عنوان:

تشخیص هرزنامه وب به کمک تکنیک های داده کاوی

فهرست مطالب:

فهرست جداول:

فهرست اشکال:

چکیده:

امروزه هرزنامه[1]ها یکی از مشکلات اصلی موتورهای جستجو هستند، به این دلیل که کیفیت نتایج جستجو را نامطلوب می سازند. در طول سالهای اخیر پیشرفتهای بسیاری در تشخیص صفحات جعلی وجود داشته است اما در پاسخ تکنیک های هرزنامه جدید نیز پدیدار شده اند. لازم است برای پیشی گرفتن به این حملات، تکنیکهای ضد هرزنامه بهبود یابد.

یک مساله عادی که ما با آن در این زمینه مواجه می شویم این است که خیلی از اسناد رتبه بالایی را توسط موتور جستجو بدست آورده اند در حالی که سزاوار آن نبوده اند. با توجه به گسترش روزافزون وب و همچنین ظهور تکنیک های جدید هرزنامه توسط هرزنامه نویسان، هدف از این پایان نامه بررسی روش های مبتنی بر داده کاوی جهت شناسایی هرچه بهتر صفحات هرزنامه از غیرهرزنامه است.

الگوریتم ها و نرم افزارهای داده کاوی از جمله ابزارهای مورد استفاده در این پژوهش هستند. از مجموعه داده استاندارد UK2007 و نرم افزار وکا جهت ارائه مدلهایی بهینه استفاده شده است و سعی بر ارائه مدلهایی است که ضمن کاهش ویژگی های مورد استفاده جهت شناسایی صفحات هرزنامه از غیرهرزنامه کارایی مطلوبی را نیز ارائه دهد.

[1] Spam

فصل اول

مقدمه

۱-۱- پیش گفتار:

دادهکاوي که با عنوان کشف دانش از پايگاههاي داده هم شناخته ميشود، فرايندي براي استخراج الگوهاي مفيد از پايگاههاي داده ميباشد [۱]. دادهکاوي مي تواند الگوهاي مفيد مورد نظر کاربران خود را از انواع مختلفي از پايگاههاي داده استخراج کند. بيشتر محققان دادهکاوي را مترادف با کشف دانش در پايگاههاي داده ميدانند. کشف دانش شامل مراحل زير است که به صورت پي در پي انجام مي شوند:

- پالايش داده: آشفتگيها و دادة ناسازگار را حذف میکند.
- يکپارچهسازي داده: درصورت لزوم منابع دادهاي را ترکيب میکند.
- تبديل داده: داده را به فرم مناسب براي داده کاوي تبديل ميکند.
- داده کاوي: فرايندی ضروري است که در آن روشهاي هوشمند الگوهاي دادهاي مناسب را استخراج ميکنند.
- ارزيابي الگو: الگوهاي استخراج شده را ارزيابی میکند.
- نمايش دانش: در اين مرحله تکنيکهاي مختلف نمايش دانش براي نشان دادن دانش کشف و کاوش شده به کاربر استفاده ميشوند.

افزايش توانايي تکنيکها و ابزارهاي مختلف در ايجاد و جمعآوري دادهها و اهميتي که پايگاههاي داده به دليل در دسترس بودن و قوي بودنشان در صنايع و تحقيقات مختلف دارند، همچنين شبکة گستردة جهاني که به عنوان يک منبع اطلاعاتي مهم بشمار ميرود، ما را با حجم عظيمي از داده و پايگاههاي داده روبرو ساخته است.

اگرچه موتورهای جستجو تکنیک های زیادی را برای شناسایی هرزنامه وب گسترش داده اند اما هرزنامه نویسان وب تاکتیک های جدیدی را برای تاثیر گذاری روی نتایج الگوریتم های رده بندی موتورهای جستجوگر، به منظور دستیابی به رده های بالاتر توسعه داده اند.

داده کاوی به عنوان ابزاری مهم و نو کاربرد گسترده ای در شناسایی صفحات هرزنامه از غیرهرزنامه دارد.

۱-۲- بیان مسئله:

موتورهای جستجو به مکانی برای جستجوی اطلاعات بر روی وب تبدیل شده اند. با توجه به پدیده هرزنامه، نتایج جستجو همواره مطلوب نیست.

بیش از دو دهه است که پژوهش بر روی بازیابی اطلاعات خصمانه در دانشگاه و صنعت علاقه مندان زیادی دارد. هرزنامه ها بر هر سیستم اطلاعاتی، ایمیل، وب و وبلاگ ها و شبکه های اجتماعی سایه افکنده اند. این مفهوم برای اولین بار در سال ۱۹۹۶ مطرح شد و به زودی به عنوان یک چالش برای موتورهای جستجو مطرح شد.
اخیراً همه شرکت های بزرگ موتور جستجو به دلیل اثرات متعدد ومنفی ناشی از ظهور هرزنامه ها، بازیابی اطلاعات خصمانه را به عنوان یک اولویت بالا تعیین کرده اند [۲,۳]. نخست آنکه هرزنامه ها کیفیت نتایج جستجو را نامطلوب میسازند و بازده ای که سایت های قانونی می توانند در غیاب هرزنامه ها داشته باشند را کاهش می دهند.

دوم آنکه باعث عدم اطمینان یک کاربر به موتور جستجو شده و نهایتاً منجر به تعویض موتور جستجو که برای کاربر هزینه ای در بر نخواهد داشت می گردد.

هدف تعیین ویژگی های متفاوت صفحات وب به منظور رتبه بندی نتایج موتور جستجو است و بر این اساس کلاس بندی به منظور شناسایی سایتهای هرزنامه از سایتهای معتبر انجام می پذیرد.

۳-۱- اهمیت و ضرورت انجام تحقیق:

هرزنامه ها به عنوان ابزاری برای انتشار محتوای مربوط به بزرگسالان و بدافزار ها و حملات مطرح می شوند. به عنوان مثال، رتبه بندی ۱۰۰ میلیون صفحه براساس الگوریتم های رتبه بندی صفحه نشان داد که ۱۱ نتیجه از ۲۰ نتیجه، سایت های پرونوگرافی بوده اند که با دستکاری محتوا و پیوند ها به این نتیجه رسیده اند [۴,۵]. در گذشته این امر باعث می شد مقدار قابل توجهی منابع محاسباتی و ذخیره سازی از شرکتهای موتورهای جستجو، به هدر رود. در سال ۲۰۰۵ ضرر و زیان ناشی از هرزنامه ها ۵۰ میلیارد دلار تخمین زده شد. در سال ۲۰۰۹ نیز ۱۳۰ میلیارد دلارتخمین زده شد [۶]. از جمله چالش های جدید، رشد سریع وب و ناهمگونی آن و ساده سازی ابزارهای ایجاد محتوا (به عنوان مثال ویکی وب سایت، سکوهای بلاگ نویسی و ...) و کاهش هزینه نگهداری وب سایت (نظیر ثبت دامنه، میزبانی وب و...) می باشد که باعث تحول هرزنامه ها و ظهور سویه های جدید هرزنامه وب که نمی تواند با روش های موفق قبلی شناسایی شود، شده است.

کسری از ارجاعات به صفحات وب که از موتورهای جستجو می آیند قابل توجه هستند و کاربران تمایل به بررسی نتایج با رتبه بالا دارند . برای ۸۵ درصد از پرسش ها تنها نخستین صفحه نتیجه مورد توجه واقع شده است و تنها سه پیوند کلیک شده است [۷]. بنابراین تلاش برای گنجانده شدن در نخستین صفحه نتیجه موتور جستجو با توجه به افزایش ترافیک وب سایت ها انگیزه روشن اقتصادی خواهد داشت. به منظور نائل شدن به این هدف، صاحبان وب سایت ها، برای دستکاری نتایج رتبه بندی موتورهای جستجو تلاش می کنند. مطابق با مطالعات انجام شده مقدار هرزنامه ها از ۶ تا ۲۲ درصد متغییر است و این امر نشان دهنده حوزه و دامنه مشکل است [۸,۹].

