگیری شود. این روش ها به علت غیر ساختاری بودن و وابستگی بسیار کم به دانش زبانی در سیستم های چند زبانه کاربرد بهتری دارند همچنین بطور عمومی با توجه به پیچیدگی های طراحی کمتر از نتایج مطلوبی برخوردارند.

اما روش های غیر ساختاری از دو مشکل بزرگ رنج می برند:

2. با وجود پیچدگی های کم تر طراحی پیچیدگی زمانی و حافظه ای بیشتری دارند
3. مجموعه کلمات فارسی برای این روش ها وجود ندارد

با توجه به اینکه ریشه یابی فرایندی است که در هر دو روش تا حدودی از زبان مورد تحقیق تاثیر می-پذیرد، کارهای انجام شده در این زمینه را به دو بخش زبان انگلیسی و زبان فارسی تقسیم کرده ایم

2-3-4-3 ریشه یابی برای زبان انگلیسی

در راه حل های ساختاری الگوریتم Porter [24] معروفترین الگوریتم ریشه یابی انگلیسی است. روش Porter بر پایه زبان شناسی و دسته بندی کلمه ها به کمک واج ها و هجاها بنا نهاده شده است. پس از آن وندهای کلمات درون مجموعه کلمات بطور خودکار برداشته می شوند. از الگوریتم های خوب دیگر در این زمینه می توان به کار B. Lovins [25] نیز اشاره کرد.

در میان الگوریتم های غیرساختاری که به نوعی الگوریتم های ریشه یابی جدیدی محسوب می‌شوند الگوریتم Bacchine [26][27] را می توان یکی از کاملترین الگوریتم‌ها بر شمرد.

2-3-4-4 ریشه یابی برای فارسی

از آنجا که تفسیر علمی گرامر و قواعد فارسی قدمت چندانی ندارد، سیستم های ریشه یابی فارسی نیز معمولا کارهایی تازه هستند و با توجه که بیشتر در روش های ساختاری به زبان فارسی پرداخته شده است اشکالاتی نیز در این تحقیقات وجود دارد.

ریشه یاب «بن» [28] الگوریتمی برای ریشه یابی در زبان فارسی با استفاده از الگوریتمی شبیه به الگوریتم ریشه یاب Porter ارائه کرده است. در این الگوریتم پایان کلمات به عنوان وند در نظر گرفته شده است و در یک جدول که در حقیقت کار یک DFA[1] را بر عهده دارد به جستجوی آن می پردازد و پس از یافتن پسوند احتمالی کلمه آن را حذف می کند. در صورتیکه پایان یک کلمه با دو پسوند همخوانی داشته باشد پسوند بزرگتر به عنوان پسوند انتخاب و حذف می شود. همچنین در این تحقیق حد سه نویسه برای حداقل طول کلمات بدون پسوند در نظر گرفته شده است تا از تخریب کلماتی که حروف پایانی شبیه به پسوند دارند جلوگیری کند. (به عنوان مثال کلمه «تاوان» در صورت حذف «ان» به عنوان پسوند جمع به کلمه ای بی‌معنی تبدیل می‌شود.)

تقوا [4] در روشی کاملا یکسان با [28] به طراحی یک DFA برای ریشه یابی کلمات فارسی پرداخته است. DFA طراحی شده در این تحقیق از 70 حالت برای حل مسئله کمک می گیرد و بسیاری از پسوندها را شامل می شود.

از دیگر تحقیقات ریشه یابی کلمات که پایه آن DFA است می توان به مرجع[29] اشاره کرد.

اما در مرجع [30] در روشی تازه با استفاده از الگوریتمی که بر پایه الگوریتم Bacchine بنا نهاده شده است به ریشه یابی کلمات فارسی با استفاده از روش های آماری پرداخته است. مجموعه مستندات برای

[1] Deterministic Finite Accepters

آموزش در این سیستم از سایت خبرگذاری دانشجویان ایران ISNA انتخاب شده است. سیستم ارائه شده در این مقاله با بررسی این مستندات به عنوان مجموعه آموزش لیستی از اشتقاق ها و پسوندها و پیشوندها را برای خود می سازد و به هرکدام از آن ها یک وزن نسبت می دهد. در هنگام ریشه یابی برای یک کلمه حالات مختلفی از تقسیم آن کلمه در نظر گرفته می شود و با توجه به لیست های گفته شده بهترین ریشه برای آن استخراج می شود. نتایج الگوریتم طبق ادعای نویسنده با نتایج سایر روش ها برابری می‌کند.

2-4 روش های استخراج کلمات کلیدی

در این بخش از فصل دوم به بررسی روش های استخراج کلمات کلیدی می‌پردازیم .

2-4-1 تقسیم بندی تکنیکی روش ها

از نظر تکنیکی روش های استخراج کلمات کلیدی را می توان به چهار بخش تقسیم کرد:

2-4-1-1 روش های آماری

این روش ها نیازی به مجموعه داده های آموزش ندارند و فقط با استفاده از اطلاعات آماری کلمات به تشخیص کلمات کلیدی در یک سند می پردازند.

مرجع [31] به عنوان یک روش آماری با استفاده از N-gram ها به شاخص گذاری اتوماتیک پرداخته است. N-gramها یک زیرتوالی n عنصری از یک توالی داده شده محسوب می شوند که بسته به کاربرد عناصر زیر توالی درخواست شده می توانند پدیده ها، حروف، واج ها، یا کلمات باشند. مدل n-gram یک مدل احتمالی برای پیش بینی عنصر بعدی در یک توالی به شمار می رود که در تحلیل آماری NLP و

توالی های ژنتیک کاربرد فراوان دارد. اگر سایز n برابر با 1 باشد Unigram، برابر با 2 Bigram، درحالت سه عنصری trigram و برای سایر حالات همان n-gram خوانده می شود. بعضی از مدل های زبانی ساخته شده از روی n-gram به عنوان مرتبه n-1 مدل مارکوف شناخته می شوند.

مرجع[32] با استفاده از هم‌آیی[1] کلمات یا اصطلاحا کلمات هم رخداد عمل استخراج کلمات کلیدی را انجام داده است.

همچنین مرجع [33] از PAT-tree[2] برای استخراج کلمات کلیدی بهره گرفته است. PAT-tree یک ساختمان داده بسیار قدرتمند است که بطور گسترده در تحقیقات بازیابی اطلاعات مورد استفاده قرار گرفته است. به طور مفهومی این ساختمان داده با یک درخت جستجوی رقمی[3] برابر می کند اما نوعا ساختار ساده تری نسبت به یک درخت جستجوی رقمی دارد.

2-4-1-2 روش های مبتنی بر ساختارهای زبانی

این روش ها کلمات کلیدی را بر استفاده از خصوصیات زبانی آن ها انتخاب می کنند. این روش ها شامل سه دسته لغوی مانند مقاله ارائه شده در مرجع [34]، نحوی مانند مرجع [35] و آنالیز معنا محور مانند مرجع [36] می باشند.

2-4-1-3 روش های برپایه یادگیری ماشین

این روش ها با استفاده از کلمات کلیدی استخراج شده توسط انسان به ایجاد یک مدل می پردازند و از این مدل ایجاد شده برای استخراج کلمات کلیدی در اسناد جدید بهره می گیرد.

---

[1] Co-occurrence

[2] PATrica tree که به نام suffix tree (درخت پسوندی) نیز شناخته می شود.

[3] Digital search tree

این روش ها به توجه به فراوانی و اهمیت یادگیری ماشینی در هوش مصنوعی به طرز گسترده ای مورد استفاده قرار گرفته اند. از این میان می توان به استفاده از قانون بیز[1] [37]، ماشین های بردار پشتیبان[2][38] بقچه کلمات[3] [35] اشاره کرد. همچنین برخی از ابزار های استخراج کلمات کلیدی مانند KEA[4] [39] و GenEx[5][40] نیز با استفاده از روش های این دسته ساخته شده‌اند.

2-4-1-4 روش های ترکیبی

روش های ترکیبی با استفاده از ترکیب روش های ذکر شده قبلی بدست آمده اند. در واقع روش هایی که با استفاده از توابع اکتشافی بر اساس مکان و طول و خصوصیات نمایشی کلمات کلیدی یا با استفاده از تگ های HTML [41] به استخراج کلمات کلیدی دست می زنند نیز در این دسته رده بندی می‌شود.

در پایان باید گفت به طور کلی در اکثر روش های معروف تعداد کلمات استخراج شده به عنوان کلمه کلیدی 10 الی 15 کلمه می باشد. همچنین اکثر روش های استخراج کلمات کلیدی مبتنی بر پردازش زبان طبیعی[6] از دیکشنری برای مشخص کردن ریشه کلمات و بخشهای گفتار استفاده می کنند.

2-4-2 روش های امتیاز دهی به کلمات

وزن دهی به کلمات یکی از مهمترین مراحل استخراج کلمات کلیدی است. در این مرحله به هریک از کلمات موجود در لیست بدست آمده از مرحله پیش پردازش، امتیازی داده می شود که میزان کلیدی بودن کلمه را تعیین می کند. سپس با اعمال یک آستانه که می تواند براساس امتیاز کلمات یا تعداد مشخص کلمات در رتبه های مختلف باشد کلمات کلیدی انتخاب می شوند.

---

[1] Naïve Bayes
[2] SVM (support vector machine)
[3] Bagging
[4] Keyword Extraction Algorthim
[5] نام روش از ترکیب دو عبارت genetic algoritm و extraction گرفته شده است.
[6] NLP (Natural Language Processing)

2-4-2-1 وزن دهی غیر آماری

در این روش وزن دهی به کلمات بر اساس پارامترهای انسانی که اهمیت یک کلمه در متن را مشخص می‌کند صورت می گیرد. لیست زیر به نمونه هایی از این پارامترها اشاره می‌کند[36].

- مکان قرارگیری کلمه در متن (چیکده، عنوان، و ...)
- مفهوم هر کلمه، که بیانگر ارتباط کلمه با کلمه های دیگر است.(مترادف، متضاد)
- کاربرد خاص کلمه: مثلا اسامی خاص دارای بار معنایی بیشتری هستند.

2-4-2-2 وزن دهی آماری

برعکس روش های وزن دهی غیر آماری که تکیه بر متون ساخت یافته یا لغتنامه ها دارند، روش‌های وزن‌دهی آماری تنها بر حضور کلمات تکیه دارند درنتیجه به نوعی مستقل از زبان محسوب شده و می-توانند در سطح وسیعی از اسناد مورد استفاده قرار گیرند.

وزن دهی آماری بر دو پایه فراوانی مطلق که وزن دهی را بر اساس تعداد حضور کلمه در یک سند منفرد و فراوانی نسبی که ارزش دهی را براساس حضور کلمه در اسناد مختلف انجام می دهد، بنا نهاده شده است. روش های فراوانی مطلق در سیستم های برخط[1] که نمی تواند به حجم مقبولی محدود شود کارایی بیشتری دارد.

روش های زیادی برای وزن دهی آماری تعریف شده که برخی از آن ها مرور می کنیم:

روش وزن دهی TFIDF[2]

[1] Online
[2] Term Frequency * Invert Document Frequency

این روش به خاطر سادگی محاسبات و نیز نتایج قابل قبول یکی از پرکاربردترین روش های وزن دهی آماری به کلمات بشمار می رود[42]. برای کلمه i در سند k ام TFIDF بصورت رابطه تعریف می شود:

$$TFIDF(w_{ik}) = freq(w_{ik}, D_k) * \log_2(\frac{N}{n} + 1) \qquad 1\text{-}2$$

که در آن

$freq(w_{ik}, D_k)$ به معنای تعداد تکرار کلمه i در سند k

N: تعداد کل اسناد

n: تعداد اسنادی که کلمه i در آن وجود دارد.

سیگنال و نویز

در این روش هرچه احتمال دیده شدن کلمه بیشتر باشد بار اطلاعاتی کمتری برایش در نظر گرفته می-شود. از آنجا که کلمات با ارزش تر در تعداد سند کمتری ظاهر شده اند نویز کمتری هم دارند.

نویز را بصورت رابطه 2-2 تعریف می کنیم:

$$NOISE_k = \sum_{i=1}^{n} \frac{FREQ_{ik}}{TOTFREQ_k} * \log_2 \frac{TOTFREQ_k}{FREQ_{ik}} \qquad 2\text{-}2$$

پارامتر سیگنال به گونه ای تعریف می شود که ارزش بیشتر را با معکوس نویز نشان دهد. (رابطه 2-3)

$$signal_k = \log_2(TOTFREQ_k) - NOISE_k \qquad 3\text{-}2$$

حالا وزن کلمه k در سندi ام بصورت 2-4 محاسبه خواهد شد.

$$WEIGHT_{ik} = FREQ_{ik} * signal_k \qquad 4\text{-}2$$

میزان ایجاد تمایز

در این پارامتر کلماتی که موجب بیشترین تمایز متن با سایر متون می شوند به عنوان کلمات کلیدی برچسب می خورند. میزان تمایز بر اساس معکوس میزان ایجاد شباهت، که معیارهای گوناگونی برای آن تعریف شده است بدست می آید. باهم به بررسی معیارهای شباهت می پردازیم

فرض کنید دو متن را بصورت دو بردار

$$X = (x_1, x_2, ..., x_t)$$
$$Y = (y_1, y_2, ..., y_t)$$

در نظر می گیریم. که درایه های x_i و y_i فراوانی کلمات در اسناد مربوطه باشد. در این صورت تابع متوسط مشابهت (رابطه 2-5) بیان کننده میزان شباهت کل اسناد مجموعه به یکدیگر است. این تابع هر اسناد به یکدیگر شبیه تر باشند مقدار بیشتری می گیرد.

$$AVGSIM = \frac{1}{n(n-1)} * \sum_{i=1}^{n} \sum_{j=1}^{n} similarity(D_i, D_j) \qquad i \neq j \qquad 5\text{-}2$$

آنچه که در این تابع به نام شباهت[1] تعریف شده است تابعی است که میزان مشابهت دو سند به یکدیگر را مشخص می کند. برای این تابع تعاریف مختلفی ارائه شده است که در جدول 2-3 تعدادی از آن ها را مشاهده می کنید.

[1] similarity

جدول 2-3 تعدادی از توابع شباهت مرسوم

معیار مشابهت	$similarity(X,Y)$
ضرب داخلی	$\sum_{i=1}^{t} x_i y_i$
ضریب Dice	$2 * \frac{\sum_{i=1}^{t} x_i y_i}{\sum_{i=1}^{t} x_i^2 + \sum_{i=1}^{t} y_i^2}$
ضریب کسینوسی	$\frac{\sum_{i=1}^{t} x_i y_i}{\sqrt{\sum_{i=1}^{t} x_i^2} * \sqrt{\sum_{i=1}^{t} y_i^2}}$
ضریب jaccard	$\frac{\sum_{i=1}^{t} x_i y_i}{\sum_{i=1}^{t} x_i^2 + \sum_{i=1}^{t} y_i^2 - \sum_{i=1}^{t} x_i y_i}$
ضریب همپوشانی	$\frac{\sum_{i=1}^{t} x_i y_i}{\min\left(\sum_{i=1}^{t} x_i^2, \sum_{i=1}^{t} y_i^2\right)}$

میزان تمایز برای کلمه K را بصورت $AVGSIM_k$ تعریف می کنیم که به معنای متوسط مشابهت در زمان عدم حضور کلمه در کلیه اسناد است. اگر K کلمه ای عمومی باشد آنگاه حذف آن موجب تمایز بیشتر اسناد با یکدیگر شده و $AVGSIM$ کاهش می یابد اما اگر k یک کلمه خاص و در اصطلاح متمایز

کننده باشد حذف آن منجر به شباهت بیشتر اسناد به هم شده و $AVGSIM$ هم افزایش می یابد با استفاده از تعاریف ذکر شده مقدار تمایز و وزن کلمه K در سند i با روابط 2-6و2-7محاسبه می شود.

$$Discvalue_k = (AVGSIM)_k - AVGSIM \quad 6\text{-}2$$

$$WEIGHT_{ik} = FREQ_{ik} * Discvalue_k \quad 7\text{-}2$$

2-4-3 پارامترهای ارزیابی کلمات استخراج شده

کلمات استخراج شده از یک متن توسط یک روش بازیابی اطلاعات را از دو منظر می‌توان ارزیابی کرد:

- در برگیری به این معنا که نشان می‌دهد چه مقدار از مفاهیم یک متن ممکن است با کلمات ارائه شده بازیابی شود افزایش تعداد کلمات مناسب این ملاک را افزایش می دهد.
- تعیین کنندگی : به معنای میزان دقیق بودن کلمات است و به این معنا که کلمات با ارتباط کمتر با اصل متن کمتر ارائه شده باشند..

برای ارزیابی کلمات به دوشکل عمل می شود:

1. **داوری مبتنی بر کارشناس انسانی**:

أ. نتایج بر اساس کلمات کلیدی استخراج شده توسط یک کارشناس ارزیابی شود.

این نوع ارزیابی بسیار رایج است و در این ارزیابی نتایج نهایی این پروژه نیز به همین روش انجام گرفته است اما اشکالات زیر را می توان بر این نوع ارزیابی وارد دانست.

1.کلمات کارشناس همیشه از روی خود متن استخراج نمی شوند به این معنا که یک کارشناس گاهی کلماتی را در لیست کلمات کلیدی ارائه می کند که مفهومی از متن را در خود دارد اما این کلمه مستقیما در خود متن آورده نشده

است. این نوع کلمات به هیچ وجه نمی تواند با استفاده از روش های استخراج کلمات کلیدی رایج استخراج شود.

2.معمولا به پایگاه داده‌ای که برای متون آن کلمات کلیدی استخراج شده باشد و در عین حال از جامعیت موضوعی برخوردار باشد دسترسی کمتری وجود دارد در واقع بیشتر پایگاه داده هایی که برای متون آن کلمات کلیدی استخراج شده باشد محدود به اسناد رسمی از قبیل صورت جلسه ها و یا اسنادی مانند مقالات علمی باشد.

3.همانطور که در بخش قبل گفته شد کلمات کلیدی برای متونی خاص با منظور خاص استفاده می شود بهمین خاطر معمولا از نویسنده خواسته می‌شود تعداد کلمات استخراجی را محدود نگه دارد.

ب. نتایج به یک کارشناس برای ارزیابی سپرده می شود:

در این حالت کارشناس مقادیر را از نظر ارزشی به کلمات انتخاب شده می دهد به عنوان مثال (خوب ، متوسط ،بد). این نوع داوری اگرچه مشکلات حالت قبلی را ندارد اما هزینه‌بر و زمانبر بوده و نیز می تواند دچار عدم عدالت در داوری باشد.

2. داوری مبتنی بر سیستم های بازیابی اطلاعات:

در این حالت برای ارزیابی کلمات کلیدی، باید یک مجموعه از متون به همراه مجموعه ای از پرس و جوهای مرتبط با اسناد و هم چنین تمام اسناد مرتبط با پرس و جوها وجود داشته باشد . پرس و جوها به مجموعه اعمال می شوند، و اسناد با توجه به مشابهت پرس و جو و اسناد ، بازیابی می شوند.

در حالت کلی نتایج بدست آمده از داوری ها، افراز های گوناگونی را به وجود می‌آورد که براساس این افرازها (جدول 2-4) ملاک های ارزیابی روش های استخراج کلمات کلیدی تعریف می شوند.

جدول 2-4 افراز های ممکن برای مجموعه اسناد

	نتایج بدست آمده	نتایج بدست نیامده	
پاسخ های بدست آمده	$A \cap B$	$\bar{A} \cap B$	B
پاسخ های بدست نیامده	$A \cap \bar{B}$	$\bar{A} \cap \bar{B}$	$\bar{B}$
	A	$\bar{A}$	

در جدول 2-4 افراز اول مجموعه نتایج را به دو حالت نتایج بدست آمده از داوری (A) و نتایج بدست نیامده از داوری $(\bar{A})$، و افراز دوم نتایج را به دو حالت پاسخ های بدست آمده از سیستم (B) و پاسخ های ایجاد نشده در سیستم $(\bar{B})$ تقسیم می کند

پارامترهایی که برای ارزیابی نتایج سیستم در تحقیقات به کار گرفته می شوند بصورت زیر از جدول 2-4 استخراج می شوند(توجه کنید که در رابطه های ذکر شده تعداد عناصر موجود در هر مجموعه به کار رفته است) :

2-4-3-1 میزان بازخوانی

پارامتر میزان بازخوانی[1] تعیین می کند که چه مقدار از نتایجی که سیستم ارزیابی تعیین کرده بوده بدست آمده است. به عنوان مثال اگر در بحث استخراج کلمات کلیدی این پارامتر تعیین می کند که چند درصد از کلمات کلیدی مورد انتظار را سیستم بدست آورده است. پارامتر بازخوانی بصورت رابطه2-8 بدست می آید. پارامتر میزان بازخوانی را می توان میزان صحت کارکرد سیستم دانست

[1] recall

$$R = \frac{A \cap B}{A} \qquad 8\text{-}2$$

2-4-3-2 میزان دقت

پارامتر میزان دقت[1] تعیین می کند که سیستم طراحی شده چقدر توانایی به ارائه صرف نتایج مورد نظر کاربر دارد. به عنوان مثال پارامتر دقت در یک سیستم استخراج کلمات کلیدی تعیین می کند که چه تعداد از کلمات ارائه شده دقیقا کلماتی است که کاربر انتظار داشته است در حالیکه پارامتر بازخوانی که تعیین می کرد چه تعداد از کلمات کاربر ارائه شده است. پارامتر میزان دقت با استفاده از رابطه 2-9 بدست می آید.

$$P = \frac{A \cap B}{B} \qquad 9\text{-}2$$

شایان ذکر است که پارامتر های میزان دقت و میزان بازخوانی در عمل تا حدودی با یکدیگر رابطه‌ی معکوس دارند به این معنا که در صورتیکه بخواهیم پارامتر دقت را افزایش دهیم باید مجموعه ی پاسخ سیستم را کوچک نگه داریم تا از ورود کلمات اضافه جلوگیری کنیم در حالیکه برای افزایش پارامتر میزان بازخوانی باید مجموعه پاسخ را افزایش دهیم تا شانس حضور بیشتری از کلمات مورد انتظار کاربر را ایجاد کرده باشیم. به همین خاطر در ارزیابی دو سیستم بازیابی اطلاعات، سیستمی بر دیگری برتری دارد که در هر دو پارامتر نتایج بهتری کسب کند.

2-4-3-3 پارامتر Fmeasure

این پارامتر به منظور در نظر گرفتن هر دو معیار بازخوانی و دقت بصورت همزمان است . این پارمتر بصورت میانگین هارمونیک بازخوانی و دقت تعریف می شود (رابطه 2-10)و هدف سیستم های بازیابی اطلاعات را می‌توان بیشینه کردن این پارامتر عنوان کرد.

[1] precision

$$F_{measure} = \frac{2*P*R}{P+R} \qquad 10\text{-}2$$

در پایان این بخش مثالی را برای تمامی این پارامترها حل می کنیم.

اگر کلمات کلیدی اعلامی یک کارشناس برای یک متن ورزشی فرضی« امیر قلعه نویی – باشگاه استقلال – قهرمانی لیگ برتر – فرهاد مجیدی» باشد و یک سیستم استخراج کلمات کلیدی بصورت اتوماتیک کلمات « امیر قلعه نویی – قهرمانی لیگ برتر – استقلال – سقوط ابومسلم مشهد – فرهاد مجیدی» را کلیدی اعلام کند در این صورت ارزیابی سیستم بصورت $precision = \frac{2}{5} = 40\%$ و $recal = \frac{2}{4} = 50\%$ خواهد بود.

همچنین با توجه به رابطه 2-10 خواهیم داشت.

$$F_{measure} = \frac{2*0.4*0.5}{0.4+0.5} = 0.44$$

2-5 مروری بر تحقیقات انجام شده

استخراج کلمات کلیدی به *روش آماری*، به استثنای برخی فرایندهای پیش پردازش تا حدی زیادی مستقل از زبان است به همین علت در ادامه تعدادی از کارهای انجام شده در سایر زبان ها آورده شده است. برای زبان فارسی تنها توانسته ایم از یک کار در حوزه نمایه سازی نام ببریم

روش‌های یادگیری با ناظر به علت امکان ارزیابی ملموس و نیز انطباق اصول آن با روش‌های رایج هوش مصنوعی از راهکارهای پر استفاده در استخراج کلمات کلیدی محسوب می‌شوند. این روش‌ها نیازمند تعدادی متن که کلمات کلیدی آن ها مشخص شده باشد، به عنوان پایگاه داده هاست. سیستم در طول

فاز آموزش سعی در یادگیری نحوه انتخاب کلمات کلیدی می کند و در فاز تست نتایج بر اساس پارامترهای بازخوانی و دقت مشخص می شود. از آنجا که روش استفاده شده در این تحقیق نیز بر این اساس پایه ریزی شده برخی از مقالاتی را که تاثیر بیشتری در این تحقیق داشته اند و برپایه این روش‌ها طراحی شده اند، مرور می کنیم.

در مرجع [3] نویسنده با استفاده از روش آماری CRF [1] و همچنین نگاه کردن به مسئله استخراج کلمات کلیدی به عنوان یک مسئله برچسب گذاری رشته ها به استخراج کلمات کلیدی به روش با ناظر، در متون چینی پرداخته است.

خصوصیاتی که برای امتیاز دهی کلمات استفاده می شود به نوبه خود می تواند تاثیر زیادی بر نتایج کار بگذارد. مرجع [43] یکی از مقالاتی است که برای استخراج کلمات کلیدی با ناظر به معرفی تعدادی خصوصیت پرداخته و میزان تاثیر گذاری هریک از این خصوصیات را در انتخاب کلمات کلیدی با استفاده از تست ANOVA بررسی کرده است.

زمانی که نخواهیم از یک مجموعه سند به عنوان پایگاه داده اولیه استفاده کنیم مجبوریم از خصوصیاتی در انتخاب کلمات کلیدی استفاده کنیم که تنها از یک سند منحصر به فرد بدست می آید. این خصوصیات می توانند مبتنی بر دانش آماری موجود در همان سند یا مبتنی بر ویژگی های غیر آماری مانند برچسب های HTML(در اسناد تحت وب) و ... باشد. مرجع [41] همانطور که قبلا هم ذکر شد با استفاده از برچسب های HTML به استخراج کلمات کلیدی پرداخته است.

مرجع [44] با استفاده از چهار خصوصیت تعداد تکرار عبارت نرمال شده با تقسیم شدن بر تعداد تکرار پرتکرارترین کلمه (NTF)، معکوس تعداد اسناد شامل کلمه (IDF) ، توزیع نرمال تعداد تکرار پاراگراف و

[1] Conditional Random Fields

اینکه آیا متن در عنوان ، تیتر یا زیرتیتر حضور دارد مسئله استخراج کلمات کلیدی را به عنوان یک مسئله تشخیص کلاس قلمداد کرده است و سپس با استفاده از یک شبکه عصبی پرسپترون چند لایه به حل مسئله پرداخته است.

مرجع [40] برای ارزیابی اینکه یک کلمه کلیدی است یا خیر؟ از یک درخت تصمیم[1] استفاده کرده است. همچنین این روش از 9 خصوصیت آماری تعداد کلمه موجود در عبارت، مکان اولین وقوع عبارت در متن، تعداد وقوع عبارت در سند، طول نسبی عبارت، و همچنین سه خصوصیت نحوی «آیا عبارت حاوی اسم خاص است؟»،«آیا کل عبارت در پایان به یک صفت ختم می شود؟» و «آیا عبارت حاوی یک فعل عمومی است؟» برای حل مسئله بهره می گیرد.

پروژه KEA [39]که پیش از این به آن اشاره شد با استفاده از سه خصوصیت TF و IDF و مکان اولین وقوع کلمه و یک کلاسبند بیزین تصمیم می گیرد که یک کلمه آیا کلیدی است یا نه. همچنین روش GenEx روش دیگریست که در این منبع معرفی شده و از الگوریتم ژنتیک استفاده می کند. این مقاله بابررسی شرایط توضیح می دهد که هریک از این دو روش در یک شرایط خاص نتایج بهتری نسبت به دیگری بدست می آورد.

در نمایه ساز «سینا»[42] به عنوان یک نمایه ساز متون فارسی ابتدا کلمات پر تکرار بر اساس یک لیست ثابت 180 کلمه ای حذف شده اند. سپس با استفاده از یک روش مبتنی بر حذف پسوندها و پیشوندها به فرایند ریشه یابی انجام شده است. این نمایه ساز برای فرایند استخراج کلمات کلیدی بعد از تست چهار روش TFIDF، LNU، ITN و NTC با توجه به نتایج TFIDFو LNU را برای امتیازدهی گزینش و روش خود را بر450 متن شامل چکیده مقالات مرتبط با کامپیوتر را به عنوان پایگاه داده اعمال کرده است.

[1] Decision tree

فصل سوم:

آشنایی با اصول اولیه SEO

3-1 مقدمه

تابع ارزیابی کلمات کاندید که در فصل ششم مورد استفاده می گیرد بر اساس عملکردهای موتورهای جستجو طراحی شده است. برای شرح بهتر این تابع فصل جاری را به اصول بهینه سازی ساختار سایت ها برای دستیابی به رتبه های بالاتر یا اصطلاحا SEO [1] پرداخته ایم. بیشتر مطالب این فصل بر اساس اسناد منتشر توسط موتور جستجوی گوگل به خصوص سند ذکر شده در مرجع [45] می باشد و مسلما فقط نکاتی که در طراحی تابع ارزیابی کلمات کاندید لحاظ شده است بررسی شده اند. در واقع آنچه که در ادامه آمده است پنجره‌ای کوچک به دنیای بزرگ و پیچیده SEO محسوب می شود.

3-2 ساختار یک موتور جستجو

به طور خلاصه یک موتور جستجو شامل پنج قسمت است:

1. خزنده[2] یا کشف کننده
2. عنکبوت[3] یا تارگذار
3. شاخص گذار[4]
4. پایگاه داده
5. الگوریتم رتبه بندی

[1] Search Engine Optimization
[2] Crowler
[3] Spider
[4] indexer

3-2-1 تارگذار

در یک موتور جستجو، تارگذار در واقع کار یک کاربر اتوماتیک را انجام می دهد. این بخش از موتور جستجو به صفحات مختلف سایت ها سر می زند و محتوای صفحات را در اختیار سایر بخش ها می‌گذارد.

3-2-2 خزنده

خزنده به عنوان تصمیم گیرنده برای تارگذار عمل می کند و با فرمان‌هایش برای تارگذار مشخص می‌کند که پیوندهای یک صفحه را دنبال کرده یا در همانجا متوقف می شود.

3-2-3 شاخص گذار

اطلاعات جمع آوری شده توسط تارگذار، بوسیله شاخص‌گذار بررسی شده و بصورت کلمات و امتیاز کلمات ذخیره می‌شود. در واقع شاخص گذار وظیفه تبدیل سیستم انسانی متن به یک سیستم عددی را دارد همچنین اکثر شاخص‌گذار ها برای کاهش حجم داده ها از کلمات پرتکرار صرفنظر می‌کنند به همین خاطر توصیه می شود در صورتیکه می خواهید از یک توالی کلمات پرتکرار استفاده کنید آن را در میان علامت نقل قول («» یا "") ارائه کنید

3-2-4 پایگاه داده

داده های تحلیل شده شاخص گذار به پایگاه داده فرستاده شده و در اینجا بعد از گروه بندی ذخیره می-شود. هرچه پایگاه داده یک موتور جستجو بزرگتر باشد توانایی پاسخگویی و جامعیت پاسخ موتور جستجو بیشتر است. در واقع وقتی پرسشی از یک موتور جستجو مطرح می‌شود موتور جستجو نه کل وب! که پایگاه داده اش را برای ارسال پاسخ جستجو می کند.

3-2-5 سیستم رتبه‌بندی

زمانی که کاربر کلماتی را برای جستجو به موتور جستجوگر تحویل می دهد موتور جستجو صفحاتی را که به موضوع مرتبط باشد مشخص می کند سپس آن ها را به ترتیب از بیشترین ارتباط تا کمترین ارتباط مرتب می‌کند به عنوان پاسخ ارائه می‌کند.

اگرچه الگوریتم رتبه بندی یک موتور جستجو از کاربران مخفی نگه داشته می شود، اما کاربران با روش های مختلف سعی در کشف آن و ارائه مطالب سایت بصورتی که رتبه بالاتری کسب کنند، دارند.

3-3 اصول اولیه SEO

بهینه سازی برای موتورهای جستجو یا SEO به طور معمول به ایجاد تغییرات در بخش هایی از سایت گفته می شود که منجر به تاثیرات مطلوب در دستیابی پذیری بیشتر توسط موتورهای جستجو می شود.

در ادامه به مواردی که حضور کلمات کلیدی در آن تاثیر زیادی در میزان دسترسی‌پذیری سایت توسط موتورهای جستجوی می شود اشاره می کنیم.

3-3-1 تگ عنوان (<title>)

عنوان یک صفحه وب هم به کاربران و هم به یک موتور جستجو می گوید که موضوع اصلی و خاص صفحه مورد نظر چیست. این تگ در میان تگ <head> قرار می گیرد. در حالت ایده آل شما باید برای

هر صفحه از سایتتان یک عنوان منحصر به فرد قرار دهید. از آنجا که عنوان انتخابی شما در خلاصه توضیح[1] نتایج جستجو نشان داده می شود این مسئله نیز در انتخاب باید موثر باشد.

طول عنوان نیز مهم است بطوریکه کلمات کلیدی موجود در یک عنوان طولانی کمتر از کلمات کلیدی در یک عنوان کوتاه ارزش دارند.

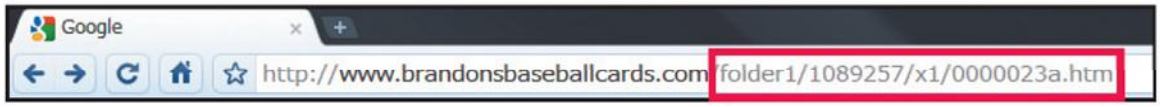
3-3-2 متا تگ های توضیح

این تگ ها که در تگ <head> آورده می شوند اطلاعات مفیدی برای موتورهای جستجو فراهم می‌کنند. چرا که بر عکس عنوان که بهتر است چند کلمه باشد یک متا تگ را می‌توان در اندازه یک پاراگراف کوتاه تنظیم کرد. همچنین به نظر می‌رسد موتورهای جستجو به مطالب این تگ ها اهمیت ویژه ای می دهند بطوریکه در بیشتر موارد خلاصه توضیحات را بر پایه این تگ قرار می دهند، به همین علت بهتر است از گذاشتن یک لیست صرف کلمات یا عباراتی که توضیح کاملی در مورد سایت ندارند خودداری شود. حتی در حالت ایده‌آل سعی شود متا تگ های توضیحی نیز بصورت منحصر به فرد نوشته شود. (البته در مواردی که یک سایت صفحات زیادی دارد می توان با استفاده از اتوماتیک سازی برای این کار استفاده کرد.)

3-3-3 آدرس ها (URL)

استفاده از نام‌های با معنی برای فایل ها و فولدرهای یک سایت نه تنها به عنوان یک توصیه مناسب برای طراحان سایت مطرح می‌شود بلکه منجر به کشف بهتر سایت توسط خزنده‌های موتورهای جستجو نیز می‌شود. و در واقع موتورهای جستجو به سایت هایی که در آن به آدرس‌های با معنی ارجاع وجود دارد امتیاز بیشتری می دهند. شکل 3-1 دو آدرس خوانا و ناخوانا را نمایش می‌دهد.

[1] snippet



1. یک آدرس بد برای سایتی که جهت ارائه کارت های بیس بال طراحی شده است.

2. یک آدرس مناسب، کلمات مشخص شده به کاربران و موتورهای جستجو کمک می کند تا فهم راحت تری از مفهوم ارائه شده در سایت داشته باشند.

شکل 3-1 نمونه ای از دو آدرس ناخوانا (شکل بالا) و ناخوانا (شکل پایین)

هرچه عمق آدرس کمتر و ساختار درختی و نحوه نامگذاری ها مناسب تر باشد هم سایت شما برای موتور جستجو دسترس پذیرتر است و هم احتمال انتخاب شدن توسط کاربران به علت ذکر آدرس در خلاصه توضیحات بیشتر خواهد بود.

بهتر است سعی شود تا صفحاتی که از ریشه به آن ها می رسیم با صفحات شاخه ها متفاوت باشد به این معنی که داشتن آدرس domain.com\p1.htm همزمان با sub.domain.com\p1.htm تاثیر بدی بر رتبه-بندی خواهد داشت. همچنین انتظار می رود که آدرس ها با حروف کوچک نوشته شوند.

3-3-4 کلمات

به کلماتی فکر کنید که کاربران ممکن است برای یافتن مطلب شما جستجو کنند. مطمئنا کاربری که در یک موضوع خبره است کلماتی متفاوت از کسی که اطلاعات کمی دارد جستجو می کند به عنوان مثال کاربری که در مورد استخراج کلمات کلیدی تحقیق می‌کند ممکن است عبارت «بهبود بازخوانی» را جستجو کند اما کاربری که تازه به جمع محققین این زمینه پیوسته در جستوی مطالبیست که عبارت

«ملاک های کارایی» را در خود داشته باشد. موتور جستجوی گوگل توسط Google AdWords ابزار ساده Keyword Tool [1] را جهت کشف کلمات کلیدی متفاوت و بررسی حجم جستجو تقریبی این کلمات در اختیار کاربران قرار داده است. همچنین می توان با استفاده از ابزارهای تهیه شده در top search queries [2] نشان می دهد که سایت شما بیشترین بازدید را توسط چه کلماتی به خود اختصاص داده است.

در راهنمای SEO برای کاربران گوگل آمده است هرچه متن یک سایت بر موضوع بیشتر متمرکز باشد امتیاز بیشتری توسط موتور جستجو می‌گیرد که البته باتوجه به رابطه منطقی فاکتور TFIDF در بررسی آماری متون قابل پیش‌بینی نیز می باشد. همچنین گفته شده از کپی محض یا تکرار مطالب نزدیک به مطالب دیگران نیز تا حد امکان پرهیز کنید.