ساختار پایان نامه:

با توجه به موضوع پایان نامه، در ابتدا در فصل دوم به بررسی ساختار وب و مفاهیم هرزنامه و انواع هرزنامه و برخی از مهمترین روش های یادگیری ماشین پرداخته شده است. در فصل سوم به معرفی مجموعه داده های موجود پرداخته و تکنیک های مقابله با هرزنامه های لینک و هرزنامه های محتوا و هرزنامه های لینک-محتوا مورد بررسی قرار گرفته است. در فصل چهارم به معرفی و مجموعه داده انتخابی پرداخته شده و نتایج مربوط به مدلهای بهینه داده کاوی بیان گردیده است. در فصل پنجم نیز به عنوان فصل پایانی، نتیجه نهایی کار جمع بندی شده و مسائلی که می توانند به عنوان موضوع پایان نامه کارشناسی ارشد در آینده مورد توجه و بررسی قرار گیرند، بیان گردیده است.

فصل دوم :
وب و هرزنامه های وب

با توجه به موضوع پایان نامه در ابتدا بررسی ساختار وب و انواع هرزنامه و همچنین مهمترین الگوریتم های یادگیری ماشین ضروری به نظر می رسد. بنابراین در این فصل در ابتدا مفاهیم وب و سپس انواع هرزنامه و در پایان الگوریتم های یادگیری ماشین مورد بررسی قرار گرفته است.

۱-۲- وب جهان گستر:

وب جهان گستر را می توان به عنوان یک پایگاه داده مدیریت شده توسط بشریت برای ذخیره سازی و اشتراک اسناد مختلف در نظر گرفت، با این حال، وب با پایگاه داده های معمول در اندازه، پویایی بسیار سریع و ناهمگونی تفاوت دارد. نخست آنکه وب بسیار بزرگ است، نمی توان اندازه آن را به طور دقیق مشخص کرد و اندازه گیری نمود. تعداد صفحات نامحدود است و محتوا نیز به اطلاعات وارد شده توسط کاربر بستگی دارد. جولی و سیگنورینی‌گزارش کرده اند که وب ۱۱ میلیارد صفحه را در ژانویه ۲۰۰۵ شامل می شده است [۱۰]. در ۲۰۰۸ گوگل ادعا کرد که سیستم آنها یک تریلیون URL را روی وب پردازش کرده است [۱۱].

یک چالش دیگر این است که محتویات وب به سرعت تغییر می کند. چو و گارسیا-مولینا[2] سرعت تغییر را با دانلود ۷۲۰۰۰۰ صفحه در یک دوره چهار ماهه سال ۱۹۹۹ ارزیابی کرده اند [۱۲]. آنها به این نتیجه رسیدند که محتویات صفحه برای ۲۳ درصد مجموعه اصلاح شد و در طی ۵۰ روز، ۵۰ درصد مجموعه ویرایش یا برداشته شدند [۱۳].

اسناد وب به دلایل مختلف و دیدگاه های مختلف، ناهمگون هستند. علاوه بر متن، وب سایت ها می تواند شامل تصاویر، فیلم ها و فایل های صوتی در فرمتهای مختلف باشد. اندازه این اسناد می تواند از یک بایت تا هزاران مگابایت متفاوت باشد. در میان متداولترین فایل های HTML، می توانید نسخه های متفاوت و صفحات با نحو ناصحیح را پیدا کنید که از استانداردهای W3C پیروی نمی کنند اما هنوز هم توسط مرورگرهای[4]وب قابل مشاهده هستند. محتوای وب اغلب بدون ساختار است، از زبانها و سبک های مختلف تشکیل شده است و کیفیت آن در طیف گسترده ای متفاوت است. اگرچه صفحات HTML، تعدادی فراداده[5]را در بر می گیرند اما در حالت کلی قابل اعتماد نیستند. هدف اصلی وب معنایی، قرار دادن داده ها به صورت ساختار یافته ی دوستدار ماشین[6]است که همکاری میان انسان و ماشین را ممکن می سازد [۱۴].

۱-۱-۲-وب به عنوان یک گراف:

وب در مقایسه با اسناد متداول بی نظم تر به نظر می رسد، با این حال پیوندها در سراسر اسناد، وب را یک منبع غنی و مفید از داده ها می سازند. فوق پیوندها، یک شبکه بزرگ را تعریف می کنند که اسناد وب را بهم متصل می کند.

ساختار پیوند روی وب می تواند به عنوان یگ گراف مستقیم G=(V,E) بیان شود که V مجموعه صفحات است و E مجموعه پیوند هاست، $(u,v)\in E$ است اگر صفحه u به صفحه v پیوند داشته باشد. در این گراف $d^-(u)$ درجه ورودی U را نشان می دهد برای مثال تعداد صفحاتی که به u لینک می

[2] Gulli and Signorini
[3] Cho and Garcia-Molina
[4] browser
[5] Meta Data
[6] machine friendly

شوند و $d^+(u)$ درجه خروجی، تعداد صفحاتی که به وسیله u مورد اشاره قرار می گیرند. گراف وب نیز مانند دیگر شبکه های ایجاد شده توسط بشر دارای خواص جالبی است [۲۴].

۲-۱-۲- گراف وب در صفحه و سطح میزبان:

لینک ها مابین صفحات همیشه همانند نیستند. برخلاف لینک های مابین سایت های مختلف که تفسیرهای بشری را حمل می کنند، لینک های هادی (هدایت کننده)، بین صفحات داخلی وب سایت ممکن است نتوانند مقدار اضافی را برای صفحه مورد اشاره، اظهار کنند. دیویسون نشان داد که چگونه می توان لینکهای داخلی وب سایت را برای اهداف هدایت کننده تعیین نمود. این لینک ها در صورت لزوم می توانند از گراف وب حذف شوند [۱۵].

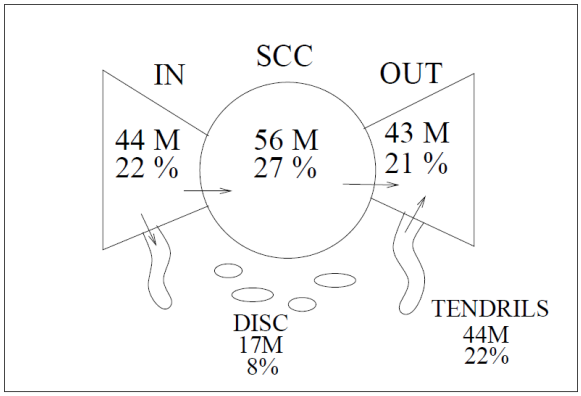
یک راه ساده تر برای تشخیص لینک های تفسیری از لینک های هادی، چشم پوشی از هر لینک داخل وب سایت و ایجاد یک گراف کوچکتر که گره ها، سایتها هستند و یالها فقط لینک های ما بین سایتها هستند، است.

در این مورد نمی توان صفحات همان وب سایت را تشخیص داد و بنابراین الگوریتم رتبه بندی، نمرات را به گره های گراف تخصیص می دهد، همه صفحات آن سایت، نمرات همانند بدست می آورند [۲۴].

۳-۱-۲- اتصال:

اگر فوق پیوندهای وب به عنوان لبه های بدون جهت تلقی شوند، به جز چند صفحه مجزا، گراف وب یک جزء[۷]متصل تکی را شکل می دهد. اگر لبه ها جهت دار تلقی شوند، گراف وب یک ساختار متفاوت را شکل می دهد .تعداد زیادی از گره ها از بقیه گراف قابل دسترس نیستند ، بر این اساس گره های گراف می توانند به ۵ بخش تقسیم بندی شوند. این ساختار توسط بُرُدِر[۸]و همکاران کشف شد. آنها ۲۰۳ مگابایت صفحه وب را آنالیز کرده اند و آن را اغلب به ساختار بُو-تای[۹]ارجاع داده اند. ساختار زیر در صفحات وب تعریف شده که در طرح زیر نمایش داده شده است [۱۶]. قسمت اصلی هسته مرکزی است (SCC)، بزرگترین جزء متصل شده به گراف است. در SCC، هر گره در تعداد گام های کم از سایر گره ها در SCC قابل دسترس است. اندازه SCC، در آزمایش بُرُدِر ۵۰ مگابایت است، در حالی که دومین بزرگترین جزء متصل شده تنها ۱۵۰ کیلوبایت صفحه را در بر می گیرد.

[7] Component
[8] Broder
[9] bow-tie



شکل۲-۱: ساختار بوی-تای وب شرح داده شده در [۱۶]

گره های خارج از SCC که از SCC قابل دسترسی هستند. هر دو جزء OUT و IN، ۲۲ درصد از کل گراف را تشکیل می دهند. گره های باقیمانده می توانند از IN مورد دستیابی قرار گیرند و یا به OUT برسند و یا نمی توانند به هیچ کدام از قسمتهای بالا دستیابی داشته باشند (DISK) [۱۶].