3-3-5 متون پیوندی

متون پیوندی[3] به آن متن های اتلاق می شود که قابل کلیک بوده و در صورت کلیک کاربر به آدرسی دیگر منتقل می شود. این متون که با استفاده از تگ <a> ایجاد می شود دارای دوبخش آدرس (herf) و متن نمایش می باشند. متن نمایش در واقع برای کاربران و موتورهای جستجو محتوای آدرس مورد اشاره را در چند کلمه شرح می دهد. برای نتیجه گیری بهتر است از اشتباهات رایج زیر اجتناب کنید.

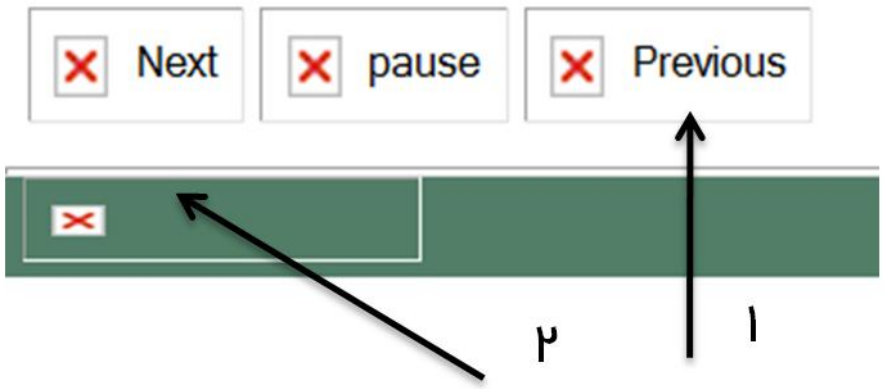
- استفاده از کلمات عمومی مثل «اینجا را کلیک کنید» و غیره
- استفاده از متون و کلماتی که ارتباطی با صفحه مورد اشاره ندارند
- استفاده از متون بلند

[1] https://adwords.google.com/select/KeywordToolExternal
[2] http://www.google.com/webmasters/edu/quickstartguide/sub1guide5.html
[3] Anchor text

- استفاده از خود آدرس برای متن نمایش
- متن توضیح آدرس

3-3-6 تصاویر

تصاویر در HTML با استفاده از تگ img معرفی می شوند اما در میان صفت alt در این تگ نقش تعیین‌کننده‌ای برای موتورهای جستجو دارد. صفت alt به طراح سایت اجازه می‌دهد که یک متن جایگزین برای تصاویر تعریف کند تا در صورت عدم نمایش تصویر توسط مرورگر به هر علت کاربر بداند که در این مکان چه چیزی نمایش داده شده است. شکل 3-2 قسمتی از یک وب سایت را نشان می دهد که مرورگر نتوانسته است تصاویر آن را نمایش دهد. پیکان 1 به عکسی اشاره دارد که برای آن صفت alt برابر با previous قرار داده شده است و پیکان 2 به تصویری اشاره دارد که alt آن مقداردهی نشده است.



شکل 3-2 نمونه ای از یک عکس دارای alt (1)و عکس بدون alt(2)

مقدار صفت alt یا به تعبیری محتوای صفت alt در واقع شرح متنی یک تصویر است از این رو برای موتورهای جستجو که نمی تواند عکس ها را تفسیر کنند اهمیت زیادی دارد.

3-3-7 تگ های تیتر و زیر تیتر

تگ های تیتر[1] <h1> تا <h6> در صورتیکه به درستی استفاده شوند تاثیر زیادی در کسب رتبه مناسبی در نتایج موتورهای جستجو می‌شوند چرا که به طور طبیعی واژه های کلیدی بیشتری به علت ارائه درونمایه سایت در خود دارند. البته یادمان باشد که این تگ ها برای مدیریت مطالب سایت استفاده می-شوند و برای برجسته کردن یک مطلب و ... از تگ های مخصوص اینکار مثل یا باید استفاده کرد. در کل توصیه می شود از اشتباهات زیر در رابطه با تگ های تیتر اجتناب کنید:

- قرار دادن جملات طولانی در یک تگ تیتر
- استفاده‌ای برای کاربردهایی غیر از ارائه ساختار سایت

[1] Headings

فصل چهارم:

پایگاه داده ها

4-1 مقدمه

همانگونه که در فصل قبل ذکر شد یکی از مشکلاتی که معمولا تحقیقات مرتبط با پردازش زبان فارسی با آن روبرو می شود نبود منابع زبانی کافی و معتبر برای فارسی است. همانطور که در بخش کارهای گذشته ذکر شد در رابطه با این مسئله تلاش هایی صورت گرفته است اما تا زمان اجرای مراحل این تحقیق هیچ پایگاه داده ای که بصورت مشخص برای مسئله استخراج کلمات کلیدی باناظر قابل استفاده باشد (در ساده‌ترین حالت کلمات کلیدی آن از دید یک کارشناس استخراج شده باشد) ، ارائه نشده است. به همین جهت برای پروژه استخراج کلمات کلیدی اقدام به ایجاد یک پایگاه داده کوچک نموده ایم که در این فصل به معرفی این پایگاه داده می پردازیم.

4-2 نقطه شروع

بسته به نوع تحقیق متون استفاده شده برای ایجاد پایگاه داده می توانند متون رسمی، خبری، اداری و حتی محاوره ای باشد. اما در صورتیکه بخواهیم پایگاه داده جامعیت پیدا کرده و برای تحقیقات مشابه نیز قابل استفاده باشد، به کارگیری متون رسمی دارای تنوع در موضوعات نسبت به سایر موارد ارجح است.

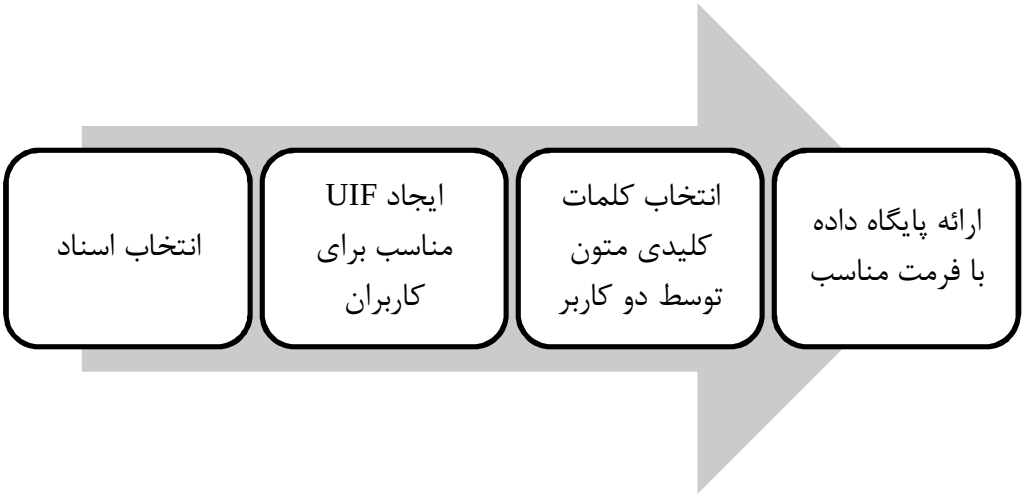
پایگاه داده پروژه های استخراج کلمات کلیدی می تواند تاثیر بسازایی در کارایی تحقیق داشته باشد این تاثیر با توجه به این مسئله که فاکتورهایی مثل IDF وابستگی مستقیم به پایگاه داده دارند بیشتر از سایر تحقیقات مرتبط با بازیابی اطلاعات خواهد بود.

به عنوان نمونه ای همانطور که در بحث کلمات پرتکرار مطرح شد یک کلمه می تواند در یک مجموعه متون مشخص فاقد ارزش تلقی شود در حالیکه در مجموعه دیگر کاملا کلمه ای کلیدی محسوب می‌شود.

به عنوان مثال فرض کنید پایگاه داده یک تحقیق فقط شامل متون ورزشی باشد. آنگاه کلمه هایی مثل استادیوم، گل ، امتیاز، حریف و ... کلماتی بی ارزش تلقی می شوند. این در حالیست که اگر در یک مجموعه که فقط شامل متون معماری است یکی از کلماتی کلیدی متن «پژوهشی بر معماری استادیوم آزادی» کلمه «استادیوم» خواهد بود.

برای شروع کار از پایگاه داده همشهری استفاده کردیم. این پایگاه داده همانطور که در فصل دوم توضیح داده شد از جامعیت خوبی در تعداد سند و تعدد موضوعات برخوردار است. همچنین براحتی در دسترس بود و به علت آنکه مکرر در تحقیقات کنونی در حال استفاده شدن است می توان گفت اعتبار نسبتا خوبی نیز دارد.

روند ایجاد پایگاه داده در شکل 4-1 نشان داده شده است.

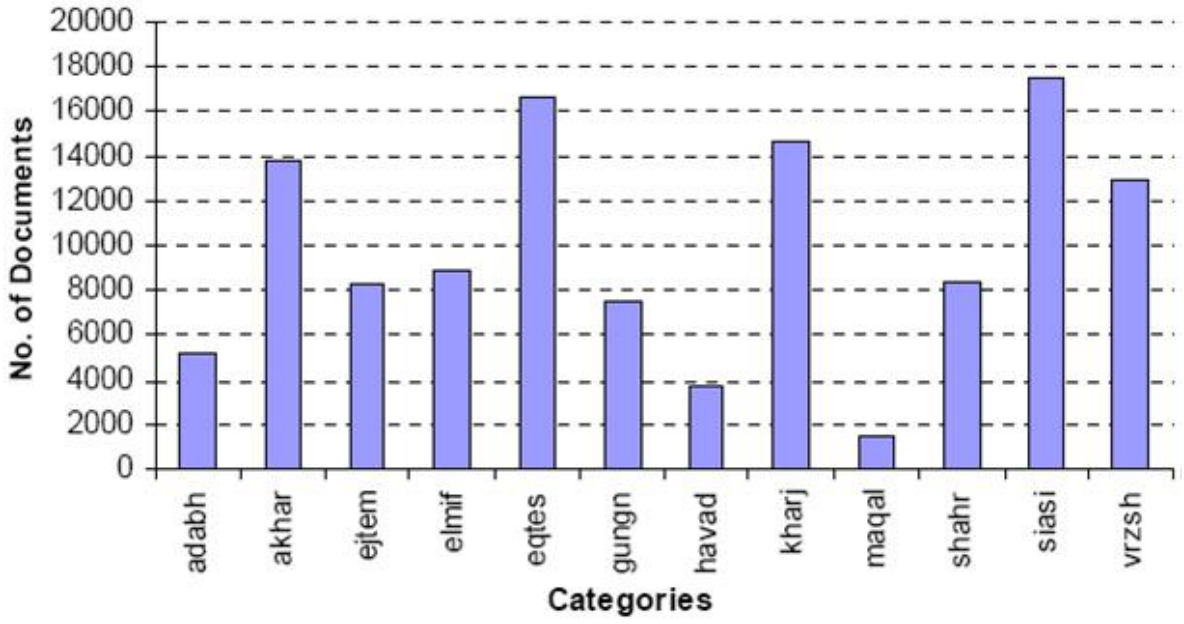


شکل 4-1 روند اتخاذ شده برای ایجاد پایگاه داده برای این پروژه

4-3 انتخاب اسناد

پایگاه داده همشهری از نظر توزیع اسناد دارای نموداری به صورت شکل 4-2 است. طول این اسناد متفاوت بوده و در برخی موارد به 8000 تا 9000 کلمه در یک سند نیز می رسد. اما از آنجا که برای انجام این تحقیق با کمبود منابع انسانی برای استخراج مواجه بوده ایم از اسنادی که کمتر از 1000 کلمه داشته‌اند استفاده کرده‌ایم. همچنین برای ایجاد جامعیت از 11 رسته مقالات پایگاه داده همشهری تعداد مناسبی از اسناد را انتخاب کرده ایم که در مجموعه به 1100 سند رسیده ایم. (از رسته مقالات به علت طولانی بودن هیچ سندی انتخاب نشده است.)

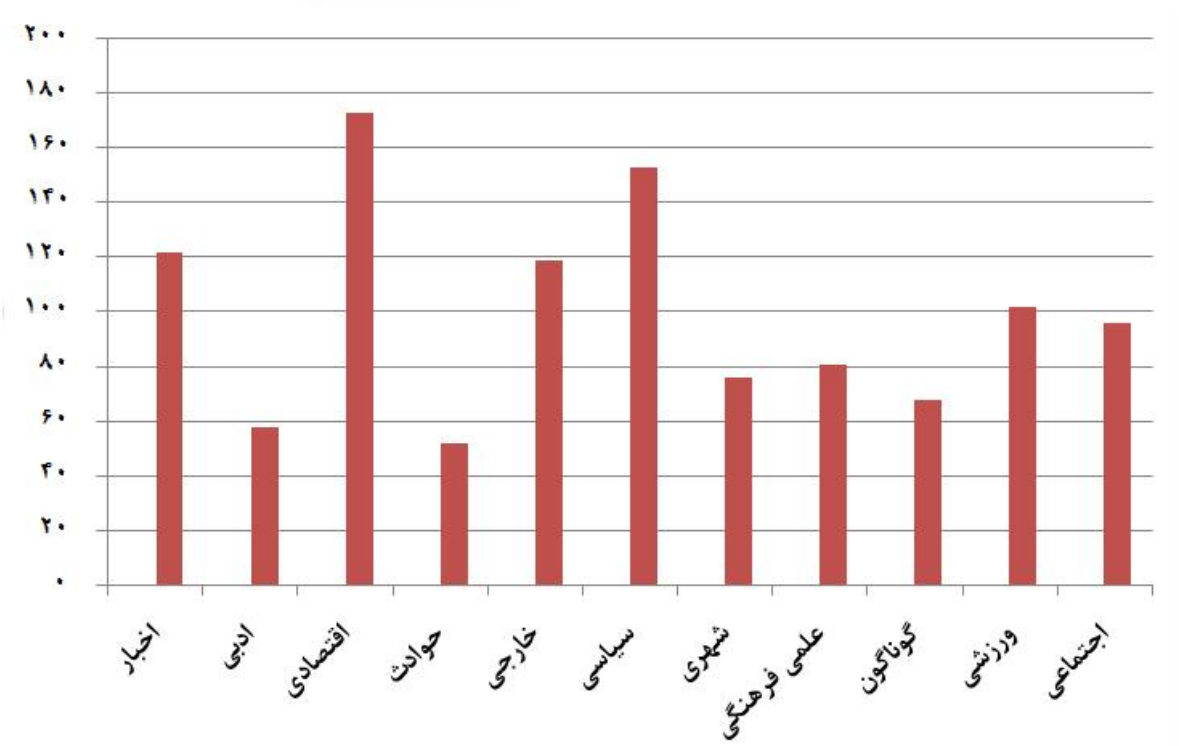
تعداد اسناد انتخاب شده از رسته های مختلف در شکل 4-3 نمایش داده شده است.



شکل 4-2 نمودار توزیع آماری اسناد در پایگاه داده همشهری از دید موضوع

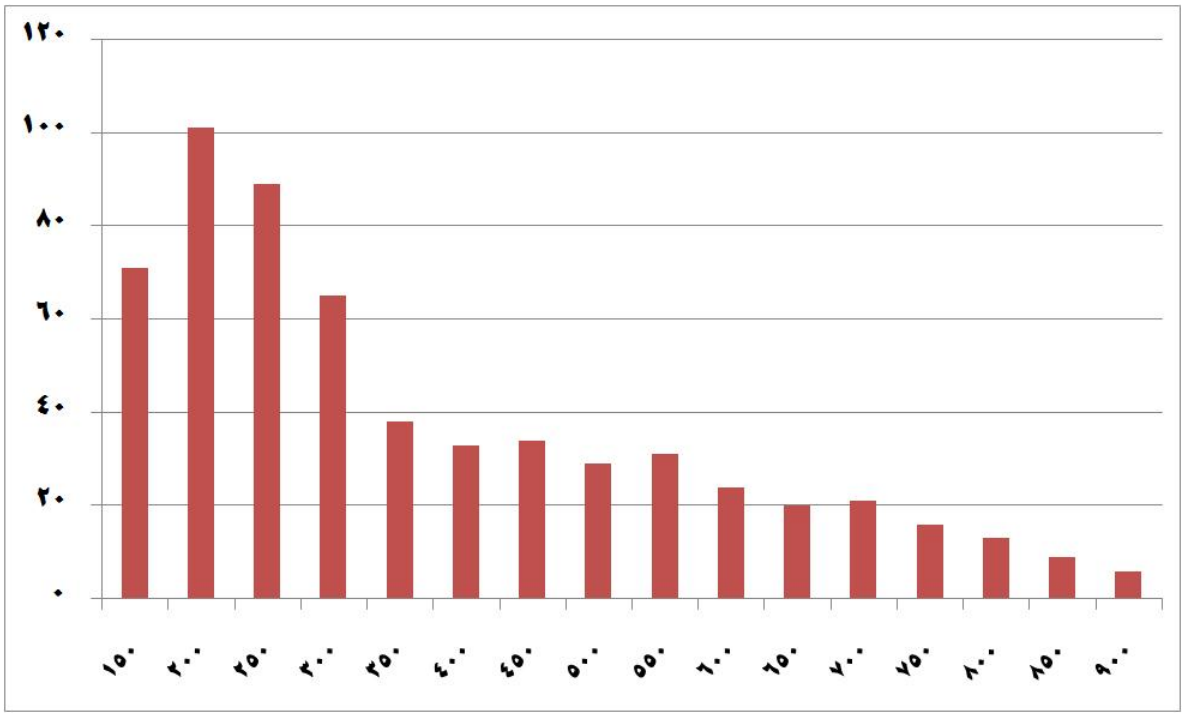
در واقع تعداد اسناد انتخابی بر اساس این موضوع که هریک از این رسته ها چند سند در پایگاه داده همشهری داشته اند انتخاب شده است.

اما برای آنکه جامعیت پروژه مخدوش نشود پایگاه داده در دو حالت تعداد سند مساوی (50 سند از هر دسته) و تعداد سند متغیر در هر رسته در نتایج پایانی به کار برده شده و نتایج این مسئله نیز مورد بررسی قرار گرفته است.



شکل 4-3 توزیع موضوعی اسناد پایگاه داده ایجاد شده در این پروژه

بازه تعداد کلمات موجود در اسناد انتخابی از 150 کلمه تا 900 کلمه است که بطور میانگین هر سند 372,5 کلمه را داراست. نوع توزیع اسناد انتخابی بر اساس تعداد کلمه موجود در اسناد در شکل 4-4 آورده شده است.



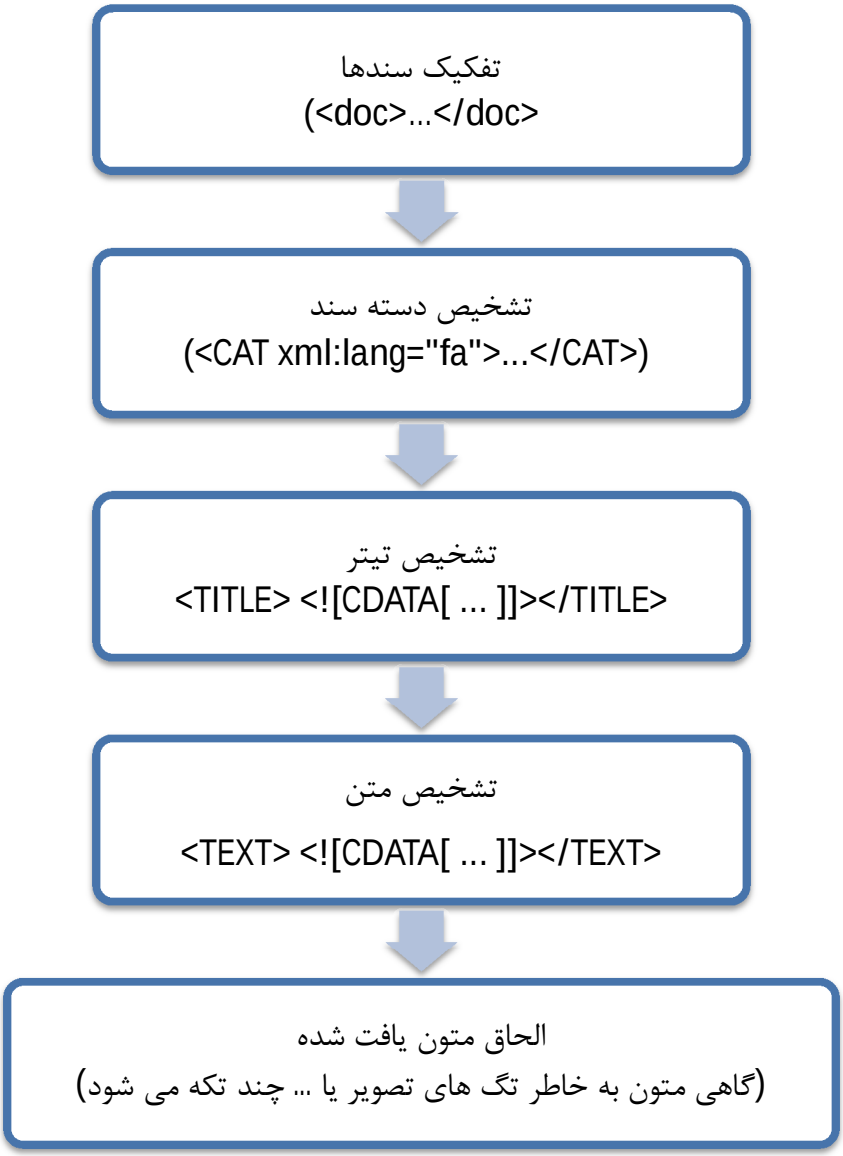
شکل 4-4-تعداد اسناد پایگاه داده براساس طول بر حسب کلمه

پس از این مرحله پایگاه داده برای مرور و استخراج کلمات کلیدی با استفاده از یک واسط کاربری ساده به خوانندگان ارائه شده است.

4-4 پاک سازی تگ ها برای پایگاه داده

از آنجا که ساختار پایگاه داده همشهری بصورت XML طراحی شده است. حذف تگ های XML برای نمایش به کاربر و تطابق با پایگاه داده طراحی شده ضروری است. برای اینکار از روندی مطابق با شکل 5-4 استفاده کرده ایم.

پس از این مرحله پایگاه داده برای مرور و استخراج کلمات کلیدی با استفاده از یک واسط کاربری ساده به خوانندگان ارائه شده است.



شکل 4-5 روند پاک سازی یک سند پایگاه داده همشهری

4-5 انتخاب کلمات کلیدی

در این مرحله برای هر سند از دو کاربر خواسته ایم تا به انتخاب کلمات کلیدی بپردازد تا از کانالیزه شدن کلمات بر اساس تفکر یک کاربر جلوگیری شود.

واسط کاربری ایجاد شده به نحو بسیار ساده ای طراحی شده است. این واسط ابتدا متن هر سند را از پایگاه داده همشهری جدا کرده و پس از دریافت کلمات کلیدی کاربران آن را بصورت شکل 4-6 در یک رکورد پایگاه داده ذخیره می کند.

نام فیلد	ID	category	RTF doc address	Keyword1	Keyword2
توضیحات	یک شماره منحصر به فرد	دسته سند (سیاسی و...)	آدرس فایل ذخیره شده سند	کلید خارجی از جدول کلمات کاربران	کلید خارجی از جدول کلمات کاربران

شکل 4-6 شکل رکورد ذخیره سازی اطلاعات و کلمات کلیدی یک سند

همچنین یکی از نکاتی که منجر به بدست آوردن بازخوانی کمتر می شود معمولا انتخاب کلماتی توسط کاربران است که در متن وجود ندارد. برای آن که تحلیلی در مورد تاثیرات این پدیده نیز داشته باشیم کلمات کلیدی را که رکورد های آن بصورت شکل 4-7 می باشد ذخیره کرده ایم.

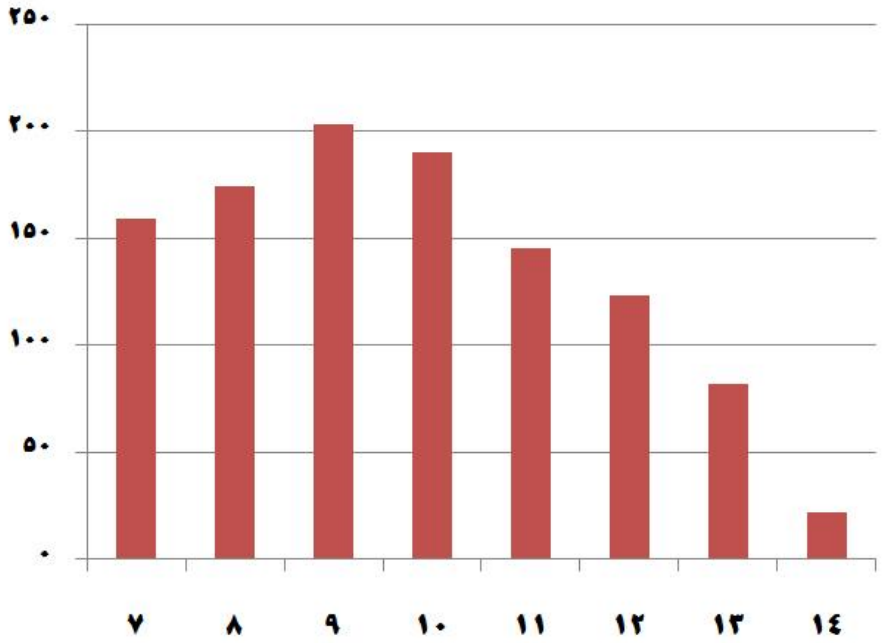
ID	Keyword	Does_exist
کلید جدول	کلمه کلیدی	یک فیلد باینری

شکل 4-7 ساختار رکورد جدول ذخیره کننده کلمات کلیدی

اگر کلمه کلیدی انتخابی کاربر دقیقا در متن وجود داشته باشد فیلد باینری Does_exist دارای مقداری برابر با 1 است و در غیر این صورت دارای مقدار صفر خواهد بود.

با توجه به این که از خوانندگان خواسته شده بود حداکثر 10 و حداقل 4 کلمه کلیدی برای هر متن انتخاب کنند در مجموع بین 7 تا 14 کلمه به عنوان کلمه کلیدی به هر سند منصوب شده است. میانگین

تعداد کلمه انتخابی برای هر متن 9,72 می باشد. شکل 4-8 توزیع تعداد کلمات کلیدی را در متون نشان می دهد.



شکل 4-8- تعداد اسناد موجود در پایگاه داده بر اساس تعداد کلمات کلیدی انتخاب شده برای آن

شکل 4-9 یک نمونه از اسناد پایگاه داده را نشان می دهد. این سند 305 کلمه دارد و کلمات کلیدی منتخب کاربران برای آن به شرح زیر است:

کلمات کلیدی کاربر1: کوفی عنان – دبیرکل سازمان ملل – حقوق بشر – ایدز – تغییرات جوی – جهانی شدن اقتصاد – خطرات بی مرز

کلمات کلیدی کاربر2: کوفی عنان - فلسفه وجودی سازمان ملل - همکاری های مشترک بین المللی - تغییرات اقتصادی - آمریکا - بدون همکاری بین المللی - جهانی شدن اقتصاد و تجارت - حقوق بشر - جامعه بشری

كوفي عنان: جهانيان ! براي مقابله با
خطرات بي مرز متحد شويد!

سازمان ملل - خبرگزاري جمهوري اسلامي: دبير كل سازمان ملل در پيامي به مناسبت آغاز سال جديد ميلادي، خواستارگسترش همكاري هاي بين المللي براي مقابله با مشكلات جهاني شد.
كوفي عنان گفت: مردم جهان سرنوشت مشتركي دارند اما فقط زماني مي توانند بر اين سرنوشت مشترك مسلط شوند كه با يكديگر، با آن روبرو شوند.
عنان با تكيه بر اينكه فلسفه وجودي سازمان ملل، همكاري هاي مشترك بين المللي است، گفت: قرني كه آغاز مي شود اميدهاي جديدي را بر مي انگيزد اما مي تواند خطرهاي جديدي نيز به ارمغان آورد و يا خطرهاي كهنه را در اشكالي جديد و هشداردهنده عرضه كند.
دبير كل سازمان ملل از تغييرات اقتصادي، تعصب، خشونت، بيماري ها وصدمه به محيط زيست، به عنوان خطرهاي عمده اي كه جهان را تهديد مي كند نام برد.
عنان تاكيد كرد: هيچكس از ميزان شدت اين خطرات دقيقا" مطلع نيست اما واضح است كه اين خطرات مرزهاي كشورها را در مي نوردد.
دبير كل سازمان ملل با اشاره تلويحي به آمريكا گفت: حتي قدرتمندترين كشورها نيز نمي تواند به تنهايي و بدون همكاري بين المللي، اتباع خود را در مقابل اين خطرات حفظ كند.
عنان افزود: از طريق سازمان ملل و با تلاش مشترك مي توان با ايدز وديگر بيماري هاي واگيردار مقابله كرد، تغييرات جوي را كنترل نمود و آب و هواي پاكيزه براي همه فراهم ساخت.
وي خاطرنشان كرد: به علاوه سازمان ملل تلاش مي كند جهاني شدن اقتصاد و تجارت به نفع همه باشد و به فقرا اجازه دهد خود را از فقر خارج سازند.
عنان تاكيد كرد: بايد از طريق سازمان ملل، حقوق بشر را براي همه به يك واقعيت تبديل كرد و به كل جامعه بشري اين فرصت را داد در تصميماتي كه بر زندگي آنها تاثير مي گذارد، حق اظهار نظر واقعي داشته باشند.

شكل 4-9 نمونه ای از یک سند پایگاه داده ایجاد شده

فصل پنجم:

فرایند های پیش پردازش

5-1 مقدمه

در این فصل به معرفی فرایند پیش پردازش ایجاد شده برای این پروژه می پردازیم. در ابتدا با استفاده از تجربیات سایر تحقیقات به اعمال یکسری از قوانین ساده یکسان سازی می‌پردازیم. سپس در بخش بعد با نگاهی تکنیکی تر نسبت به آنچه در فصل دوم درباره کلمات پرتکرار ارائه شد به حذف آن‌ها می‌پردازیم. راه حل ارائه شده برای حذف کلمات پر تکرار در این پروژه استفاده از یک لیست ثابت است اما تغییرات کوچکی برای بهبود فرایند حذف کلمات بی ازرش ارائه شده است. بخش انتهایی این فصل به مسئله توکنیزه کردن کلمات، و جملات می پردازد. توکنیزه کردن جملات از آن جهت انجام می شود که بخشی از خصوصیات استفاده در فصل آینده به جملات و وضعیت آن ها بستگی دارد. این پروژه از ریشه یابی خاصی استفاده نکرده است چرا که الگوریتم های ریشه یابی موجود هنوز در ابتدای راه هستند و حتی برای برخی قواعد دستوری زبان فارسی در بعضی موارد هنوز تعریف رسمی یکسانی وجود ندارد.

5-2 یکسان سازی

در بعد یکسان سازی تکنیکی با توجه به غیر جامع بودن این تحقیق از جنبه اسناد ورودی چه از منظر نوع ساختارهای ورودی (در بخش پایگاه داده ذکر شد که اسناد به فرمت RTF برای استفاده ذخیره شده اند.) و چه از منظر نوع رمزگذاری (با توجه به انتخاب اسناد از پایگاه داده همشهری که دارای رمزگذاری یکسان در سرتاسر اسناد است.) یکسان سازی خاصی نیاز نداریم . اما در بخش قواعد زبانی با استفاده از مجموعه ای کوچک از قواعد که در مقاله [17] ذکر شده است به یکسان سازی نسبتا خوبی رسیده ایم.

- تبدیل نویسه‌های «ؤ» به «و»، «ئ» به «ی» و «أ» به «ا».

- تبدیل نویسه‌های «ۀ» و «ۀ» به «ه» در آخر واژه‌ها.
- حذف «ی» از آخر واژه‌هایی مانند «خانه ی».
- حذف فتحه، کسره و ضمه (نویسه‌های ـَ ، ـِ ، ـُ) از واژه‌ها.
- حذف تنوین (نویسه‌های ـً ، ـٍ ، ـٌ) از انتهای واژه‌ها.
- حذف تشدید (شناسه‌ی ـّ) از واژه‌ها
- حذف شناسه‌ی «ء» در آخر بعضی واژه‌ها مانند «شهداء»
- چسباندن پیشوندهای «می»، «درمی»، «برمی»، «نمی» و «بی» به ابتدای واژه‌ها
- چسباندن پیشوند «هم» به واژه‌های مانند «هم چنین» به ابتدای واژه‌ها
- چسباندن پسوندهای «ها»، «های»، «هایی»، «هایم»، «هایت»، «هایش»، «هایمان»، «هایتان» و «هایشان» به انتهای واژه‌ها
- چسباندن پسوندهای «تر» و «ترین» به آخر واژه‌ها
- حذف فاصله بعد از پیشوند «بر» در واژه‌هایی مانند «بر می‌گردد».
- حذف نویسه‌ی «ـ» که برای کشش نویسه‌های چسبان جهت تراز شدن طول خطها مورد استفاده قرار می‌گیرد. مانند تبدیل «بـــر» و «بـر» به «بر».

شیوه یکسان سازی به این صورت بوده است که عبارات اشاره شده در قوانین در تمامی متون مورد بررسی جستجو شده و در صورت انطباق قانون مورد نظر اعمال شده است.

به عنوان مثال برای اعمال یکسان سازی منطبق بر آخرین قانون ابتدا نویسه «_» مورد جستجو قرار گرفته سپس در صورت یافت شدن ادامه رشته تا رسیدن به کاراکتری غیر از «_» برای پیدا کردن تعداد

بیشتر این نویسه مورد بررسی قرار گرفته است. سپس نویسه «_» حذف شده و دو زیر رشته[1] به جا مانده از حذف به یکدیگر الحاق می شوند.

3-5 حذف کلمات پر تکرار

در این پروژه برای حذف کلمات پر تکرار از یک لیست ثابت استفاده کرده ایم. اگرچه استفاده از روش های پویا دقت بالاتری داشته و مشکلات لیست ثابت را که در بخش قبل به آن اشاره شد در پی ندارد اما پیچیدگی های زمانی روش پویا بالاتر است. در واقع در توازن بین انتخاب بهتر کلمات پر تکرار و هزینه زمانی کمتر، برای روش ما با توجه به ارزیابی هایی که بعد از این نیز وجود دارد، وجود یک کلمه کم‌ارزش در لیست کلمات کلیدی به مراتب قابل تحمل تر از هزینه ی زمانی ساخت لیست بصورت پویاست.

لیست ثابت به این نحو تولید شده است که لیست 927 کلمه ای مرجع[20] را با لیست دیگری که طی پروژه‌ای دیگر با استفاده از مرور 45000 وبلاگ فارسی بدست آمده بود ادغام کرده‌ایم.

اما آنچه که به عنوان ابتکار در این بخش صورت گرفته است توجه بیشتر به توالی کلمات پر تکرار است. تجربه نشان می دهد که یک توالی طولانی از کلمات پر تکرار خود یک عبارت با ارزش است. مثلا «بودن یا نبودن مسئله اینست» یا «در این درگه که گه گه که که و که که شود ناگه». در این پروژها توالی های بیش از 4 کلمه را به عنوان یک عبارت کاندید ستاره دار در نظر می گیریم. و با عبارات کاندید ستاره دار مانند یک کلمه کلیدی استخراج شده توسط تابع امتیازدهی برخورد می کنیم.

روش حذف کلمات پرتکرار به این نحو است که در صورت انطباق کلمه ای با یکی از کلمات لیست ، یک توالی به طول یک تشکیل شده و ادامه رشته کلمات برای انطباق بیشتر پردازش می شود و هر انطباق به

[1] زیر رشته اول از نویسه فاصله تا قبل از اولین نویسه «_» و زیر رشته دوم بعد از آخرین نویسه «_» تا رسیدن به اولین فاصله در نظر گرفته شده است

این توالی اضافه شده و طول افزایش می یابد با اولین عدم انطباق طول توالی بررسی خواهد شد. در صورتیکه طول توالی بیش از 3 کلمه باشد توالی به عنوان یک عبارت کاندید ستاره دار ثبت می شود و در غیر اینصورت به عنوان یک یا چند عبارت پر تکرار حذف خواهد شد.

4-5 تعیین مرز جملات

در این قسمت روال ساده‌ای را، که با توجه به خصوصیات زبان فارسی برای حل مشکل نقطه جهت تعیین مرز جملات طراحی کرده‌ایم، شرح می‌دهیم.

1. برای حل مشکل نقطه « . » در علائم اختصار یا اختصارات اسمی، هر نقطه ای که به یک تک حرف چسبیده باشد را نقطه پایان جمله حساب نمی کنیم. و در صورتیکه نقطه از هر دو طرف به یک تک حرف محدود شده باشد آن را یک علامت اختصار حساب می کنیم.
2. برای حل مشکل «.» در آدرس های وب، هرگاه نقطه‌ای از دو طرف به دنباله هایی چسبیده باشد که شامل اختصارات کلیدی وب از قبیل «.com» ، «.www» و ... باشد، در این صورت آن دنباله را یک آدرس وب تلقی می کنیم.
3. در صورتیکه هریک از رخدادهای بالا اتفاق بیفتد دنباله بوجود آمده تفکیک نشده و به عنوان یک واژه رها می‌شود در غیر اینصورت رشته نویسه های سمت چپ نقطه تا رسیدن به آخرین نقطه دیده شده به عنوان یک جمله حساب می شود.

5-5 تعیین مرز کلمات

برای توکنیزه کردن کلمات به علت حل تعداد زیادی از مشکلات در فرایند یکسان‌سازی از هیچ الگوریتم خاصی استفاده نکرده ایم و به تفکیک با استفاده از نویسه های رایج بسنده کرده ایم. تنها در چند مورد فرایندهای ساده ای را اجرا کرده ایم.

1. در زبان فارسی کلمه تک حرفی با معنی غیر از «و» وجود ندارد به همین علت در صورت برخورد با کلمات تک حرفی در صورتیکه کلمه پیش از آن ها به حرف از راست پیوند ناپذیر[1] ختم شده باشد کلمه تک حرفی به عنوان حرف آخر کلمه قبل در نظر گرفته شده است و در غیر اینصورت به عنوان یک اشتباه حذف شده است.
2. برای نشانه «/» کلمات جانشین به عنوان دو کلمه هم اتفاق[2] در نظر گرفته شده اند و تفکیک نشده اند این مسئله باعث می شود که در آدرس ها و تاریخ ها نیز تفکیکی صورت نگیرد. در واقع نویسه «/» به عنوان یک نویسه معمولی در نظر گرفته شده است.
3. برای نشانه «،» در صورتیکه در هر دو طرف نشانه عدد باشد از حذف نشانه چشم پوشی کرده و کلمات و نشانه بینشان را به عنوان یک کلمه یکسان اعلام می‌کنیم.