۲-۲- موتورهای جستجو:

در روزهای اولیه پیدایش وب، کاربران جهت مرور صفحات وب به دو مکانیسم اصلی متکی بودند. در هر صورت آنها URL صفحه مورد نظر را در نوار آدرس مرورگر تایپ می کردند و یا آنها روی فوق پیوندها در صفحات وب کلیک می کردند. با افزایش تعداد صفحات وب دو روش دیگر نمود پیدا کرد : لغت نامه های وب و موتورهای جستجو وب.

دایرکتوری های وب، تعداد زیادی فوق پیوند را که به صفحات وب مفید اشاره می کنند، در بر می گیرد. این لینک ها معمولاً در یک طبقه بندی سلسله مراتبی سازماندهی شده اند و معمولاً توسط بشر به آنها پرداخته شده است. نمونه شناخته شده، دایرکتوری یاهو در dir.yahoo.com و پروژه دایرکتوری باز dmoz.org است. متاسفانه ایجاد و نگهداری دایرکتوری های وب، نیاز به تلاش وسیع انسانی دارد. از آنجایی که سایتهای وب به سرعت در حال تغییر هستند، نگهداری و بروزرسانی دایرکتوری های وب بسیار مشکل است. از طرف دیگر تلاش غیر قابل اغماضی از سمت کاربران در جهت انتخاب رده بندی مناسب در سلسله مراتب چند سطحی وجود دارد [۱۳].

موتورهای جستجو تقریبا خودکار[1] هستند و نیاز به تلاش کمتر بشر برای بروز نگه داشتن پایگاه داده دارند و می توانند از پس طبیعت رو به رشد وب برآیند. آنها سرویسی را ارائه می دهند که صفحات را بر اساس کلمات کلیدی ارائه شده توسط کاربر ارائه می دهد. معمولاً کاربر کلمات کلیدی یا متنهای کوتاه را در کادر جستجو وارد نموده و روی دکمه "جستجو" کلیک می نماید و موتور جستجو تعدادی لینک ها را که امیدوار است مطابق با علاقه ی کاربر باشد، فهرست میکند.

[1] Automatic

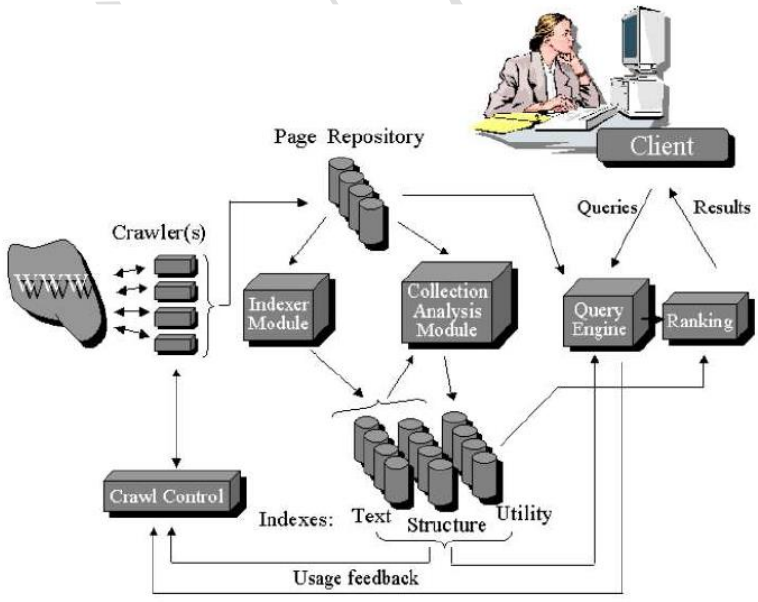
موتورهای جستجو توانایی برگرداندن نتایج متمرکزتری را نسبت به دایرکتوری های وب دارند. هر چند که برای پرس و جو های مبهم و بدفرم، نتایج بدست آمده اغلب بی فایده و گمراه کننده می باشد. چالش اصلی موتورهای جستجو درک پرسش و بازگرداندن بهترین نتایج است. وب معمولاً میلیونها صفحه را که با کلمات کلیدی کاربر مطابقت دارد در بر می گیرد، اما کاربران نیاز به بازدید تعدادی کمی از سایتهای وب مرتبط دارند. آنالیز لاگ فایل های پرس و جوی وب نشان داد که کاربران ده نتیجه نخستین فهرست شده توسط موتور جستجو را بازدید می کنند [۱۷].

تمرکز اصلی رتبه بندی، بازیابی اطلاعات[1] می باشد. یک زمینه تحقیقاتی چند رشته ای برای انتخاب و رتبه بندی اسنادی که با کلمات کلیدی تطابق دارد. وب در مقایسه با متنهای کلاسیک متفاوت است. صفحات با فوق پیوندها به هم لینک شده اند که می تواند در محاسبات رابطه مورد سوء استفاده قرار بگیرد [۲۴].

۱-۲-۲- معماری موتورهای جستجوی وب:

اگرچه اطلاعات در مورد اجزاء و الگوریتم های موتورهای جستجوی تجاری در دسترس عموم نیست، اعتقاد بر این است که ساختار کلی شبیه شکل زیر است [۱۸, ۱۹, ۲۰]. معماری را می توان به سه بخش منطقی مجزا از هم تقسیم نمود.

۱- کاوشگر[2] صفحات وب را در یک مخزن محلی دانلود می کند.
۲- شاخص ساز[3] صفحات دانلود شده را پردازش می کند.
۳- موتور پرس و جو، برای رسیدگی به پرس و جو های کاربران با استفاده از داده های پیش پردازش تولید شده توسط شاخص سازی، به کار می رود.



شکل ۲-۲: معماری کلی موتور جستجو [۱۸]

[1] Information Retrieval
[2] Crawler
[3] Indexer

کاوشگر:
کاوشگر، گراف وب را به وسیله فوق پیوندها می پیماید و صفحات کشف شده در این فرآیند را دانلود می کند. صفحات دانلود شده در یک مخزن محلی نگهداری می شود.

چالش اصلی در کاوشگرها این موارد هستند :
پوشش:
کاوشگر صفحات را تا بیشترین حد ممکن دانلود میکند اما اجازه سربار سایت های وب را نمی دهد، هم چنین کاوشگر از قوانینی پیروی می کند که اجازه درخواست مکرر را نمی دهد.
تازگی :
صفحات منسوخ شده باید دوباره واکشی شوند. انواع مختلف صفحات ممکن است طول عمر متفاوت داشته باشند. صفحات تغییر کرده غالباً باید دوباره واکشی شوند اما صفحاتی که به ندرت تغییر می کنند ندرتاً بروزرسانی می شوند. وظیفه کاوشگر به طور همزمان حفظ تازگی و پوشش است.

شاخص سازی:
به منظور پاسخ به جستجو در کسری از ثانیه، پیش محاسبه حجیمی نیاز است. در طول پیش محاسبه پایگاه داده های متفاوت، شاخص ها ساخته می شوند.
برای هر کلمه w، شاخص اصلی، فهرست اعلان تمام شناسه های سند و مکان های وقوع آنها را در بر می گیرد. فهرست اعلان برای همه کلمات پرسش به منظور تولید فهرستی از اسناد که با پرسش مطابقت داشته باشد، پردازش می شود. این شاخص اغلب به عنوان شاخص معکوس مورد ارجاع قرار می گیرد بدلیل اینکه نمود ترانهاده ای از رابطه سند – کلمه را بیان می کند.
ساختن شاخص معمولاً به شکل موازی انجام می شود که شامل تجزیه و تحلیل و نشانه گذاری اسناد در مخزن می باشد.
علاوه بر شاخص اصلی، پایگاه داده ها و شاخص هایی دیگر ساخته می شود که اغلب توسط کاوشگر و یا برای رتبه بندی استفاده می شود. به عنوان مثال گراف وب توسط تجزیه و تحلیل فوق پیوندها ساخته می شود. معیارهای کیفیت مستقل پرس و جو (به عنوان مثال رتبه) می تواند برای هر سند در شاخص های سودمند اضافی محاسبه و ذخیره شود.
۲-۲-۲-سرویس دهنده[4]پرس و جوی موتور جستجو:
هدف اصلی موتور پرس و جو، پردازش پرس و جو های ارسال شده توسط کاربران است. اگر پرس و جو شامل بیش از یک عبارت باشد معمولاً موتورهای جستجو یک AND بین آنها به صورت پیش فرض در نظر می گیرند، یعنی کاربر صفحه ای که شامل تمام کلمات مورد جستجو است را دریافت خواهد کرد. به منظور تولید این فهرست اسناد، موتور پرس و جو، فهرست اعلان کلمات پرس و جو را در شاخص مورد توجه قرار داده، تناسب آنها را محاسبه نموده، رتبه بندی را به کار برده و موفقیت[5]ها را به کاربر ارائه می نماید. ارائه نتایج به کاربر ممکن است شامل محاسبات دیگر و دستیابی به پایگاه داده های دیگر باشد، نظیر تولید snippet، گزیده ای کوتاه از اسناد که کلمات پرس و جو را در بر می گیرد. همه این فرآیندها باید در کسری از ثانیه انجام بگیرد و این زمان پاسخ نیاز به الگوریتم های کارا دارد و پیش محاسبات در طی شاخص سازی را توجیه می کند [۲۴].