باید توجه داشت که با توجه به استفاده از قالب های 1 تا 4 کلمه ای و نیز نوع مسئله استخراج کلمات کلیدی که بدون توجه به ساختارهای نحوی نیز با کیفیت خوبی قابل انجام است، از هر نوع تشخیص هم-آیی کلمات ، افعال مرکب و ... چشم پوشی کرده ایم. این چشم پوشی تاثیر چندان زیادی بر نتیجه کلی نخواهد داشت.

[1] الف – د – ذ – ر – ز – ژ – و

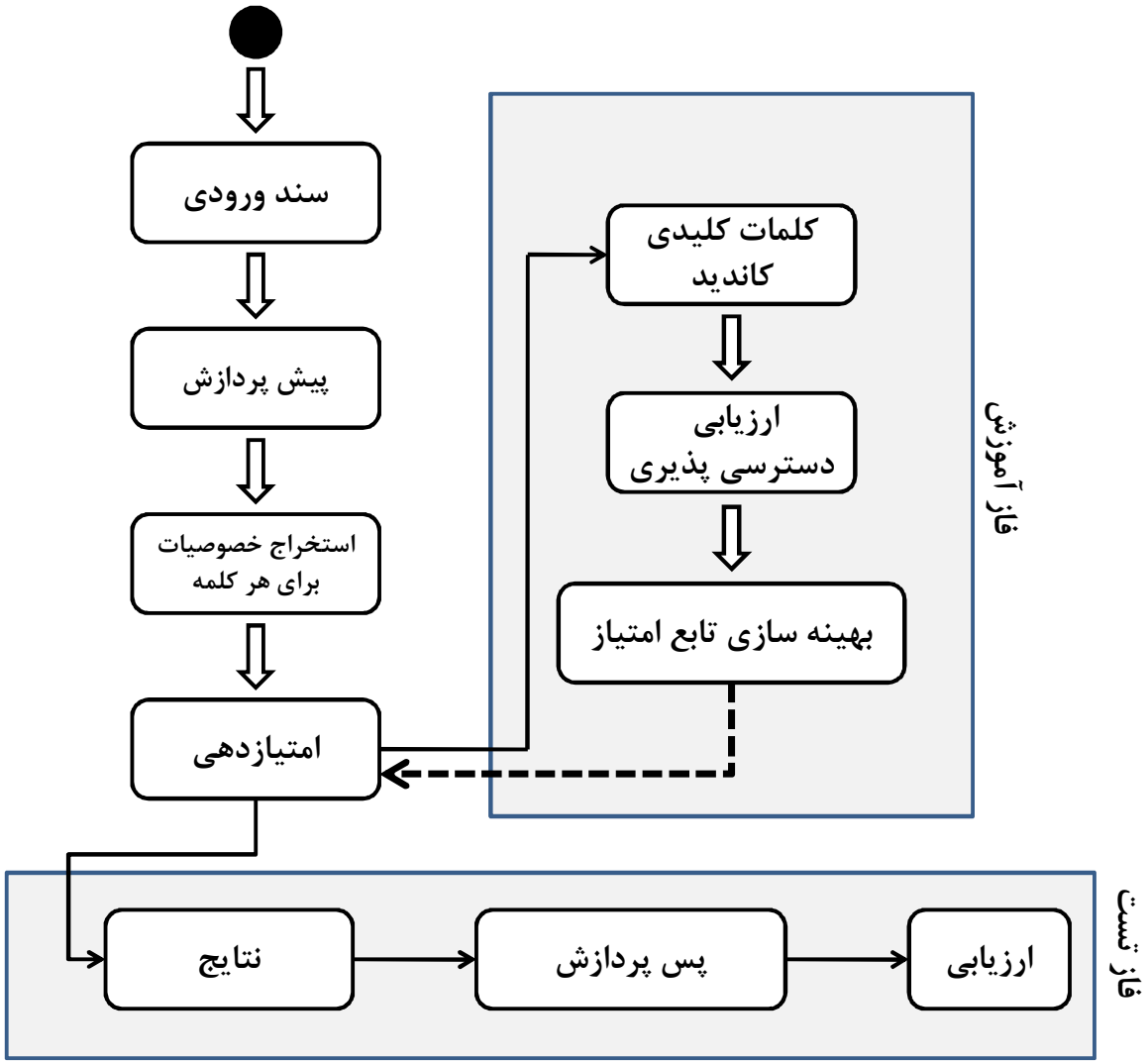
[2] Co-ocurrence

فصل ششم:

استخراج کلمات کلیدی

6-1 مقدمه

شکل 6-1 شمای کلی روش ارائه شده در این پروژه برای استخراج کلمات کلیدی را نشان می دهد فاز استخراج کلمات کلیدی شامل دو بخش می باشد: بخش آموزش و بخش تست



شکل 6-1 شمای کلی روش ارائه شده در این پروژه برای استخراج کلمات کلیدی

روش ارائه شده در این پروژه از دو مجموعه خصوصیات و دو تابع ارزیابی استفاده می کند. تابع اول که به نام تابع امتیاز دهی نامگذاری شده با استفاده از مجموعه خصوصیات1 به هر یک از توکن های استخراج شده در بخش قبل امتیازی می دهد. بعضی از مجموعه خصوصیات1 خصوصیات جمعی هستند و به همین علت این مجموعه خصوصیات وابستگی مستقیم به پایگاه داده دارند.

تابع دوم که به نام تابع ارزیابی نتایج موتورهای جستجو[1] (SERE) معرفی شده است تابعی است که بر اساس مجموعه خصوصیات2 که از یک سند منفرد استخراج می شوند براساس نتایج موتورهای جستجو به کلمات امتیاز می دهد. در واقع بازخورد اهمیت کلمه انتخاب شده توسط تابع امتیاز دهی در تابع SERE نمایان می شود. تابع امتیاز دهی به این نحو طراحی شده طراحی شده است که ضرایب آن توسط الگوریتم ژنتیک بهینه سازی می شود. تابع ارزیابی الگوریتم ژنتیک مورد استفاده برای بهینه سازی برپایه SERE بنا نهاده شده و جمعیت را ضرایب استفاده شده در تابع امتیاز دهی معرفی کرده ایم. فرایند استخراج کلمات کلیدی طوری پیش می رود که ضرایب تابع امتیاز دهی در طی انجام پروسه ی آموزش به نحوی تنظیم شوند، که ماکزیمم امتیاز در تابع SERE توسط کلماتی که ماکزیمم امتیاز را در تابع امتیاز دهی بدست می آورند، کسب شود. در ادامه این فصل ابتدا نگاهی داریم به بخش های آموزش و تست سپس به مرور کامل این دو تابع، مجموعه های خصوصیات و سایر توابع و فرایندهای مورد استفاده در پروژه می پردازیم.

2-6 بخش آموزش

در این بخش هر دو تابع امتیازدهی و SERE مورد استفاده قرار می گیرند.

[1] Search engines results evaluation (SERE)

در این پروژه از پنجره‌های 1تا4 کلمه ای استفاده کرده ایم به دلیل آنکه تحقیقات جامعی در مورد پیمانه-های[1] کلمات در فارسی وجود ندارد به همین علت از یک پنجره ساده استفاده کرده‌ایم. در مرحله اول این بخش برای هریک از پنجره های n تایی خصوصیات مجموعه1 برای استفاده در تابع امتیاز دهی استخراج می‌شود. خصوصیات مجموعه اول، گزیده‌ای از خصوصیات تعریف شده در تحقیقات دیگر هستند که تاثیرگذاری بیشتری در مسئله تشخیص کلمات کلیدی داشته اند. پس از مرحله امتیازدهی توکن‌ها بصورت صعودی از نظر امتیاز مرتب می شوند و با استفاده از رابطه 1-6 کلمات کاندید استخراج می شوند

$$Score(word_{i,d}) > thershold \qquad 1\text{-}6$$

مقدار threshold یک مرز امتیازی است تا کلماتی که بیش از این مرز امتیاز می‌گیرند به عنوان کلمات کلیدی تلقی شوند. این مرز به نحوی تعیین می‌شود که حداقل 60% از کلمات انتخاب شده کاربران در این لیست دیده شود.

پس از آنکه لیست کلمات کاندید ساخته شد این لیست به ماژول ارزیابی و بهینه سازی تحویل می‌شود. ماژول ارزیابی و بهینه سازی ابتدا لیست را برای موتورهای جستجوی محبوب مانند گوگل، یاهو، MSN و... فرستاده و صفحه اول نتایج را به عنوان بازخورد دریافت می کند. صفحه اول نتایج با استفاده از SERE مورد تحلیل قرار می گیرد.

ماژول ارزیابی و بهینه سازی با استفاده از الگوریتم ژنتیک به بهینه سازی ضرایب موجود در تابع امتیازدهی می پردازد و این فرایند آموزش تا رسیدن به بیشینه رسیدن تابع ارزیابی ادامه می یابد.

[1] N-gram

6-2-1 تابع امتیاز دهی

تابع امتیاز دهی مورد استفاده در رابطه 6-2 نشان داده شده است. این تابع هم در بخش آموزش و هم در بخش تست مورد استفاده قرار می گیرد. اما در بخش آموزش با استفاده از الگوریتم ژنتیک بهینه سازی می شود.

$$\text{Scoring function } (w) = NPW*(\alpha*NPLen + \beta*NSL + \gamma*NTFIDF)/3 \qquad 6\text{-}2$$

در ابتدا تمامی ضرایب α، β و γ برابر با یک عدد تصادفی در بازه $[0.4,0.6]$ قرار داده می شوند اما در طول فاز آموزش هریک از آن ها به سمت ماکزیمم کردن تابع ارزیابی پیش می روند که این منجر به بهتر شدن امتیاز کلماتی خواهد شد، که از نظر موتورهای جستجو مهمتر تلقی می شوند. همچنین به خاطر نوع تعریف مقدار threshold می توانیم تاحدود زیادی از بدست آوردن یک امتیاز بازخوانی خود مطمئن باشیم.

خصوصیات مورد استفاده به عنوان پارامترهای تابع امتیاز دهی که خصوصیات مجموعه اول را تشکیل می‌دهند به شرح زیر هستند:

6-2-1-1 NPW نرمالیزه شده تعداد کلمات

NPW[1] تعداد کلمات یک عبارت است که با تقسیم بر ماکزیمم تعداد کلمات پیمانه نرمالیزه شده است. از آنجا که حداکثر پیمانه مورد استفاده در این تحقیق 4 بوده پس مقادیری که این متغیر می تواند بدست آورد برابر با 1 برای عبارت مورد بررسی چهار کلمه ای ، 0,75 برای عبارت سه کلمه ای، 0,5 برای یک عبارت دو کلمه‌ای و 0,25 برای عبارت یک کلمه ای است. انگیزه استفاده از این خصوصیت بصورت یک

[1] Normalized phrases words

ضریب برای سایر امتیازها، آنست که امتیاز بیشتری به عبارات چهار کلمه ای که با ارزش تر از عبارات سه کلمه ای هستند داده شود(همین طور این مسئله در مقایسه با یک کلمه ای و ...).

6-1-2-2 NPLen طول عبارت بصورت نرمالیز شده

NPLen [1] بصورت تعداد کلمات یک عبارت کاندید تقسیم بر تعداد کلمات جمله ای که شامل آن می‌باشد بدست می آید. این خصوصیت در صورتیکه کل جمله یک عبارت کاندید باشد برابر 1 می شود. انگیزه استفاده از این خصوصیت آن است که، عبارت های موجود در عنوان و زیر عنوان ها به علت کوتاه بودن جملاتشان، امتیاز بیشتری بگیرند.

6-1-2-3 NSL امتیاز جایگاه جمله بصورت نرمالیزه شده

در بیشتر ارزیابی‌ها و روش های استخراج کلمات کلیدی ادعا شده است که کلماتی که در ابتدا یا انتهای یک متن وجود دارد به طور نسبی اطلاعاتی بیشتری در مورد اصل موضوع آن متن در خود دارند، به همین علت خصوصیت NSL [2] با استفاده از رابطه 6-3 طوری تعریف شده است که جملات ابتدایی و انتهایی در متن امتیاز بیشتری بگیرند.

$$NSL = \left((2 * (\frac{L}{m}) - 1 \right)^2 \qquad 6\text{-}3$$

در رابطه 6-3 «L» نمایانگر رتبه جمله در متن و m به معنای تعداد جملات موجود در کل متن است. مقدار NSL برای اولین جمله (L=0) و آخرین جمله (L=m) برابر با 1 و هرچه به مرکز متن نزدیک‌تر شویم کمتر از 1 می‌شود.

[1] Normalized Phrase Length
[2] Normalized Sentence Location

NTFIDF 4-1-2-6

TFIDF یکی از مهمترین روش‌های وزن دهی به کلمات است. این ویژگی در بسیاری از روش های وزن-دهی مهم به عنوان یک پایه محسوب می شود[46]. برای مقید کردن این خصوصیت به بازه [1و0] آن را با استفاده از رابطه 4-6 نرمالیزه کرده ایم.

$$NTFIDF(word) = (\frac{WF}{MF}) * \log_2(\frac{N+1}{n+1}) \qquad 4\text{-}6$$

WF تعداد تکرار کلمه در این سند و MF برابر با تعداد تکرار مکررترین کلمه سند است. N و n همانند آنچه در فصل قبل تعریف شد برابر با تعداد کل اسند و تعداد اسناد شامل این کلمه می باشند.

2-2-6 تابع ارزیابی

این تابع در فاز آموزش به عنوان تابع ارزیابی الگوریتم ژنتیک مورد استفاده قرار می گیرد. از دید تکنیکی این تابع در واقع یک تابع وزن دهی به کلمات در یک سند منفرد به شمار می رود. از آنجا که بازخوردهای موتورهای جستجو صفحات وب هستند، استفاده از روش های آنالیز اهمیت عبارات با استفاده از برچسب-های HTML برای کمک به وزن دهی بهتر در این تابع نیز امکان پذیر است.

تابع SERE در رابطه 5-6 نشان داده شده است.

$$SERE = PRF*(TET+NLCT+NHCT+NIRT) \qquad 5\text{-}6$$

پارامترهای تعریف شده در پرانتز در رابطه 5-6 بر اساس برچسب های HTML بدست آمده اند. با هم به مرور پارامترهای تابع SERE می پردازیم.

6-2-2-1 PRF تعداد تکرار نسبی

PRF [1] نشان دهنده تعداد تکرار کلمه در سند است که با تقسیم بر تعداد تکرار مکررترین کلمه نرمالیزه شده است. این پارامتر در واقع همان TF است که بالاترین مقدار یعنی 1 را در مکررترین کلمه گرفته و برای سایر کلمات ارزشی کمتر از 1 خواهد داشت.

6-2-2-2 TET وجود کلمه در عنوان

TET[2] برابر است با تعداد کلمات عبارت تقسیم بر تعداد کلمات موجود در عنوان سایت که با تگ <title> دیده می‌شود.

6-2-2-3 NLCT تعداد لینک های شامل عبارت

NLCT [3] تعداد وقوع عبارت در برچسب < a> را نشان می دهد. برای نرمالیزه کردن این عدد را بر تعداد کل لینک های موجود تقسیم می کنیم.

6-2-2-4 NHCT تعداد تیترهای شامل عبارت

NCHT نشان دهنده تعداد وقوع عبارت در برچسب <h1> در نظر گرفته شده است. برای نرمال کردن این عدد آن را بر تعداد کل برچسب های <h1> تقسیم می کنیم.

6-2-2-5 NIRT تعداد تصاویر مرتبط با عبارت

NIRT [4] تعداد تصاویر مرتبط با عبارت مورد ارزیابی را نشان می‌دهد. در تگ Img که نشان دهنده تصاویر است در HTML است دو صفت[5] مهم وجود دارد:

[1] The Phrase Relative Frequency
[2] Term Exist in Title
[3] Number of Links Containing the Term
[4] Number of Images Related with the Term
[5] Attribute

Src: که آدرس تصویر را در خود دارد. (از این صفت به علت آنکه آدرس و نام فیزیکی تصاویر در بیشتر موارد به زبان انگلیسی است استفاده نکرده ایم.)

Alt: این صفت همانطور که در فصل سوم توضیح داده شد به عنوان توضیحی بر تصویر به کار می‌رود.

مقدار NIRT برابر با تعداد وقوع عبارت در تگ Img تقسیم بر تعداد کل تگ های Img موجود است.

در این تحقیق به همین چهار برچسب HTML که مهمترند بسنده کرده ایم اما می توان برای دقت بیشتر با برچسب های دیگر نیز به این مسئله پرداخت.جدول 6-1 تعدادی از برچسب های HTML به همراه توضیح مختصری برای آن ها و همچنین مثالی از کاربرد را نشان می دهد.

جدول 6-1 نمونه ای از برچسب های HTML به همراه توضیح و مثال

برچسب	توضیح	مثال
<html>	برای نشان دادن شروع و پایان کل یک صفحه وب به کار می رود	<html> All other content goes here... </html>
<head> <title>	این برچسب برای نشان دادن عنوان صفحه وب به کار می رود و تقریبا مهمترین برچسب بشمار می رود.	<head> <title>Welcome to my site</title> </head>
<body>	برای مشخص کردن قسمت اصلی متن یک صفحه استفاده می شود	<body> Page body goes here <body/>
<a>	برای نشان دادن لینک به صفحات دیگر به کار می رود	Visit the my site

	توجه کنید که این برچسب برچسب مهمی است که وجود کلمه ای چه در قسمت آدرس آن (herf) و چه در قسمت توضیح آن می تواند اهمیت زیادی داشته باشد	
<img>	برای درونی سازی یک تصویر به کار می رود	
  <u>	به ترتیب از بالا برای تیره سازی[1] کج نویسی[2] و نوشته های زیر خطدار به کار می روند.	This is bold text while this text is in italic
<table> <tr> <td>	این برچسب ها در ساخت جدول ها به کار می روند. tr سطرهای جدول و td ستون های آن را ایجاد می کنند. از آنجا که معمولا اطلاعات ارائه شده در جداول اطلاعاتی خلاصه و مفید هستند. این برچسب ها در سیستمهای امتیاز دهی مبتنی بر صفحات وب می توانند با ارزش تلقی شوند	<table> <tr> <td>This is a Cell in Column 1</td> <td>This is a Cell in Column 2</td> <tr/> <table/>
<h1>	این برچسب که مشخص کننده تیتر است می تواند تا زیر تیترهای درجه 6 ادامه پیدا کند بسته به کاربرد می توان برای هر درجه از تیتر ارزش خاصی قائل شد.	Hello World
    	تگ هایی که برای ساخت لیست های شماره‌دار و بی شماره به کار می روند. از آنجا که لیست ها معمولا شامل اطلاعات مرتبط به هم هستند این	 - First item with a bullet - Second item with a bullet

[1] Bold
[2] Italic

	برچسب ها را می توان به نوبه خود با اهمیت تلقی کرد.	 - First item with a number - Second item with a number

3-2-6 الگوریتم ژنتیک اجرا شده

الگوریتم ژنتیک یک روش جستجوی تصادفیست که بر پایه انتخاب طبیعت و ارزیابی بنا نهاده شده است. یک GA برای حل مسئله، مجموعه بزرگی از راه حل ها را تولید می کند و هر راه حل را با استفاده از یک "تابع تناسب[1]" ارزیابی می کند. آنگاه تعدادی از بهترین راه حل ها با استفاده از اپراتور های ژنتیکی باعث تولید راه حل های جدیدتری می‌شوند این کار تا رسیدن به جواب مطلوب ادامه پیدا می کند. به این ترتیب فضای جستجو در جهتی تکامل پیدا می کند که به راه حل مطلوب برسد. به طور عمومی اپراتورهای ژنتیکی به در دو دسته آمیزش و جهش تقسیم می شوند و هریک دارای انواع مختلفی می-باشند. می توان ادعا کرد در صورت انتخاب صحیح پارامترها الگوریتم ژنتیک می تواند بسیار موثر عمل کند. در واقع ژنتیک الگوریتم به جای جستجو راه حل ها راه حل ها را با استفاده از ترکیب راه حل های فعلی می سازد و در هر بار نسل جدید را جایگزین بخشی از نسل قدیم می کند.