[1] Server
[1] hit
[4]
[5]

۳-۲- رتبه بندی

۱-۳-۲-رتبه بندی مبتنی بر محتوا:

رتبه بندی مبتنی بر محتوا یک مولفه موتور جستجو می باشد که تمرکز آن بیشتر روی بازیابی اطلاعات کلاسیک است که یک رشته چند شاخه ای می باشد که به بررسی مدلهایی برای انتخاب و رتبه بندی اسناد که با کلمات کلیدی داده شده مطابقت دارد، می پردازد. موتورهای جستجو نمره رابطه را به هر سند اختصاص می دهند که ارتباط سند با پرسش را بیان می کند. با سند و پرسش به عنوان ترتیبی از کلمات رفتار می شود. پرسشی که برای الگوریتم رتبه بندی متناسب شده، در برخی مواقع با پرسشی که کاربر تایپ کرده متفاوت می باشد. برخی از واژه ها ممکن است حذف شوند و برخی کلمات ممکن است به پرسش اضافه شوند. برای نمونه کلمات پر تکرار (the,a,and,….) اغلب از پرسش ها حذف می شوند و گاهی اوقات ممکن است به خوبی تمییز داده نشوند و باعث نویز در رتبه بندی می شوند. از سوی دیگر، کلمات مرتبط نظیر مترادف ها یا صرف فعل ممکن است به منظور پیدا کردن اسنادی که شکل های دیگری از کلمات پرسش را در بر می گیرند، اضافه شوند. دو مدل هست که اغلب در موتورهای جستجو و بازیابی اطلاعات مورد استفاده قرار می گیرد. مدل فضای برداری که اسناد و پرسش ها را با بردار بیان می کند و به مدل ارتباط احتمالی که با شاخه ی احتمالات مطابقت دارد [۲۳,۲۲].

مدل فضای برداری و TF-IDF:

در مدل فضای برداری، هر سند به وسیله ی یک بردار خلوت بیان می شود که هر ورودی متناسب با یک کلمه است. ورودی های متناسب با کلمات در سند غیرصفر هستند. به طور مشابه، پرسش ها نیز با یک بردار خلوت نشان داده می شوند و رابطه، به وسیله تشابه مابین بردارهای پرسش و سند بیان می شوند. راه های متفاوتی برای تعریف مقدار یک کلمه در بردار سند برای بیان اهمیت متناسب با کلمه وجود دارد. بهترین و شناخته ترین طرح، TF-IDF خوانده می شود [۲۴].

فرض کنید پرسش، اصطلاح واحد t را در بر می گیرد. چگونه می توانیم رابطه سند d برای این پرسش را تعیین کنیم؟ ایده اولیه همان تعداد دفعات وقوع t است. این ایده زمانی موفق است که همه اسناد اندازه مشابه داشته باشند، اگر یکی کوتاه و دیگری بلند باشد و هر دو شامل تعداد دفعات مشابه باشند، اینگونه احساس می شود آنکه کوتاهتر است نسبت به آنکه بلندتر است ارتباط بیشتری با پرسش دارد، به این دلیل که سند بلندتر محتویات غیرمرتبط اضافی را دربر می گیرد.

به همین دلیل است که ما فرکانس اصطلاح را تعریف می کنیم. اگر N_{ij} تعداد دفعات اصطلاح t که در سند d آشکار می شود، باشد، در اینصورت فرکانس اصطلاح t در d [۲۴]:

$$TF_{td}=\frac{Nt,d}{\sum_{i\in d} Ni,d} \qquad (۲-۱)$$

فرض کنید پرسش ما "the big Apple" باشد و ما از مجموع سه فرکانس اصطلاح به عنوان نمره رابطه استفاده می کنیم.واضح است که سه اصطلاح در پرسش اهمیت مساوی ندارند. سندی که به تعداد دفعات زیاد کلمات "the" و "big" را در بر می گیرد اما یک بار کلمه ی "Apple" را شامل می شود،نسبت به سندی که دربرگیرنده کلمه "Apple" به تعداد دفعات بیشتر است، ارتباط کمتری با پرسش دارد.

فرکانس سند اصطلاح t را به عنوان کسری از اسناد که حداقل یک بار t را در بر می گیرد، تعریف می کنیم. فرکانس سند معکوس، لگاریتم معکوس فرکانس سند که به عنوان وزن برای اصطلاحات استفاده می شود، است [۲۴]: .

$$IDF_t=\log\frac{|D|}{|\{d\in D:t\in d\}} \qquad (۲-۲)$$

نهایتاً نمره TF-IDF سند d برای پرسش q را با ترکیب فرکانس اصطلاح و فرکانس سند معکوس به دست می آوریم [۲۴]: .

$$TF\text{-}IDF_{d,q}=\sum_{t\in d,t\in q} TFt,d.IDFt \qquad (۲-۳)$$

قسمتهای مختلف اهمیت متفاوتی دارند، بنابراین مکان وقوع اصطلاح پرسش ممکن است (عنوان، URL، نام فایل، کلمات کلیدی) باشد و یا نوع (اندازه، فونت، حروف برجسته) متفاوت باشد. و منطقی است که به اصطلاحاتی که در عنوان آشکار می شوند نسبت به اصطلاحاتی که در بدنه آشکار می شوند، نمره بالاتری تعلق گیرد، یا می توان وزن بالاتری برای کلمات نوشته شده با اندازه فونت بزرگتر یا کلماتی که در شروع سند آشکار می شوند، به کار برد [۲۵].

۲-۳-۲- الگوریتم های مبتنی بر لینک:

رتبه صفحه:[1]

رتبه صفحه به وسیله برین و پیج[2] اختراع شد و شاید بهترین الگوریتم مبتنی بر لینک است [۲۶].الگوریتم رتبه بندی، نمره صفحه را مبتنی بر وجود یک لبه (u,v) محاسبه می کند. یک پیاده سازی به منظور رتبه بندی صفحات، استفاده از $d^-(v)$ درجه ورودی است، تعداد صفحاتی که به v مرتبط می شوند.

این روش اشکالاتی نیز دارد:

- با همه پیوند ها به صورت مساوی رفتار می شود، بنابراین یک شخص می تواند تعداد زیادی از صفحات ساختگی ایجاد کند که به صفحه ی هدف مرتبط می شوند و سبب افزایش درجه ورودی می شوند. به عبارت دیگر درجه ورودی، به آسانی هرزنامه پذیر است.
- پیوند از یک صفحه محبوب نظیرcnn.com باید ارزش بیشتری را نسبت به پیوندهای متفاوت از صفحات با کیفیت کم داشته باشد.
- صفحات با درجه خروجی بیشتر تاثیر بالاتری روی رتبه صفحات دیگر دارند. بنابراین سهم پیوند از صفحه u باید متناسب با $1/d^+(u)$ باشد.

این مسائل منجر به تعریف بازگشتی برای کیفیت صفحه شده است.

$$P(v)=\sum_{u:(u,v)\in E} p(u)/d^+(u) \qquad (۲-٤)$$

[1] Page Rank
[1] Brin and Page
[6]
[7]

به عنوان مثال یک صفحه اگر به وسیله ی صفحات با کیفیت بالاتر مورد اشاره قرار بگیرد، دارای کیفیت بالاتری است.
یک گره w با درجه خروجی صفر (که اغلب "گره آویزان[۱۳]" خوانده می شود) به عنوان چاله[۱]عمل می کند. هر رتبه تخصیص داده شده به w از دست می رود، زیرا که w هرگز در سمت راست معادله بالا ظاهر نمی شود. با خلاصه سازی معادله ی بالا برای همه u ها، ما مشاهده می کنیم که باید برای همه گره های آویزان باید داشته باشیم0=p(u).
در نتیجه تمام گره هایی که از یک مسیر مستقیم در G به یک گره با درجه خروجی صفر می رسند باید رتبه ی صفر داشته باشند. بنابراین معادله ی بالا به شکل زیر اصلاح می شود، به طوری که همه گره های آویزان پیوندهایی به همه گره های موجود دارند و این به آن معنی است که همه ی گره ها $\frac{1}{n}$ رتبه را از هر گره آویزان دریافت می کنند که n در اینجا تعداد گره ها است.
اگر B نشان دهنده ماتریس مجاورت گراف وب با ردیف های نرمال سازی شده با استفاده از اصلاحات گره های آویزان که در بالا شرح داده شد باشد [۲۴]:

$$B(u,v)=\begin{cases}\frac{1}{d^+(u)} & \text{اگر صفحه u به صفحه V}\\ 0 \quad =d^+(u) & \text{اگر}\\ \frac{1}{n} & \text{در بقیه}\\ 0 & \text{موارد}\end{cases}$$

پس معادله بالا به شکل زیر در می آید:

$$P=B^TP$$

ماتریس انتقال B و زنجیره مارکوف مطابق با آن نامتناوب است اگر و فقط اگر بزرگترین مقسوم علیه مشترک طول همه ی چرخه های مستقیم در گراف وب یکی باشد. B قابل کاهش نیست اگر و فقط اگر G شامل یک کامپوننت متصل تنها (SCC) باشد، یعنی یک مسیر مستقیم مابین هر زوج صفحات در G باشد و این در عمل موردنظر نیست و این را می توان با محدود کردن محاسبات به بزرگترین جزء متصل گراف وب بدست آورد اما بزرگترین SCC گراف وب واقعی تنها ۳۰%-۲۵ گره ها را در بر می گیرد [۱۶]. به منظور غلبه بر این مسائل، ما یک گراف کامل وزن دار کوچک را به گراف وب خود افزوده ایم. ما بردار "رتبه صفحه" P با P(i)≥0 و $\|P\|_1=1$ تعریف کرده ایم.

$$P(v)= c.r(v) + (1-c).\sum_{u:(u,v)\in E} p(u)/d^+(u) \qquad (۲-۵)$$

در اینجا $r=(r(1),\ldots..,r(n))^T$ بردار شخصی با r(i)≥0 و $\|r\|_1=1$ و c احتمال انتقال با مقدار c≈۰/۱۵.
اگر r یکنواخت است، به عنوان مثال $r(v)=\frac{1}{n}$ برای همه v ها، سپس P رتبه صفحه (PR) می باشد.
در نشانگذاری ماتریس [۲۴]:

$$M(u,v) = cr(v)+(1-c)\begin{cases}\frac{1}{d^+(u)} & \text{اگر صفحه u به صفحه V اشاره کند}\\ 0 \quad =d^+(u) & \text{اگر}\\ \frac{1}{n} & \end{cases}$$

[1] dangling
[1] Sink

0

در بقیه

پس داریم:

$$P=cr+(1-c)B^TP=M^TP$$

این ماتریس نیز نامتناوب است، زیرا گره ها با r(v)>0 طول یک چرخه با احتمال انتقال مثبت M(vv)>0 دارند. اگر برای همه ی v ها در r(v)>0،G باشد، پس همه ی گره ها از همه گره ها قابل دسترس هستند، بنابراین همه ی گره ها یک SCC واحد را شکل می دهند، پس M غیرمتناوب است.
ایده " رتبه صفحه" به خوبی از طریق مدل بازدید کننده ی تصادفی[2] قابل توضیح است. در این مدل کاربر از طریق صفحات وب هدایت می شود. او می تواند دو نوع عملیات را اجرا کند. در هر صورت او یکی از فوق پیوندهای صفحه ی جاری را دنبال می کند و یا مستقیماً به یک صفحه ی تصادفی می رود. او نخستین یا دومین عمل را با احتمال c - ۱ انتخاب می کند. اگر او تصمیم بگیرد یکی از لینک ها را دنبال کند، این لینک به صورت تصادفی انتخاب شده است. به عبارت دیگر صفحه ی هدف v با احتمال r(v) انتخاب شده است.
دومین نوع عملیات " انتقال از راه دور" نامیده می شود که c احتمال انتقال از راه دور است و r بردار شخصی یا انتقال:
اگر P(k) به توزیع بازدید کننده تصادفی در زمان k اشاره داشته باشد با P(0)=r پس [۲۷]:

$$P^{(k)}=M^TP^{(k-1)}=(M^T)^kr$$

به وسیله ی تئوری پایه ای زنجیره ی مارکوف، k^{th} امین تکرار $P^{(k)}$ همیشه به راه حل یکتای معادله ی بالا همگرا خواهد بود و کسری از زمان که رندم سورفر روی صفحه v صرف می کند، به رتبه صفحه P(v) همگرا خواهد بود [۲۷].

- **رتبه صفحه شخصی:**

فرض کنید که n صفحه روی وب موجود است و هر صفحه A ، لینک های ورودی T1، T2 و .....Tn دارد و C(A) تعداد لینک هایی است که از صفحه A خارج می شوند و d یک مقدار پیوسته در محدود ۰ و ۱ باشد. پس رتبه صفحه، PR(A) به صورت زیر بیان می شود [۲۸]:

$$PR(A)=\frac{(1-d)}{n}+d\sum_{i=1}^{n}\frac{PR(Ti)}{C(Ti)} \qquad (۲-٦)$$

برای ماتریس کامل M معادله بالا بعد از چند تکرار همگرا خواهد شد. بعد از محاسبه ی رتبه صفحه، در صورتی که این مقدار بالا باشد، صفحه در مکان بالا رتبه بندی خواهد شد.

- **HITS:**

کلینبرگ[1] یک الگوریتم مبتنی بر تکرار را تحت عنوان HITS یا الگوریتم کلینبرگ معرفی نموده است. HITS مبتنی بر پرس و جو است که محاسبات آن باید خیلی سریع در زمان پرس و جو انجام شود. محاسبات HITS بر پایه زیرگراف کوچکی است که به روش زیر تولید شده است:

[2] Random Surfer 0
[2] Kleinberg 1

- یک مجموعه S_q از نتایج جستجو که تصور می شود با پرسش q متناسب است، بدست آورده می شود و این می تواند n نتیجه بالای موتورهای جستجو با استفاده از رتبه بندی مبتنی بر متن باشد. معمولاً صفحاتی که از S_q مورد اشاره قرار گرفته اند و صفحاتی که به S_q اشاره می کنند نیز اضافه می شوند.
- ساخت یک گراف مجاورت G_q، زیرگراف به وسیله مجموعه گسترش یافته S_q توسعه می یابد.

HITS دو نمره برای هر گره در G_q محاسبه می کند. نمره اعتبار a(v) که منعکس کننده ارتباط محتوای v با پرسش است. نمره قطبیت[۲] h(v) که کیفیت صفحه v را به عنوان یک مجموعه لینک با توجه به موضوع داده شده، تعیین می کند.

الگوریتم بر این اصل استوار است که قطب های خوب به احتمال زیاد به صفحات با اعتبار بالا لینک می شوند و اعتبارهای خوب در صفحات با قطبیت خوب آشکار می شوند.

در یک الگوریتم با تکرار، فرض کنید که h(k) و a(k) به بردارهای نمره قطبیت و نمره اعتبار اشاره می کنند. نخستین بردار قطبیت، h(0) ممکن است مقادیر مثبت دلخواه را شامل شود. (k+1) امین نمره اعتبار v از بردار قطبیت قبلی h(k) محاسبه می شود به طوری که مجموع نمرات قطبیت جاری گره هایی که به v اشاره می کنند [۲۹]:

$$a^{(k+1)}(v)=\sum_{(u,v)\in Eq} h^{k}(u) \qquad (۲-۷)$$

به طور مشابه k+1 امین نمره قطبیت v، مجموع نمره اعتبار گره هایی است که به وسیله v مورد اشاره قرار گرفته اند.

$$h(k+1)(u)=\sum_{(u,v)\in Eq} a^{(k+1)}(v)$$

اگر A_q گراف مجاورت G_q هست، معادلات بالا می تواند به شکل ماتریس بیان شوند [۲۹]:

$$a^{(k+1)}=A_q^T\, h(k)$$
$$h^{(k+1)}=A_q\, a(k+1) \qquad (۲-۸)$$

از این رو [۲۹]:

$$A(k+1)= A_q^T\, A_q\, a(k)$$
$$h^{(k+1)}=A_q\, A_q^T\, A_q\, h(k)$$

h(k) و a(k) به h و a همگرا می شوند. در عمل چند صد تکرار برای این همگرایی لازم است.