شبه کد شکل 6-2 الگوریتم ژنتیک را بصورت مختصر شرح می دهد:

[1] Fitness function

```
Function genetic algorithm (population, fitness_fn,cross_over, mutation)
return an individual
Input: population a set of individual
Fitness_fn: a function that measure the fitness of an individual
Cross_over: is a percent of population that use for child produce (cross over)
Mutation: is a percent of population that use for mutation
Population<- InitPopulation();
Repeat
new_popution <-empty set;
crossoverSize<-crossSize(Cross_over,population.size);
mutationSize<-mutationSize(Mutation,population.size);
new_population<- generatenewpopulation(population,
crossoverFunction(crossoverSize,population))
mutationFunction(new_population,mutationSize);
new_population<-bestchildof(population);
population<-new_population;
Until some individual is fit enough, or enough time has
elapsed return the best individual in population.
```

شکل 6-2 شبه کد الگوریتم ژنتیک

6-2-3-1 ویژگی های الگوریتم ژنتیک

- الگوریتم های ژنتیک در مسائلی که فضای جستجوی بزرگی داشته باشند می‌تواند بکار گرفته شود. همچنین در مسایلی با فضای فرضیه پیچیده که تاثیر اجزا آن در فرضیه کلی ناشناخته باشند می‌توان از GA برای جستجو استفاده نمود.
- در بهینه سازی گسسته بسیار مورد استفاده قرار می‌گیرد.
- الگوریتم های ژنتیک را میتوان براحتی بصورت موازی اجرا نمود از اینرو می توان کامپیوترهای ارزان قیمت تری را بصورت موازی مورد استفاده قرار داد.
- امکان به تله افتادن این الگوریتم در مینیمم محلی کمتر از سایر روشهاست.

- از لحاظ محاسباتی پرهزینه هستند.
- تضمینی برای رسیدن به جواب بهینه وجود ندارد.

6-2-3-2 خصوصیات الگوریتم ژنتیک پیاده سازی شده

همانطور که در شکل 6-2 نشان داده شده است. تابع الگوریتم ژنتیک ارائه شده دارای چهار پارامتر است:

1. جمعیت: برای این پارامتر مجموع ضرایب را بصورت یک رشته 24 بیتی در نظر گرفته ایم و از آنجا که 3 ضریب بایستی مقدار دهی شوند هر ضریب یک بایت را اشغال می کند.
2. تابع تناسب: این تابع بر طبق تابع SERE به صورت رابطه 6-6 تعریف می شود

Fitness function =Se.1weight*SERE (word in Se1.result)
+ Se.2weight*SERE (word in Se2.result)
.
.
.
+ SeN.weight*SERE (word in SeN.result)

6-6

که در این تابع برای هر موتور جستجوی دلخواه 1تاn وزن دلخواهی می توان در نظر گرفت. استفاده از این وزن دلخواه باعث می شود که بتوانیم برای موتورهای گوناگون میزان اهمیت متفاوت قائل شویم.

3. اپراتور ژنتیک استفاده شده برای آمیزش در این مسئله، یک تابع ساده آمیزش است که نیمی از بایت های ضریبی را با نیم بایت همان ضریب در جمعیت دیگر ترکیب می کند.
4. اپراتور ژنتیک برای جهش: برای تاثیر گذاری منطقی اپراتور جهش روی بیت های 3تا5 هر بایت (برای هر ضریب) بصورت تصادفی بعد از هر 100 بار آمیزیش عمل می کند.

از آنجا که تابع تناسب ارائه شده در این بخش ارزش یک کلمه کلیدی را در نتایج موتورهای جستجو ارائه می کند از این تابع تناسب به عنوان یک معیار دیگر سنجش کارایی استفاده کرده ایم و آن را **میزان دسترسی پذیری** معرفی می کنیم.

3-6 فاز تست

در این فاز دیگر از تابع SERE استفاده نمی کنیم و ضرایب هر آنچه که در مرحله قبل بدست آمده باشد ثابت می ماند.

6-3-1 استخراج نتایج

در این بخش یک سند ورودی پس از انجام مراحل پیش پرداز و امتیاز دهی کلمات توسط تابع با استفاده از مرز بدست آمده در مرحله قبل لیستی از کلمات را به عنوان کلمات کلیدی کاندید مطرح می‌کند. این لیست تعداد کلمات زیادی دارد و به خاطر فرایند استخراج و مرز تعیین شده از میزان بازخوانی خوبی برخوردار است اما میزان دقت این لیست در مقدار پایینی قرار دارد. در واقع اگر به دنبال کسب نتایج مناسب در یک موتور جستجو هستیم می‌توان کار را به این لیست یا تعداد مناسبی از کلمات بالای لیست بسنده کرد اما اگر به دنبال ملاک های علمی تر و نه عملی تر دقت و بازخوانی هستیم به طوریکه فرایند استخراج به رفتار انسانی شبیه تر باشد استفاده از فرایند پس پردازش بعد ضروری به نظر می‌رسد.

6-3-2 پس پردازش برای بهبود میزان دقت با استفاده ازکلمات برجسته ساز

همانطور که قبلا هم ذکر شد انتخاب کلمات براساس رفتار انسانی باعث افزایش میزان دقت و بازخوانی در نتایج خواهد شد. در این مرحله با استفاده از کلمات برجسته ساز[1] روش جدیدی را برای گزینش کلمات از یک لیست کلمات کاندید جهت بهبود میزان دقت ارائه داده ایم.

کلمات برجسته ساز عباراتی از قبیل :« در این مقاله» هستند که معمولا در همان جمله شامل این عبارت قبل یا بعد از آن، عباراتی که ایده اصلی یک متن را در خود دارند می آید. یکی از محاسن استفاده بیشتر از حوزه های نزدیک به این کلمات برای استخراج کلمات کلیدی می تواند کاهش بسیار زیاد محاسبات به علت محدود شدن ناحیه های جستجو و امتیازدهی باشد.

تعریف ما از کلمات برجسته ساز اینست:

کلماتی که احتمال حضور کلمات کلیدی منتخب کاربر انسانی در همسایگی آن ها بیشتر باشد.

در ادامه به شرح روش ابتکاری استخراج این کلمات و نیز نحوه به کارگیری آن ها می پردازیم اما برای شروع ابتدا بهتر است تعریف تعدادی اصطلاح که در ادامه متن زیاد به کار گرفته می شوند را تعریف کنیم.

همسایگی چپ یا LN[2] : به کلمات واقع شده در سمت چپ یک کلمه مشخص x (در اینجا یک کلمه برجسته ساز) همسایگی چپ آن کلمه می‌گوییم. رابطه همسایگی چپ در یک زبان از «راست به چپ نویس» مثل فارسی، از کلمه قبل از x شروع شده و تا پایان جمله قبل ادامه می یابد.

[1] Attention attraction strings
[2] Left neibor

همسایگی راست یا RN [1]: کلمات واقع شده در سمت راست یک کلمه مشخص y (در اینجا یک کلمه برجسته ساز) را همسایگی راست آن کلمه می گوییم. رابطه همسایگی راست در یک زبان از «راست به چپ نویس» از کلمه بعد از y شروع شده و تا انتهای جمله ادامه می یابد.

کلمات هم رخداد[2]: کلماتی که در همسایگی با یک کلمه کلیدی خاص و نه همه کلیدی حضور دارد. به عنوان مثال کلمه اگر در مجموعه متون آموزش ما عبارت «فرشچیان» یک کلمه کلیدی منتخب انسانی باشد. کلمه «استاد» یک کلمه هم اتفاق محسوب می شود نه یک کلمه برجسته ساز، چرا که به رغم حضور در همسایگی کلمه کلیدی تنها به همسایگی یک کلمه کلیدی خاص محدود می‌شود.

6-3-2-1 استخراج کلمات برجسته ساز

شکل 6-3 روند استخراج کلمات برجسته ساز در این پروژه را نشان می دهد. ابتدا تمامی کلماتی که در مجموعه آموزش، یک رابطه RN یا LN با یک کلمه کلیدی برقرار می کنند را لیست کرده ایم و آن‌ها را به عنوان کلمات برجسته ساز احتمالی برچسب گذاری می کنیم. اصطلاح احتمالی به این خاطر است که این کلمات ممکن است با کلماتی که با یک کلمه کلیدی خاص هم‌رخداد باشند اشتباه گرفته شوند. برای جلوگیری از این تداخل، لیست کلمات برجسته ساز احتمالی را با استفاده از رابطه 6-7 لیست را امتیازدهی کرده ایم.

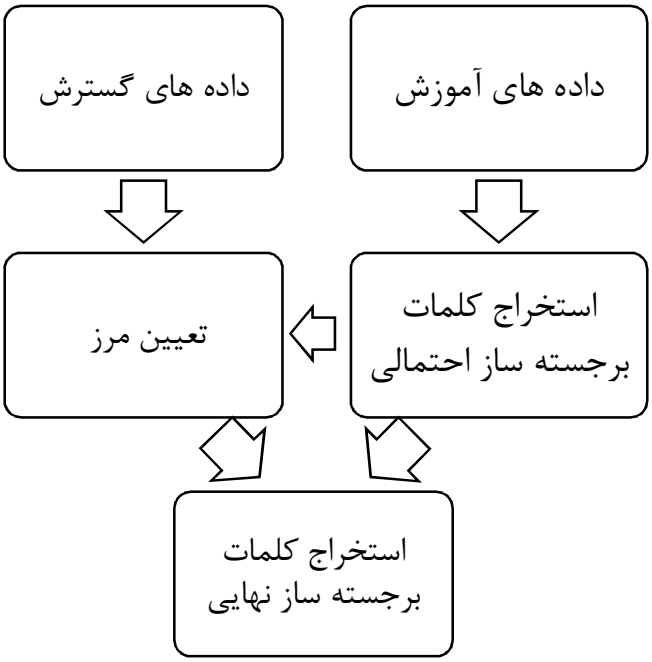
$$AAS - score(w) = \frac{D_W}{D_T} * \frac{NRN_W + NLN_W}{NWO_W} \qquad 6\text{-}7$$

- D_W: تعداد اسنادی که شامل کلمه w هستند.
- D_T: تعداد کل اسناد مورد تحقیق

[1] Right niebor
[2] Co-ocurrence

- NRN: تعداد دفعاتی که کلمه برچسب RN گرفته است.
- NLN: تعداد دفعاتی که کلمه برچسب LN گرفته است
- NWO: تعداد دفعات وقوع کلمه

از آنجا که کلمات برجسته ساز کلمات عمومی تری نسبت به کلمات هم‌رخداد محسوب می‌شوند قسمت اول رابطه 6-7 باعث می شود که این کلمات امتیاز بهتری نسبت به کلمات هم‌رخداد بگیرند و قسمت دوم مطابق با تعریف باعث می شود که کلماتی که در همسایگی یک کلمه کلیدی قرار می‌گیرند از امتیاز بالاتری برخوردار شوند.



شکل 6-3 روند استخراج کلمات برجسته ساز

بعد از این مرحله نیاز به تعریف یک مرز امتیازی برای تشخیص کلمات هم‌رخداد و کلمات برجسته‌ساز از یکدیگر است. اگر مرز تعریف شده در حد پایینی باشد کلمات هم‌رخداد نیز جزو کلمات برجسته‌ساز محسوب شده و نتایج را مخدوش می کنند و اگر این مرز در حد بالایی باشد ممکن است برخی از کلمات

برجسته ساز را از دست بدهیم. به همین خاطر به نوعی بهینه‌سازی برای تعیین این مرز نیازمندیم. برای این منظور تابع مرز را بصورت رابطه 6-8 تعریف می کنیم.

$$KWNA(threshold) = \frac{Number\ of\ \ keywords\ with\ no\ AAS\ neighbor}{Number\ of\ \ total\ keywords} \qquad 8\text{-}6$$

این تابع یک تابع گسسته است که دامنه آن را مرز انتخابی(threshold) در بازه [0,1] و برد آن را نسبت کلمات کلیدی که با این مرز، در مجموعه توسعه[1]، بدون همسایگی کلمات برجسته ساز احتمالی می ماند، به کل کلمات کلیدی، تعیین می‌کند.

مقدار مرزی که این تابع را کمینه کند یک مقدار بهینه محسوب می‌شود چرا که بیشترین میزان همسایگی بین کلمات کلیدی و کلمات برجسته ساز را در مجموعه توسعه ایجاد کرده است. برای پیدا کردن این کمینه از روش مطرح شده در مرجع [47] با استفاده از ابزار بهینه سازی[2] در نرم افزار Matlab استفاده کرده‌ایم.

6-3-2-2 گزینش کلمات کلیدی

بعد از آنکه کلماتی که امتیازی بالاتر از مرز بهینه گرفته اند را به عنوان کلمات برجسته ساز انتخاب کرده ایم و برای هریک از این کلمات احتمال حضور یک کلمه کلیدی در همسایگی چپ و راست آن هار ا با استفاده از روابط 6-9 بدست می آید.

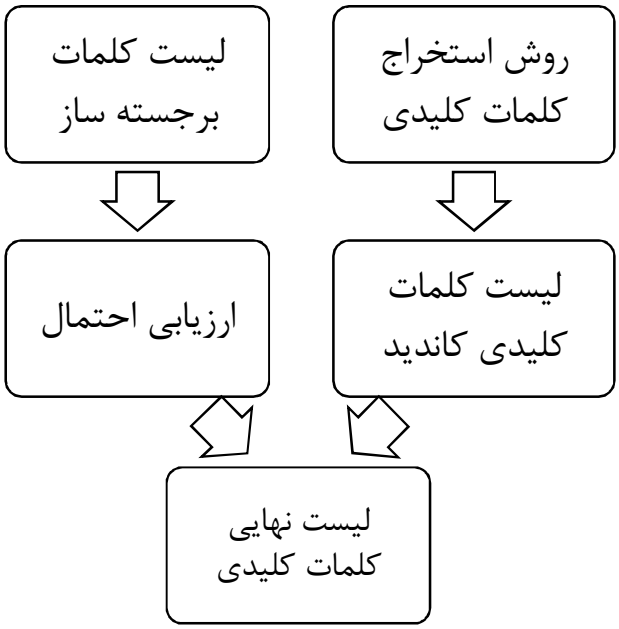
[1] Develop database
[2] Optimization tool

$$EPKLN(AAS_i) = \frac{\text{number of } AAS_i \text{ get a LN label}}{\text{total number of } AAS_i \text{ occurrance}}$$

$$EPKRN(AAS_i) = \frac{\text{number of } AAS_i \text{ get a RN label}}{\text{total number of } AAS_i \text{ occurrance}}$$

9-6

پس پردازش ارائه شده در این پروژه مستقل از شیوه استخراج لیست کلمات کلیدی کاندید عمل می‌کند و برای افزایش میزان دقت هر روشی می‌توان از آن استفاده کرد. شکل 6-4 فرایند گزینش نهایی کلمات کلیدی را نشان می‌دهد.



شکل 6-4 فرایند گزینش نهایی کلمات کلیدی

براساس قاعده احتمال کل[1] اگر $i=1,2,\ldots,m$ و B_i ها افرازی از Ω باشند برای هر واقعه دلخواه A از Ω داریم:

[1] Total Probability Theorem

$$P(A) = \sum_{i=1}^{m} P(A \mid B_i) P(B_i) \qquad 10\text{-}6$$

در صورتیکه همسایه های چپ و راست یک کلمه برجسته ساز را افراز حساب کنیم احتمال وقوع همسایه شدن یک کلمه کلیدی را می توان با استفاده از رابطه 11-6 بدست آورد.

$$PKEWN(W) = \frac{EPKLN_W + EPKRN_W}{2} \qquad 11\text{-}6$$

حال با استفاده از 11-6 تابع امتیاز دهی 12-6 را بصورت زیر تعریف کنیم

$$PS(CKW) = \sum_{j=1}^{n} PKEWN(W_j) \qquad 12\text{-}6$$

در رابطه 12-6 CKW به معنای کلمه کلیدی کاندید و n تعداد کلمات برجسته سازیست که با CKW رابطه RN یا LN دارند.

تابع امتیاز 12-6 یک تابع احتمال نیست اما هرچه خروجی این تابع بیشتر باشد احتمال اینکه کلمه ورودی، یک کلمه کلیدی منتخب کاربر باشد بیشتر است.

6-3-3 ارزیابی نهایی

بدست آوردن تمام ملاک‌های ارزیابی در حد ایده آل اگر ناممکن نباشد کاری دشوار است اما آنچه که در اینجا ارائه شده است حق انتخاب را به کاربر می‌دهد که تا ضمن کسب میزان قابل قبولی از هر ملاک به هر میزان که نیاز وجود دارد در یک ملاک خاص امتیاز بالاتری کسب کند آنچه که در فصل بعد می‌آید بررسی کارایی روش توضیح داده شده بر روی پایگاه‌داده‌ها و نتایج عملی بدست آمده از این روش است.

فصل هفتم:

نتایج عملی

7-1 مقدمه

در این فصل ابتدا به تحلیل میزان تاثیر گذاری اندازه پایگاه داده بر فرایند آموزش تابع امتیازدهی کلمات کلیدی پرداخته ایم. سپس با توجه به نتایجی که در تابع تناسب به دست آورده ایم میزان تاثیر گذاری هریک از پارامترهای در نظر گرفته شده برای تابع امتیازدهی را بدست می آوریم. همچنین میزان تاثیرگذاری روش پس‌پردازش ابتکاری نیز برروی روش ارائه شده و دو روش ساده دیگر مورد بررسی قرار گرفته است.

شایان ذکر است که در تمامی نتایجی بدست آمده، از 10 سایت اول، اولین صفحه نتایج چهار موتور جستجو Google[1]، yahoo[2]، MSN[3] و Altavista[4] با وزن های یکسان استفاده شده است. این مسئله می‌تواند در تحقیقاتی که بر روی یک موتور جستجوی خاص تمرکز بیشتری دارند بصورت استفاده صرف از این موتور جستجو یا استفاده بیشتر از نتایج ارائه شده توسط این موتور یا وزن‌دهی سنگین تر برای موتور جستجوی مطلوب مطرح شود.