۲-۴- هرزنامه وب :

جستجوی در وب در عصر اطلاعات بسیار مهم شده است. افزایش صفحات وب می تواند باعث دستاوردهای مالی و یا شهرت برای سازمانها شود. رتبه در وب شاید مهم ترین شاخص در مواجهه با صفحات وب باشد. اگر کاربری از اطلاعاتی که مربوط به صفحه وب خود است را جستجو نماید، اما صفحه اش توسط موتورهای جستجو دارای رتبه پایینی باشد، کاربر ممکن است صفحه خود را نبیند. این امر توسط سازمانها، اشخاص و حتی افراد پذیرفته شده نیست. به همین دلیل، درک الگوریتم های

[2] hub [2]

رتبه بندی و ارائه اطلاعاتی در صفحات فردی زمانی که اصطلاحات متناسب با محتویات آنها مورد جستجو واقع می شود که صفحات رتبه بالایی داشته باشند، امری مهم است.
متاسفانه این امر منجر به هرزنامه نگاری شده است که به فعالیتهای بشری به منظور گمراه کردن موتورهای جستجو برای رتبه بندی صفحات بالاتر از مکانی که سزاوار آن هستند، اشاره می کند.
همه صفحات مطابق با مقادیر اطلاعاتی شان رتبه بندی شده اند. هرزنامه نگاری عملی است که به مقادیر اطلاعات صفحه چیزی اضافه نمی کند اما مکان آن را در رتبه بندی با گمراه کردن الگوریتم های متورهای جستجو افزایش می دهد.
الگوریتم های موتورهای جستجو محتویات اطلاعات صفحه را درک نمی کنند، آنها از ویژگی های نحوی و یا ظاهری برای ارزیابی ارزش اطلاعات روی صفحه استفاده می کنند. هرزنامه نگاران از این ضعف به منظور افزایش رتبه صفحات خود استفاده می کنند.
اسپم برای کاربران آزاردهنده است زیرا باعث می شود آنها سخت تر به اطلاعات مفید دسترسی پیدا کنند و تجربه جستجویی خسته کننده را داشته باشند. اسپم هم چنین برای موتورهای جستجو نیز مناسب نیست زیرا باعث مصرف پهنای باند کاوشگر، آلوده شدن وب و تحریف رتبه بندی جستجو می شود.
در واقع شرکتهای های زیادی هستند که به بهبود رتبه بندی کمک می کنند. این شرکت ها ، شرکت های بهینه سازی موتورهای جستجو نامیده می شوند (SEO[2]). الگوریتم های جستجو فاکتورهای مبتنی بر محتوا و مبتنی بر اعتبار را در نمره هر صفحه در نظر می گیرند، در اینجا تعدادی از روش های هرزنامه نگاری را که از این عوامل بهره برداری می کنند را شرح می دهیم [۲۱].

۱-۴-۲- هرزنامه محتوا:

اکثر موتورهای جستجو از تغییر پذیری TF-IDF برای ارزیابی ارتباط یک صفحه با پرسش کاربر استفاده می کنند. روشهای هرزه نگاری مبتنی بر محتوا، محتویات فیلدهای متنی در صفحات HTML را با تعدادی از پرسش ها متناسب می کنند. از آنجا که TF-IDF بر اساس اصطلاحات محاسبه می شود هرزنامه نگاری محتوا، هرزنامه نگاری عبارات نیز خوانده می شود.
دو تکنیک اصلی برای هرزنامه عبارت موجود است که به سادگی محتوای غیرواقعی برای هرزنامه را ایجاد می کند.

۱- تکرار تعدادی عبارات مهم: این روش TFعبارات تکرارشده در سند را افزایش می دهد و به این ترتیب ارتباط این سند را با این عبارات افزایش می دهد. از آنجا که تکرار ساده به راحتی قابل تشخیص می باشد ، هرزنامه عبارت می تواند با تعدادی جملات ساخته شود که ممکن است از منابع دیگر کپی شده باشد و عبارات هرزنامه به صورت تصادفی در این جملات قرار گرفته اند. برای نمونه اگر یک هرزنامه نویس، نیاز به تکرار کلمه "mining" داشته باشد، به جای تکرار چند باره متوالی آن که به آسانی قابل تشخیص است، می تواند جمله نامربوط "the picture mining quality of this camera mining is amazing" را به کار ببرد [۳۱].
۲- انبار کردن اصطلاحات نامرتبط زیاد: این روش برای مرتبط کردن صفحه با تعداد زیادی از پرس و جو ها استفاده می شود. به منظور ایجاد سریع محتوای هرزنامه، هرزنامه نویس می تواند به سادگی عبارات را از صفحات مرتبط روی وب کپی کرده و آنها را در کنار هم قرار دهد.

آگهی ها هم می توانند تعدادی عبارت مورد جستجو را مورد سوء استفاده قرار دهند و آنها را در صفحات هدف قرار دهند، به طوری که وقتی کاربر عبارت مورد نظر را جستجو می کند، صفحات هدف به آن مربوط می شوند. برای مثال برای تبلیغ بسته های تعطیلات کروز، هرزنامه نویسان اصطلاح " تام کروز " را در صفحات تبلیغ خود قرار می دهند، زیرا تام کروز بازیگر مشهور آمریکا می باشد که نام او به دفعات زیاد مورد جستجو واقع می شود.

[2] Search Engine Optimization [3]

هرزنامه عبارات می تواند در هر فیلد متنی قرار داده شود:
عنوان: از آنجایی که موتورهای جستجو معمولاً وزن بالایی را به عبارات عنوان اختصاص می دهند، با توجه به اهمیت عنوان در صفحه، هرزنامه عنوان رایج و متداول است.
متابرچسب ها[۲۴]
متابرچسب های HTML در سر صفحه[۲۵]مالک صفحه را قادر می سازد بعضی از اطلاعات نظیر نویسنده، چکیده، کلمات کلیدی و زبان محتوا را قرار دهد. متابرچسب ها به مقدار زیاد در هرزنامه نگاری استفاده می شوند [۳۱].

* متابرچسب های توصیفی (description):
این تکنیک مشابه برچسب عنوان می باشد. متابرچسب های توصیفی به طراح صفحه اجازه می دهند تا توصیف کوتاهی راجع به صفحه داشته باشد. اگر کلمات نامرتبط در این جا قرار داده شوند، الگوریتم های موتور جستجو که شاخص سازی را بر این اساس انجام می دهند، صفحات با این کلمات نامرتبط را هدف قرار می دهند.
* متابرچسب های کلمات کلیدی : متابرچسب های کلمات کلیدی، برای نشان دادن کلمات کلیدی صفحه هستند. تعدادی از موتورهای جستجو ممکن است وزن بالایی به کلمات موجود در اینجا (کلمات کلیدی) اختصاص دهند، بنابراین هرزنامه نویسان می توانند از کلمات نامرتبط در این برچسب سوء استفاده کنند [۳۳].
بدنه : اصطلاحات هرزنامه می تواند در بدنه به منظور افزایش رتبه قرار داده شود [۳۱].
متن لنگر[۲۶]
صفحات وب دارای ویژگی های خاص در بازیابی اطلاعات وب هستند: فوق پیوندها روی صفحه وب با چند کلمه متن لنگر همراه هستند. این متن های کوتاه، معمولاً اطلاعات شخصی درباره ی صفحه ی مورد اشاره را در بر می گیرد. موتورهای جستجوی وب، متن فوق پیوندها را علاوه بر صفحه ی در برگیرنده، به عنوان محتوای صفحه ی هدف شاخص سازی می کنند. همچنین وزن در صفحه هدف به طور کلی بالاتر است [۲۴].
آنها در شاخص سازی صفحاتی که آنها را در بر می گیرند و صفحاتی که به آنها اشاره می کنند تاثیر گذار هستند. بنابراین هرزنامه نگاری روی متن لنگر روی رتبه بندی هر دو نوع صفحات تاثیر گذار خواهد بود.
برای هرزنامه نویسی متن لنگر، هرزنامه نویسان نمی توانند صفحه هدف را تغییر دهند، در عوض صفحات دیگری با لینک هایی به صفحه هدف ایجاد می کنند و اصطلاحات هرزنامه را به متن های لنگر این صفحه اضافه می کنند.

URL:
برخی از موتورهای جستجو URL صفحات را به عبارات می شکنند و آنها را در رتبه بندی در نظر می گیرند. بنابراین اسپم می تواند دربرگیرنده عبارات در URL شود.

۲-۴-۲-هرزنامه لینک:
از آنجا که لینک ها نقش مهمی را در تعیین نمره یک صفحه بازی می کنند، هرزنامه نویسان روی فوق پیوندها هرزنامه نگاری می کنند. در واقع هرزنامه لینک، دستکاری ساختار لینک یا متن لنگر به منظور دستیابی به رتبه بالاتر است [۳۱].

---

2 Meta tag
2 Header
2 Anchor text
4
5
6

۱-۲-۴-۲- هرزنامه لینک های خروجی:

اضافه کردن لینک های خروجی به صفحات شخصی که به صفحات معتبر[۱۷]اشاره می کنند، آسان است. یک صفحه هاب، صفحه ای است که به تعداد زیادی صفحه معتبر اشاره کند. برای ایجاد لینک های خروجی به شکل وسیع، هرزنامه نویسان می توانند از تکنیکی به نام شبیه سازی دایرکتوری استفاده کنند. دایرکتوری های زیادی موجود است مانند یاهو، دایرکتوری باز DMOZ که حاوی تعداد زیادی لینک به سایر صفحات هستند که با توجه به برخی سلسله مراتب موضوعی مشخص، سازماندهی شده اند. هرزنامه نویسان، به سادگی بخش بزرگ دایرکتوری را برای ایجاد لینک های خروجی به صورت سریع، در صفحات هرزنامه خود تکرار می کنند [۳۱].

۲-۲-۴-۲- هرزنامه لینک ورودی:

هرزنامه نویسی لینک ورودی سخت تر است، زیرا که اضافه کردن فوق پیوندها روی صفحات وب دیگران آسان نیست. هرزنامه نویسان یک یا تعدادی از تکنیک های زیر را استفاده می کنند [۳۱].

*مزرعه لینک[۱۸]

هرزنامه نویسان الگوریتم های مبتنی بر لینک نظیر "رتبه صفحه" و "HITS" را هدف قرار می دهند. هرزنامه نویسان می توانند با هزینه کم تعداد زیادی صفحه که به صفحه هدف متصل می شوند را ایجاد کنند. صفحات این مزرعه لینک باید از طریق کاوشگر موتور جستجو قابل دسترس باشد و این امر با اضافه کردن لینک ها روی صفحه ای که در حال حاضر در دسترس است، امکانپذیر می باشد. موتورهای جستجو از الگوریتم های مبتنی بر لینک استفاده می کنند اما این الگوریتم ها می توانند توسط ساختارهای لینک بزرگ دستکاری شوند.

برای ساخت ساختارهای لینک پیچیده به صفحات وب به عنوان گره های پشتیبان[۱۹]نیاز دارند. هرزنامه نویسان از دو نوع صفحه استفاده می کنند:

- صفحات شخصی که تحت کنترل کامل هرزنامه نویسان هستند. این صفحات توسط هرزنامه نویسان ایجاد شده اند. محتوا، URL و ساختار لینک این صفحات توسط هرزنامه نویسان ایجاد شده است.
- صفحات دسترس پذیر که مالک آنها، هرزنامه نویسان نیستند اما هر شخصی اجازه افزودن محتوای خود را دارد. این صفحات می تواند انجمن ها[۳۰] و پست های بلاگ ها باشد. هرزنامه نویسان می توانند لینک هایی را به عنوان "نظر" برای انتشار اعتبار صفحات دسترس پذیر به صفحات هدف اضافه کنند. فرض کنید که هرزنامه نویسان می خواهند الگوریتم "رتبه صفحه" را مورد حمله قرار دهند و نیاز به افزایش مقدار "رتبه صفحه" صفحه t در مزرعه لینک G دارند. برای آنالیز این سناریو، مقدار رتبه صفحه کلی G هست [۳۰]:

$$PR(G)=PR_{static}(G) + PR_{in}(G)-PR_{out}(G)-PR_{sink}(G)$$

---

[2] Authoritative
[2] Link Farm
[2] Supporter
[3] forum
[7]
[8]
[9]
[0]

که $PR_{static}(G)$ مقدار رتبه صفحه جمع آوری شده به وسیله رَندم سورفر می باشد. $PR_{in}(G)$ به وسیله صفحات دیگر که به G لینک می شوند، دریافت می شود. $PR_{out}(G)$ به وسیله صفحات دیگر (لینک های خارجی) ارسال می شود. PR_{sink} (G) رتبه صفحه در صفحات بدون لینک خروجی است.

ما یک مزرعه لینک ساده را شرح می دهیم که فرمول بالا را بیشینه می کند و هم چنین رتبه صفحه t ϵ G را بیشینه می کند.

- صفحات شخصی به منظور افزایش PR_{sink} به G اضافه می شوند.
- همه صفحات دسترس پذیر به منظور افزایش PR_{in} به G می پیوندند .
- هیچ لینک خروجی به منظور کمینه کردن PR_{out} وجود ندارد.
- همه صفحات در G به منظور بیشینه شدنPR_{sink} به صفحات دیگر لینک می شوند .
- همه لینک ها از صفحات شخصی و دسترس پذیر به صورت مستقیم به t لینک می شوند تا PR_{in} بیشینه شود.
- برای جلوگیری از، از دست رفتن رتبه و برای اینکه همه صفحات در G دسترس پذیر باشند ، t به همه صفحات شخصی در G می پیوندد [۳۲,۳۰].


Bedfordshire swingers Berkshire swingers Buckinghamshire swingers Cambridgeshire swingers Cheshire swingers Cleveland swingers Cornwall swingers County Durham swingers Cumbria swingers Derbyshire swingers Devon swingers Dorset swingers East Sussex swingers East Yorkshire swingers Essex swingers Gloucestershire swingers Greater Manchester swingers Hampshire swingers Herefordshire and Worcestershire swingers Hertfordshire swingers Humberside swingers Isle of Man swingers Isle of Wight swingers Kent swingers Lancashire swingers Leicestershire swingers Lincolnshire swingers London swingers Merseyside swingers Norfolk swingers North Yorkshire swingers Northamptonshire swingers Northumberland swingers Nottinghamshire swingers Oxfordshire swingers Shropshire swingers Somerset swingers South Yorkshire swingers Staffordshire swingers Suffolk swingers Surrey swingers Tyne and Wear swingers Warwickshire swingers West Midlands swingers West Sussex swingers West Yorkshire swingers Wiltshire swingers Swingers and swinging in aberdeenshire Swingers and swinging in alderney Swingers and swinging in anglesey Swingers and swinging in angus Swingers and swinging in argyllshire Swingers and swinging in avon Swingers and swinging in ayrshire Swingers and swinging in banffshire Swingers and swinging in berwickshire Swingers and swinging in borders Swingers and swinging in brecknockshire Swingers and swinging in central Swingers and swinging in clwyd Swingers and swinging in county antrim Swingers and swinging in county armagh Swingers and swinging in county down Swingers and swinging in county fermanagh Swingers and swinging in county leitrim Swingers and swinging in county londonderry Swingers and swinging in county tyrone Swingers and swinging in dumfries and galloway Swingers and swinging in dyfed Swingers and swinging in fife Swingers and swinging in grampian Swingers and swinging in guernsey Swingers and swinging in gwent Swingers and swinging in gwynedd Swingers and swinging in hampshire Swingers and swinging in herefordshire Swingers and swinging in herm island Swingers and swinging in hertfordshire Swingers and swinging in highlands and islands Swingers and swinging in humberside Swingers and swinging in isle of man Swingers and swinging in isle of wight Swingers and swinging in isles of scilly Swingers and swinging in jersey Swingers and swinging in kent Swingers and swinging in lancashire Swingers and swinging in leicestershire Swingers and swinging in lincolnshire Swingers and swinging in london Swingers and swinging in lothian Swingers and swinging in merseyside Swingers and swinging in mid glamorgan Swingers and swinging in northamptonshire Swingers and swinging in northumberland Swingers and swinging in nottinghamshire Swingers and swinging in oxfordshire Swingers and swinging in powys Swingers and swinging in rutland Swingers and swinging in sark

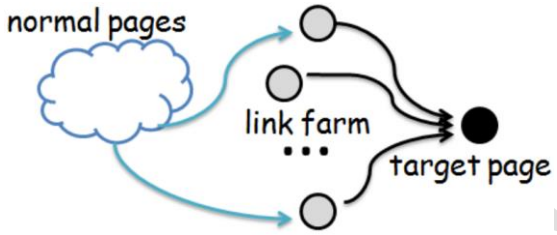
شکل ۲-۳: صفحه مزرعه لینک [۳۰]

۱- ایجاد **honey pot**:

اگر یک صفحه نیاز به نمره و اعتبار بالا دارد، نیازمند این است که صفحات با کیفیت بالا به آن اشاره کنند. این روش در واقع تلاشی برای ایجاد صفحات مهم است که حاوی لینک هایی به صفحات هرزنامه هدف هستند.