7-2 بررسی تاثیرگذاری اندازه پایگاه داده

همانطور که در فصل چهارم گفته شده جهت انجام این پروژه، برای 1100 سند از پایگاه داده همشهری کلمات کلیدی انتخاب کرده‌ایم که در این میان از 600 سند برای مرحله استخراج کلمات کلیدی استفاده می کنیم.

[1] www.google.com
[2] www.yahoo.com
[3] www.MSN.com
[4] www.altavista.com

برای تعیین میزان تاثیر حجم پایگاه داده آموزش برای فرایند استخراج لیست کلمات کلیدی از 600 سندی که برای آنان کلمات کلیدی را انتخاب کرده ایم 100 سند را به عنوان مجموعه تست انتخاب کرده ایم و سپس با مجموعه های مختلف پایگاه داده به آموزش تابع امتیازدهی پرداخته ایم. جدول 7-1 نشان دهنده نتایج متفاوت با تعداد مختلف مجموعه آموزش است.

برای آنکه آزمایشات مستقل از هم باشند برای هر مجموعه آموزش بطور جداگانه کل سیستم را از ابتدا راه‌اندازی کرده‌ایم، همچنین در هر بار انتخاب مجموعه آموزش برای آنکه پاسخ‌های سیستم به مجموعه تست مستقل از اسناد باشد سندهای تست و آموزش بصورت تصادفی انتخاب شده اند.

همانطور که مشاهده می شود بهترین نتایج سیستم در حالتی بدست می آید که از 400 سند مجموعه پایگاه داده برای آموزش استفاده کرده ایم. این نتایج در مقایسه عددی با سایر تحقیقات استخراج کلمات کلیدی [35]و[48]و[44]و... نتایج قابل قبولی به شمار می‌رود اگرچه بایستی در نظر گرفت که شرایط و حتی زبان مورد تحقیق تفاوت دارد.

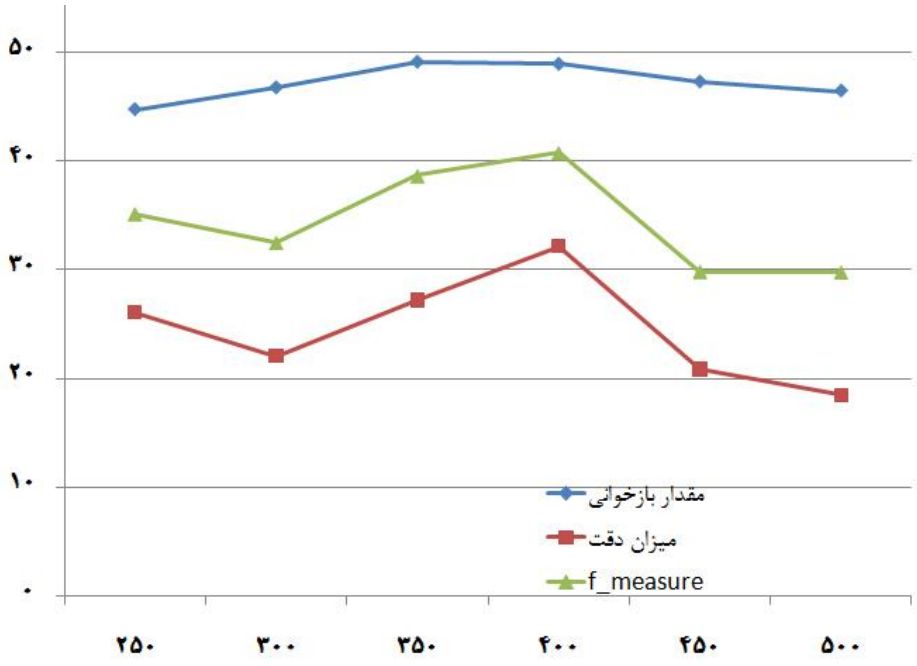
همانطور که در این نتایج می‌توان مشاهده کرد (شکل 7-1) آموزش بیش از حد باعث افزایش دنباله روی ضرایب برای تقلید از رفتار موتور جستجو شده و در نتیجه میزان دقت کاهش پیدا می‌کند. در واقع با وجود اینکه انتظار می رود افزایش اسناد مجموعه آموزش تاثیر مثبتی بر روند استخراج کلمات کلیدی بگذارد این اتفاق نیفتاده است چرا که الگوریتم موتور جستجو نیز کاملا بر فرایند انتخاب انسانی منطبق نیست و در نتیجه روند استخراج از نوع رفتار انسانی فاصله گرفته است.

هرچند به طور کلی با توجه به اعوجاجی که رفتار میزان دقت و در نتیجه آن رفتار F_measure نمی توان بطور قطع اظهار نظر کرد چراکه تعداد اسناد به کار گرفته شده در این مسئله آنقدر نیست که ثبات رفتاری توابع قابل اثبات باشد.

نکته ای دیگری که حائز اهمیت است ثابت ماندن تقریبی معیار بازخوانی با وجود ایجاد تغییرات در پایگاه داده است که البته به خاطر نوع تعریف مرزیست که ارائه کرده ایم.

جدول 7-1 تاثیرات اندازه پایگاه داده بر ملاک ارزیابی کارایی

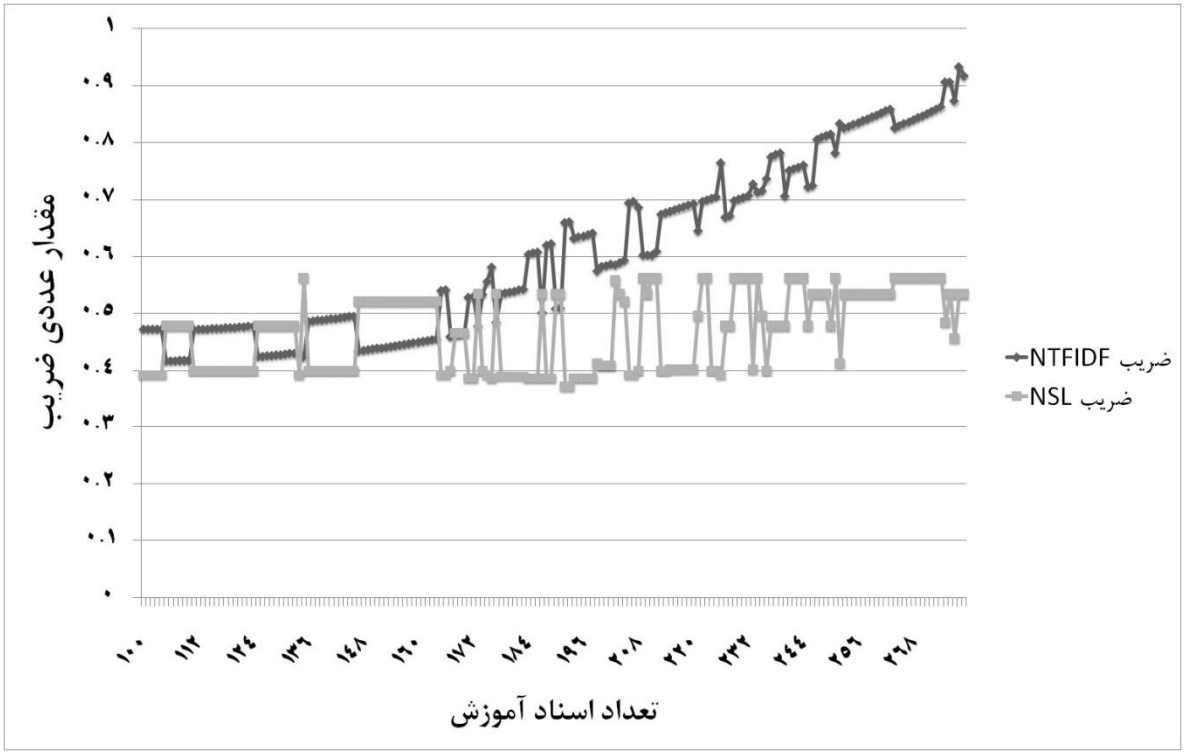
تعداد اسناد مجموعه آموزش	میانگین بازخوانی	میانگین دقت	میانگین F
250	44,77	26,14	33,18
300	46,78	22,09	34,54
350	49,14	27,26	38,51
400	48,95	32,21	40,82
450	47,32	20,9	29,86
500	46,49	18,51	29,8



شکل 7-1 نحوه تغییرات پارامترهای کارایی در اثر تغییرات تعداد اسناد پایگاه داده آموزش

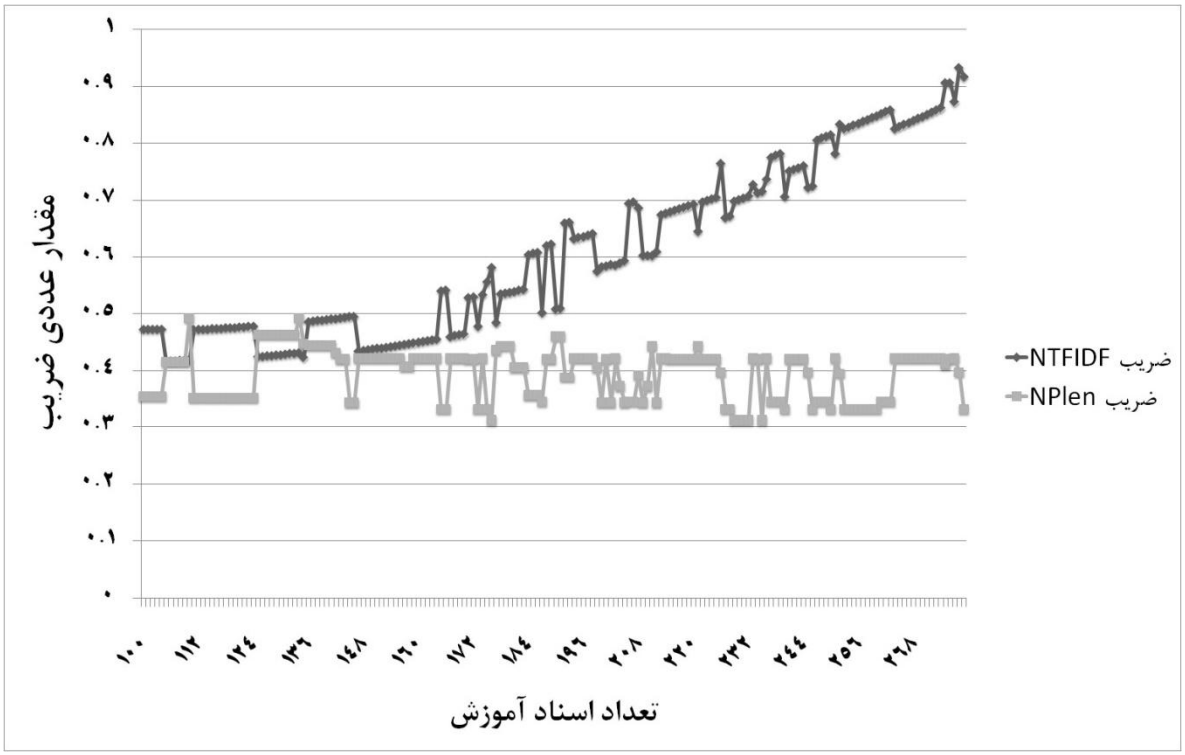
3-7 میزان تاثیر گذاری هریک از خصوصیات بر دسترسی پذیری

به نظر می‌رسد که می توان با استفاده از تطابق رشد ضرایب با توجه به صعودی مطلق بودن مقدار تابع تناسب در طول فرایند آموزش، میزان اهمیت تاثیر گذاری هریک از خصوصیات استخراج شده در این تابع را بررسی کرد (خصوصیات تعریف شده در صفحه 78 و 79). همانطور که در شکل 7-2 ملاحظه می کنید در طی فاز آموزش γ بازه بزرگتری نسبت به β می‌گیرد که این نشان از تاثیر گذاری بیشتر NTFIDF نسبت به NSL دارد.



شکل 7-2 مقایسه رشد γ (ضریب NTFIDF) با β (ضریب NSL)در طول فرایند آموزش

تاثیر گذاری γ بر میزان رشد تابع تناسب در مقایسه با α (ضریب NPLen) نیز بیشتر است. این مسئله در شکل 7-3 به وضوح نشان داده شده است چرا که بازه ارزش γ رشد سریعتر به خود می گیرد.



شکل 7-3مقایسه رشد γ (ضریب NTFIDF) با α (ضریب NPLen) در طول فرایند آموزش

7-4 نتایج فرایند پس پردازش با استفاده از کلمات برجسته ساز

برای آنکه فرایند پس پردازش استقلال کامل خود را از فرایند استخراج کلمات کلیدی حفظ کند در این مرحله از 500 سندی که در حین فرایند استخراج نقشی نداشته اند استفاده کرده ایم، 450 سند را به عنوان پایگاه داده آموزش ، 50 سند را به عنوان مجموعه توسعه [1] در نظر گرفته ایم . توجه کنید از آنجا که این روش به عنوان پیش‌پردازش بعد از استخراج کلمات کلیدی به کار گرفته می شود نیازی به مجموعه تست جداگانه وجود ندارد. برای بررسی تاثیرات میزان حجم پایگاه داده بر عملکرد فرایند پس پردازش با حجم های مختلفی به ساخت پایگاه داده پرداخته ایم و مثل مرحله قبل جهت جلوگیری از تداخل نتایج آزمایشات قبل هربار برای هر حجم پایگاه داده سیستم را مجددا بازسازی کرده ایم.

[1] Develop set

همچنین برای نشان دادن استقلال فرایند بهبود از روش استخراج، سه روش ساده دیگر براساس مراجع [40]، [44]، [39][1] طراحی و به بهبود نتایج آن ها نیز پرداخته‌ایم. روش های مذکور با نام های 1 و2و 3 در جدول آورده شده اند. برای هریک از روش‌های 1و2و3 از مجموع 250 سند برای آموزش استخراج کلمات کلیدی استفاده کرده‌ایم. از آنجا که فرایند بهبود به افزایش میزان دقت می انجامد سعی کرده ایم لیست تولید شده توسط روش های حاوی تعداد کلمات بیشتری باشد تا بازخوانی مناسبی ایجاد شود. همچنین نتایج بهبود برای روش ارائه شده در این پروژه به نام روش 4 آورده شده است برای روش 4 از تعداد 400 سند، که بهترین نتایج را به همراه داشته است، به عنوان مجموعه آموزش استفاده کرده‌ایم همچنین تعداد 100سند به عنوان داده تست برای تولید نتایج استفاده شده است.

لازم به ذکر است که این تحقیق قصد مقایسه نتایج بدست آمده از روش ها با یکدیگر را ندارد و تنها به بررسی میزان بهبود در هر روش به عنوان سندی بر درستی ادعای مستقل بودن پیش پردازش ارائه شده از روش استخراج کلمات کلیدی پرداخته است.

جدول 7-2 نتایج بدست آمده برای فرایند پس پردازش با حجم متخلف پایگاه داده طی فاز آموزش برای روش1

حجم پایگاه داده	عدم استفاده از پس‌پردازش			با استفاده از پس‌پردازش		
	بازخوانی	دقت	F_measure	بازخوانی	دقت	F_measure
200	30,67	23,52	27,54	29,16	27,91	29,2
300	28,03	25,27	26,62	25,63	32,68	29,7
400	29,73	19,96	25,39	28,34	31,64	30,35

[1] نگارنده هرگز ادعای دوباره طراحی کردن روش‌های مذکور را ندارد و سه روشی که در نتایج از آن‌ها بهره‌گیری شده تنها براساس پایه های این تحقیقات به ویژه خصوصیات استفاده شده در آن‌ها برای استخراج کلمات کلیدی بنا نهاده شده است. در نتیجه اختلافات فاحش نتایج به همین دلیل عدم تطابق طراحی می باشد استفاده از این روش ها تنها برای نشان دادن میزان بهبود است.

جدول 7-3 نتایج بدست آمده برای فرایند پس پردازش با حجم متخلف پایگاه داده طی فاز آموزش برای روش2

حجم پایگاه داده	عدم استفاده از پس‌پردازش			با استفاده از پس‌پردازش		
	بازخوانی	دقت	F_measure	بازخوانی	دقت	F_measure
200	31,89	20,87	27,11	29,67	25,4	27,42
300	27,14	18,09	22,88	25,58	25,28	25,45
400	29,52	19,7	24,32	27,27	30,21	29,33

جدول 7-4 نتایج بدست آمده برای فرایند پس پردازش با حجم متخلف پایگاه داده طی فاز آموزش برای روش3

حجم پایگاه داده	عدم استفاده از پس‌پردازش			با استفاده از پس‌پردازش		
	بازخوانی	دقت	F_measure	بازخوانی	دقت	F_measure
200	31,89	20,87	27,11	29,67	25,4	27,42
300	28,16	20,64	25,77	24,19	29,59	26,88
400	32,02	17,46	23,03	28,34	31,05	29,77

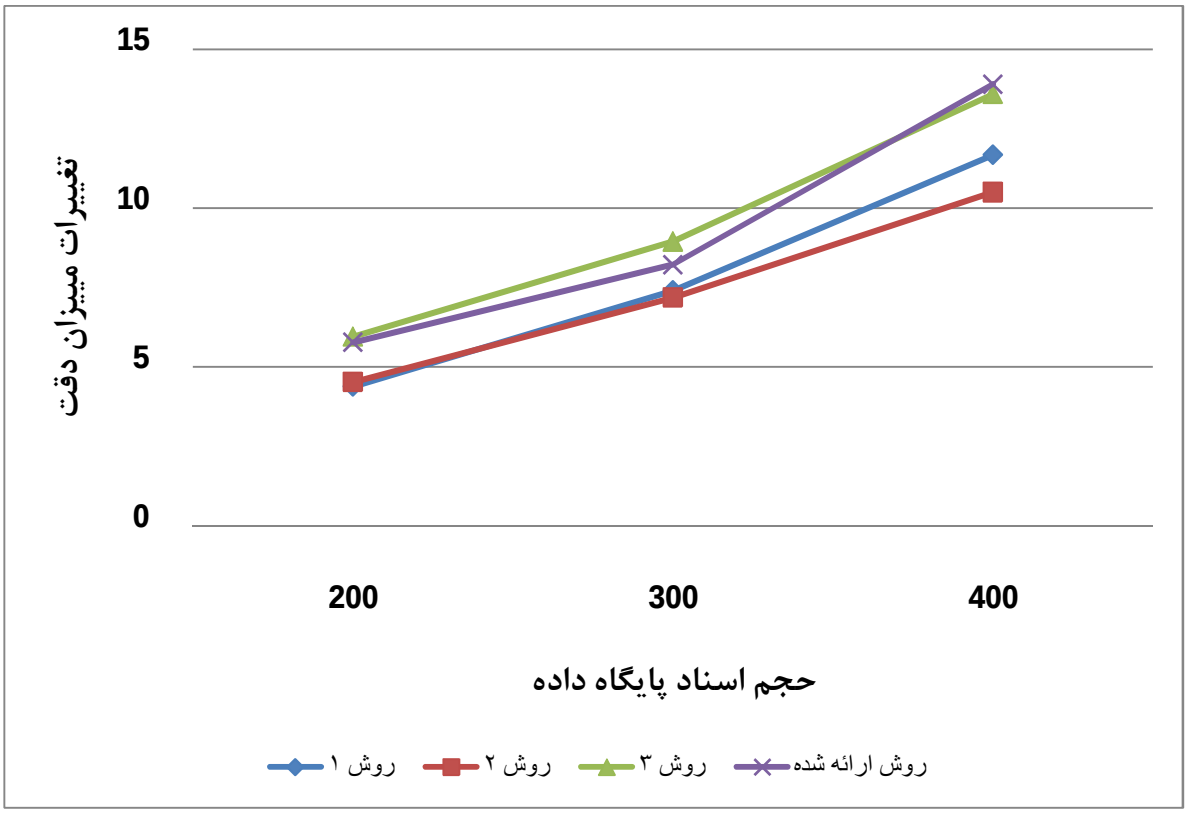
جدول 7-5 نتایج بدست آمده برای فرایند پس پردازش با حجم متخلف پایگاه داده طی فاز آموزش برای روش ارائه شده در این پروژه

حجم پایگاه داده	عدم استفاده از پس‌پردازش			با استفاده از پس‌پردازش		
	بازخوانی	دقت	F_measure	بازخوانی	دقت	F_measure
200	48,21	34,87	41,98	46,2	40,65	43,76
300	46,15	33,51	39,77	43,28	41,74	43,22
400	50,71	34,33	42,53	49,5	48,24	48,9

همانطور که در نتایج مشاهده می شود افزایش در میزان اسناد پایگاه داده در فرایند آموزش جهت استخراج کلمات برجسته ساز به طرز قابل توجهی موجب افزایش در معیار میزان دقت می شود. چراکه انتخاب هرچه دقیق تر این کلمات منجر به نزدیک شدن بیشتر گزینش کلمات به نقطه نظر انسانی می-شود. این مطلب در نمودارهای صعودی افزایش میزان دقت برحسب تعداد سند (شکل 7-4) و نیز در میزان تغییرات میانگین دقت جدول 7-6 قابل مشاهده است.

جدول 7-6 میزان تغییرات میانگین دقت در اندازه های مختلف پایگاه داده

روش	200	300	400
1	4,39	7,41	11,68
2	4,53	7,19	10,51
3	5,96	8,95	13,59
4	5,78	8,23	13,91



شکل 7-4نمودار تغییرات میزان دقت بر حسب اندازه پایگاه داده

بهترین میزان بهبود در دقت زمانی رخ می دهد که از 400 سند در پایگاه داده استفاده می کنیم. البته در این مورد با کاهش اندکی (حداکثر 3,68%) در بازخوانی روبرو هستیم . در واقع پس پردازش معرفی، در اکثر موارد کلماتی را پالایش می کند که توسط کاربران به عنوان کلمه کلیدی شناخته نشده‌اند.

در پایان در جدول 7-7 میزان تغییرات در هر سه معیار کارایی را برای بهترین حالت، یعنی زمانی که استخراج کلمات برجسته ساز با استفاده از 400 سند صورت می گیرد ارائه شده است. توجه داشته باشید که میزان منفی ثبت شده برای میزان تغییرات معیار مورد بررسی بدست آمده است نه مثدار خود ملاک زیرا مقدار تمامی ملاک های مورد بررسی همواره دربازه صفر ویک می باشد.

جدول 7-7 میزان تغییرات در متوسط ملاک های کارایی در زمان استفاده از 400 سند

روش	بازخوانی	دقت	F_measure
1	-1,39	11,68	4,96
2	-2,25	10,51	5,01
3	-3,68	13,59	6,74
4	-1,21	13,91	6,37

فصل هشتم:

نتیجه گیری و

پیشنهاداتی برای آینده

8-1 مقدمه

این فصل شامل دو بخش می باشد:

1. نتیجه گیری که در آن نتایج بدست آمده از انجام این پروژه بررسی و اعلام می شود
2. پیشنهاداتی برای آینده که به ارائه چند پیشنهاد برای تحقیقاتی می پردازد، که قصد ادامه این روش یا مقایسه روش معرفی شده با روش های دیگر را دارند.

8-2 نتیجه گیری

در این تحقیق روشی جهت استخراج خودکار کلمات کلیدی ارائه شده است که علاوه بر خصوصیات رایج، از بازخوردهای موتورهای جستجو نیز استفاده می‌کند. این روش از دو تابع امتیاز دهی و یک الگوریتم ژنتیک بهره می گیرد. تابع امتیازدهی اول یک تابع انتخاب کلمات کلیدی با استفاده از فرایندهای آماریست که ضرایب بردار خصوصیات آن قابل تغییر تعریف شده است. تابع امتیاز دهی دوم که در این تحقیق تابع ارزیابی نتایج موتورهای جستجو نامیده شده، یک تابع انتخاب کلمات کلیدی برپایه یک متن منحصر به فرد است که کلمات کلیدی بدست آمده از تابع اول را بر اساس نتایج موتورهای جستجو ارزیابی می کند. الگوریتم ژنتیک به کار گرفته شده در این روش، از یک تابع تناسب که برپایه تابع دوم بنانهاده شده، جهت بهینه سازی ضرایب تابع استخراج کلمات کلیدی کاندید می‌پردازد. نتایج بدست آمده نشان می دهد که روش ارائه شده علاوه بر انتخاب کلمات باارزش از نظر موتورهای جستجو، در دو معیار میزان دقت و بازخوانی نیز امتیاز قابل قبولی کسب می‌کند.

همچنین در این پروژه به تعریف کلمات برجسته ساز پرداخته ایم و برای چگونگی استخراج آن ها روش جدیدی به کارگرفته شده است. کلمات برجسته ساز طبق تعریف کلماتی هستند که احتمال وقوع یک کلمه کلیدی در همسایگی آن ها بیشتر است. پس از استخراج کلمات برجسته ساز با استفاده از قاعده احتمال کل ، پس‌پردازش جدیدی جهت افزایش معیار دقت ارائه گردیده است. بررسی های عملی نشان می‌دهد که پس‌پردازش ارائه شده با کمی کاهش بازخوانی، تاثیر بسزایی بر معیار دقت دارد.

از دیگر تحقیقات انجام شده در این پروژه بررسی میزان تاثیر حجم پایگاه داده آموزش، برای استخراج کلمات کلیدی و استخراج کلمات برجسته ساز می‌باشد. آزمایشات نشان می دهد که افزایش حجم پایگاه داده در روش ارائه شده برای استخراج کلمات برجسته ساز منجر به نتایج مطلوب تر می‌شود در حالیکه افزایش حجم پایگاه داده برای روش استخراج کلمات کلیدی باعث آموزش بیش از حد و پیروی رفتار تابع امتیاز دهی از رفتار موتورهای جستجو و در نتیجه کاهش دقت و بازخوانی می‌شود. هرچند نحوه رفتار تابع امتیازدهی آنقدر پایدار نیست که بتوان با قطعیت رای بر صحت نتیجه بدست آمده داد.

همچنین در بخش‌های پیش‌پردازش که به زبان مورد تحقیق وابستگی وجود دارد، ایده‌های تازه در فرایندهای توکنیزه سازی و حذف کلمات پرتکرار ارائه شده است.

3-8 پیشنهاداتی برای آینده

موارد زیر به منظور ادامه و تکمیل این پروژه پیشنهاد می شود

8-3-1 ایجاد پایگاه داده بزرگتر برای اطمینان بخشی به نتایج بدست آمده

در بخش استخراج کلمات کلیدی اگرچه به نظر می رسد افزایش حجم پایگاه داده بیش از یک میزان بهینه، باعث ضعف در عملکرد تابع می شود. اما استفاده از پایگاه داده منجر به حرکت نرمتر و با ثبات‌تر

نمودار ملاک‌های کارایی روش برحسب تعداد اسناد (شکل 7-1) شده و اطمینان بیشتری از نتایج بدست آمده حاصل می شود.

همچنین در بخش استخراج کلمات برجسته ساز افزایش حجم پایگاه داده در طول فرایند آموزش باعث جامعیت لیست کلمات استخراج شده و بهبود نتایج خواهد شد.

همچنین پیشنهاد می شود این پایگاه داده ها استفاده از ساختارهای توزیع شده و تحت وب با استفاده از تعداد بیشتری از کاربران ایجاد شود. چرا که این امر منجر به جامعیت هرچه بیشتر پایگاه داده مذکور خواهد شد.

8-3-2 استفاده از یک روش ریشه یابی

اگرچه روش های ریشه یابی در زبان فارسی هنوز در ابتدای راه خود هستند اما استفاده از این یک روش ریشه یابی می تواند منجر به بهبودهای احتمالی در نتایج بدست آمده شود.

8-3-3 بررسی تاثیرات ایجاد یک مرز دینامیک امتیاز

از آنجا که در این پروژه مقدار مرزی امتیاز جهت انتخاب کلمات کلیدی بر اساس ملاک بازیابی صورت گرفت پیشنهاد می شود روشی برای تعیین این مرز بر اساس طول و نوع ساختار هر سند برای تعیین کلمات کلیدی طراحی شود. پیش بینی می شود این نوع انتخاب به افزایش میزان دقت و برعکس کاهش ملاک کارایی منجر شود که در حالت کلی می توان آن را به عنوان یک روش تاثیر گذار بر دقت بررسی کرد.

مراجع

[1] James Sanger and Ronen Feldman, "**The Text Mining Handbook advanced approaches in analyzing unstructured data**", Cambridge university press, 2007.

[2] J. Kaur and V. Gupta, "**Effective Approaches For Extraction of Keywords,**" Journal of Computer Science, vol. 7, 2010, pp. 144-148.

[3] C. Zhang, "**Automatic Keyword Extraction from Documents Using Conditional Random Fields**," Journal of Computational Information Systems, vol. 3, 2008.

[4] M.S. K. Taghva, R. Beckley, "**A Stemming Algorithm for the Persian Language**," International Conference on Information Technology: Coding and Computing (ITCC 2005), 2005.

[5] شمس فرد م, "**پردازش متون فارسی: دستاوردهای گذشته، چالش های پیش رو** "دومین کارگاه پژوهشی زبان فارسی و رایانه ,صفحه 172-189.، 1385

[6] K. Taghva, J. Coombs, R. Pareda, and T. Nartker, "**Language Model-Based Retrieval for Farsi Documents,**" Internationa Conference on Information Technology: Coding and Computing (ITCC□04), 2004.

[7] A.N. Taghiyareh F., Darrudi E., Oroumchian F., "**Compression of Persian Text for Web-Based Applications, Without Explicit Decompression**."

[8] J.W. Amtrup and N.M.S.U.C.R. Laboratory, **"Persian-English Machine Translation : An Overview of the Shiraz Project"**, Computing Research Laboratory, New Mexico State University, 2000.

[9] S.M. Assi, "**Farsi Linguistic Database (FLDB)**," international Journal of Lexicography, vol. 10, 1997, pp. 5-10.

[10] K.S. Esmaili, H. Abolhassani, and M. Neshati, "**Mahak : A Test Collection for Evaluation of Farsi Information Retrieval Systems,**" Cities, 2007, pp. 639-644.

[11] A. AleAhmad, H. Amiri, E. Darrudi, M. Rahgozar, and F. Oroumchian, "**Hamshahri: A standard Persian text collection**," Knowledge-Based Systems, vol. 22, 2009, p. 382–387.

[12] حافظی م م ، ثامتی ح ، منصوری ن ، منتظری ن ، بحرانی م ، موثق ح, **"یک مدل دستوری برای بهبود دقت سیستم‌های بازشناسی گفتار پیوسته‌ی فارسی** ",دومین کارگاه پژوهشی زبان فارسی و رایانه،صفحه 80-91 ، 1385

[13] D.J.& J. H.Martin, "**Speech and Language Processing: An introduction to speech recognition, natural language processing, and computational linguistics**," 2005, pp. 29-32.

[14] مدرس خیابانی ش ، قیومی م, **"نقش پیکره‌های زبانی در باهم‌آیی: رویکردی مقایسه‌ای ",** دومین کارگاه پژوهشی زبان فارسی و رایانه، صفحه 55- 65، تهران 1385

[15] داداش میری پ. **"تشخیص انتهای کلمات فارسی و ایجاد فاصله میان کلمات"** پایان نامه کارشناسی، دانشگاه علم و صنعت 1380.

[16] بیجن‌خان م, **"تشخیص کسره اضافه** ",طرح تحقیقاتی پژوهشگاه فرهنگ و هنر دانشگاه تهران 1384

[17] مرتضی آنالویی ، محسن مشکی، **"یک روش آماری مبتنی بر پیکره برای جداسازی واژه‌های به هم چسبیده** ",کنفرانس فازی و هوشمند.2008 ,

[18] P. Schäuble, **Multimedia information retrieval : content-based information retrieval from large text and audio databases**, Boston: MA:kluwer Academic publishers, 1997.

[19] K. Taghva, R. Beckley, and M. Sadeh, **A list of Farsi stopwords**, las vegas: Citeseer, 2003.

[20] M.R. Davarpanah, M. Sanji, and M. Aramideh, "**Farsi lexical analysis and stop word list,**" emerald group publishing, vol. 27, 2009, p. 435–449.

[21] C.S. alahati Qadimi Fumani, M. R.; Ramachandra, "**The concept of stop words in Persian chemistry articles: A discussion in automatic indexing**.," 2008.

[22] انوری،ح. احمد گیوی ,.**دستور زبان فارسی** 2 ,تهران: انتشارات فاطمی چاپ 21, 1380.

[23] **"تحلیل سیستم یافتن خودکار کلمات کلیدی متون زبان فارسی"**،دانشگاه علم و صنعت، ویرایش اول ، 1388

[24] M.F. Porter, "**an algorithm for suffix stripping**," 1983.

[25] B. Lovins, "**Development of a Stemming Algorithm**," Mechanical transaltion and computational linguistics, vol. 11, 1968, pp. 22-31.

[26] N.F. and M.M. Maristella Agosti, Michela Bacchin, "**Improving the Automatic Retrieval of Text Documents**," Advances in Cross-Language Information Retrieval, vol. 2785, 2003.

[27] M.M. Michela Bacchin, Nicola Ferro, "**A probabilistic model for stemmer generation**," Information Processing & Management, vol. 41, 2005, pp. 121-137.

[28] M.meybodi, "**Bon: First Persian Stemmer**," First Eurasia Conference on information2, 2003.

[29] B. Esfahbod, "**FarsiTEX and the Iranian TEX Community,**" East Asia, vol. 23, 2002, pp. 41-45.

[30] مجتبی محمدی نصیری، کیومرث شیخ اسماعیلی ،حسن ابوالحسنی ., **"یک ریشه یاب آماری برای زبان فارسی** ",مجموعه مقالات یازدهمین کنفرانس بین المللی کامپیوتر.1384 ,(csicc)

[31] J.D. Cohen, "**Highlights : Language- and Domain-Independent Indexing Terms for Abstracting Automatic,**" journal of the american society for information science, vol. 46, 1995, pp. 162-174.

[32] Y. Matsuo and M. Ishizuka, "**Keyword extraction from a single document using word co-occurrence statistical information,**" International Journal on Artificial Intelligence Tools, vol. 13, 2004, p. 157–170.

[33] L.F. Chien, "**PAT-tree-based keyword extraction for Chinese information retrieval**," ACM SIGIR Forum, ACM, 1997, p. 50–58.

[34] G. Ercan and I. Cicekli, "**Using lexical chains for keyword extraction,**" Information Processing & Management, vol. 43, Nov. 2007, p. 1705–1714.

[35] A. Hulth, "**Improved automatic keyword extraction given more linguistic knowledge,**" Proceedings of the 2003 conference on Empirical methods in natural language processing -, 2003, pp. 216-223.

[36] M.J.M.G. G. Salton, "**Automatic Text Structuring and Retrieval –Experiments in Automatic Encyclopaedia Searching,**" Fourteenth SIGIR Conference, 1991, pp. 21-30.

[37] "**Domain-Specific Keyphrase Extraction**," 16th International Joint Conference on Aritifcal Intelligence, Stockholm, Sweden: 1999, pp. 668-673.

[38] K. Zhang, H. Xu, J. Tang, and J. Li, "**Keyword extraction using support vector machine,**" Advances in Web-Age Information Management, 2006, p. 85–96.

[39] I.H. Witten, G.W. Paynter, E. Frank, C. Gutwin, and C.G. Nevill-Manning, "**KEA: Practical automatic keyphrase extraction,**" Proceedings of the fourth ACM conference on Digital libraries, ACM, 1999, p. 254–255.

[40] P.D. Turney, "**Learning algorithms for keyphrase extraction**," NRC Technical Report ERB-1057, 2000.

[41] J.B.K. Humphreys, "**Phraserate: An html keyphrase extractor,**" Riverside: University of California, Riverside, 2002, pp. 1-16.

[42] تشکری.م, "**ساخت یک نمایه ساز خودکار برای متون فارسی**," دانشگاه صنعتی امیرکبیر, 1380.

[43] El-Shishtawy, T., Al-Sammak, A. "**Arabic Keyphrase Extraction using Linguistic knowledge and Machine Learning Techniques**", Proceedings of the Second International Conference on Arabic Language Resources and Tools, 2009

[44] J. Wang, H. Peng, and J.S. Hu, "**Automatic keyphrases extraction from document using Neural Network,**" Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on, IEEE, 2005, p. 3770–3774.

[45] **Search Engine Optimization Starter Guide**, Google Inc, 2010.

[46] C.T.Y. s l G. Salton, C. S. Yang, "**A Theory of Term Importance in Automatic Text Analysis**," Journal of the American society for Information Science, vol. 26, 1975, pp. 33-44.

[47] D.C. sorensen jorge J.more, "**Computing a Trust Region Step**," SIAM Journal on Scientific and Statistical Computing, vol. 3, 1983, pp. 553-572.

[48] R. Mihalcea and P. Tarau, "**textRank: bringing order into texts**," Proceedings of EMNLP 2004, 2004, pp. 404-411.

Abstract

Generally, automatic keyword extraction methods attempt to get better results in criterions such as recall and precision, which show the ability of the method to perform the desired action as an author.

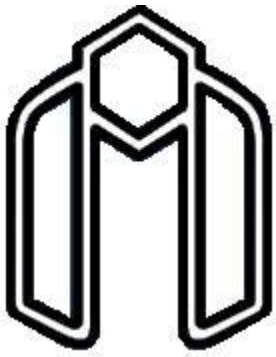
According to the undeniable role of search engines in today's world, it seems that keywords selection methods should pay the same attention to web content accessibility by search engines and the information retrieval criterions.

In this thesis, a new Automatic keyword extraction method is presented which acquires as good recall/precision as other methods and at the same time increases the contents accessibility by search engines. To extract keywords two score functions are used. The firsrt function is a statistical keyword selector and the second one is a search engine results evaluator. The keyword selector function attempts to get a better recall/precision, while the search engine result evaluator function optimizes the first one, using genetic algorithm and search engine results during training phase.

The introduced keyword extraction method also uses a new post-processing pahse which employs attention attraction strings to improve precision without losing noticeable recall.

Experimental results show that using proper documents in training phase, reasonable balance of the recall, precision and web content accessibility can be obtained.

Keywords: Automatic keywords extraction, Search engine optimization, SEO, Farsi stopwords, Genetic algorithm, Farsi attention attraction strings, Precision improvement, Accessibility improvement.



Shahrood University of Technology

Faculty of Computer and Information Technology

Optimized Keyword Selection for General Search Engines

Hamze Hodhodkian

Supervisor(s):

Dr.Morteza Zahedi

Dr.Hamid Hassanpur

July 2011