برای نمونه، هرزنامه نویسان می توانند مجموعه ای از صفحات که اطلاعات مفیدی را در بر می گیرند، ایجاد کنند، برای نمونه واژه نامه اصطلاحات داده کاوی ، صفحات راهنما.

Honey pot ها افراد را به خود جذب می کنند، زیرا که آنها اطلاعات مفید را در بر می گیرند و نمره بالایی دارند (صفحات با کیفیت بالا). honey pot ها، لینک های مخفی را به صفحات هرزنامه هدف دارند که هرزنامه نویسان نیاز به ارتقا این صفحات دارند. این استراتژی می تواند باعث تقویت صفحات هرزنامه شود. یک مزرعه لینک معمولی پیکربندی دارد که در طرح زیر نشان داده شده است [۳۱].



شکل ۲-۴: honeypot [۳۱]

۲-اضافه کردن لینک ها به دایرکتوری های وب:
تعداد زیادی از دایرکتوری ها به کاربران اجازه قرار دادن URL های خود را می دهند، هرزنامه نویسان، می توانند URL های صفحات خود را در دایرکتوری های چند گانه قرار دهند. از آنجایی که دایرکتوری ها کیفیت و نمره بالایی دارند، می توانند نمره اعتبار صفحات هرزنامه را به طور قابل توجهی افزایش دهند.
۳-پیوندهایی به محتویات تولید شده توسط کاربر(بحث های انجمن ها ، بلاگ ها و ....):
سایتهای زیادی روی وب هستند که به کابران اجازه میدهد آزادانه پیام های خود را ارسال کنند، که به آنها محتویات تولیدشده توسط کاربر می گویند. هرزنامه نویسان می توانند لینک هایی را به پیام هایی که ارسال می کنند اضافه کنند که به صفحات خودشان اشاره می کند.

۴-مشارکت در تبادل لینک:

در این مورد خیلی از هرزنامه نویسان گروهی را تشکیل می دهند و یک طرح تبادل لینک را راه اندازی می کنند به طوری که به منظور ارتقای رتبه صفحات، سایت هایشان به همدیگر اشاره می کنند.

۵- ایجاد spam farm:

در این مورد هرزنامه نویسان نیاز به کنترل تعداد زیادی وب سایت دارند ، سپس هر ساختار لینکی می تواند برای افزایش رتبه صفحات هرزنامه ایجاد شود.

۶-دامنه های منقضی شده:

هرزنامه نویسان دامنه های منقضی شده را خریداری نموده و محتوای بی فایده خود را روی آنها قرار می دهند. از آنجا که بعضی از این دامنه ها دارای پیشینه خوبی هستند، هنوز هم در رتبه بندی جایگاه خوبی را دارند.

۷-هرزنامه بمباران لینک:

تعدادی از موتورهای جستجو از روی متن لنگرهای لینک هایی که به یک صفحه اشاره می کنند، روی آن صفحه قضاوت می کنند. به دلیل بمباران لینک، کلمات پرس و جو نامرتبط که در متن های لنگر آشکار می شوند، صفحه هدف دارای رتبه بالاتری خواهد بود. به بمباران لینک اغلب بمب باران گوگل نیز گفته می شود.

۸-هرزنامه بلاگ یا نظرات :

هرزنامه نویسان لینک هایی را به بلاگ ها یا سیستم های ویکی اضافه می کنند. از آنجایی که بعضی از بلاگ ها یا ویکی ها از اعتبار خوبی برخوردارند، هرزنامه نویسان می توانند از تولید این هرزنامه های نظرات، سود ببرند [۳۳].

۹-هرزنامه لینک های وابسته :

فروشگاه های آنلاین مشهور برنامه های وابسته ای را تهیه دیده اند، یعنی اگر یک سایت ارجاعی به یک تراکنش را برگرداند، صاحب وب سایت می تواند مقداری پول از این ارجاع بدست آورد. هرزنامه نویسان شروع به ساختن صفحاتی با هدف سود بردن از این برنامه های وابسته کرده اند. به عنوان مثال، هرزنامه نویسان می توانند همه صفحات از ebay.com یا amazon.com را کپی کنند و لینک های وابسته را روی این صفحات اضافه کنند. این صفحات تنها صفحات کپی شده هستند، بدون هیچ اطلاعات اضافه ای [۳۳].

۳-۴-۲- تکنیک های مخفی:

در بیشتر موارد هرزنامه نویسان می خواهند عبارات، اصطلاحات و یا لینک را مخفی می کنند به طوری که از دید کاربران پنهان بماند. آنها می توانند از تعدادی تکنیک ها استفاده کنند.

پنهان کردن مطالب و محتوا :

آیتم های هرزنامه نامرئی هستند. یک روش ساده آن است که اصطلاحات هرزنامه به همان رنگ زمینه باشند.

یکی از مواردی که برای پنهان کردن قابل استفاده است در زیر آماده است.

```
<body background = white>
<font color = white> spam items</font>
…
        </body>
```

برای پنهان کردن یک فوق پیوند شما می توانید از یک تصویر خیلی کوچک و یک تصویر خالی استفاده کنید. برای نمونه :

```
<a href = target.html”><img src=”blank.gif”> </a>
```

یک هرزنامه نویس همچنین می تواند از اسکریپت ها برای مخفی کردن برخی عناصر بصری بر روی صفحه استفاده کند، برای مثال با تنظیم ویژگی مرئی Style HTML (visible) به مقدار flase [۳۰].

پنهان سازی **(cloaking)**:

هرزنامه نویسان ممکن است به دنبال یافتن این موضوع باشند که صفحات آنها توسط کاربر دانلود شده یا کاوشگر وب. اگر سرویس دهنده که صفحات هرزنامه را میزبانی می کند بداند که درخواست از یک کاوشگر می آید می تواند صفحه ای متفاوت از آنچه که برای مرورگر در نظر گرفته بفرستد.صفحه برای کاربران می تواند یک صفحه معتبر و با محتوای مفید، بدون هیچ نشانه ای از هرزنامه باشد، این صفحه قرار نیست توسط موتور جستجو دیده شود. به عبارت دیگر صفحه برای کاوشگر نیاز نیست که شامل هیچ محتوای مفیدی باشد.

سرویس دهنده های هرزنامه می توانند کاوشگر های وب را به دو روش زیر شناسایی کنند:

۱- آنها یک فهرست از آدرس IP های موتورهای جستجو را نگهداری می کنند و کاوشگر های وب را با مطابقت دادن آدرس IP شناسایی می کنند.

۲- آنها مرورگرهای وب را براساس فیلد User-agent در درخواست HTTP شناسایی می کنند. برای مثال نام User-agent در درخواست HTTP زیر استفاده شده توسط IE6 مایکروسافت است [۳۱].

```
GET /pub/WWW/TheProject.html HTTP/1.1
Host: www.w3.org
User–Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
```

تغییر مسیر:

هرزنامه نویسان می توانند یک صفحه هرزنامه را به وسیله تغییر مسیر خودکار مرورگر به یک URL که به زودی بارگذاری می شود، پنهان کنند. بنابراین صفحه هرزنامه به موتور جستجو برای شاخص سازی داده می شود و صفحه هدف به کاربر از طریق تغییر مسیر نشان داده می شود. یک راه برای تغییر مسیر استفاده از برچسب های متا و تنظیم زمان refresh به صفر است، راه دیگر استفاده از اسکریپت ها است [۳۱].

هرزنامه مبتنی بر لایه:

لایه های CSS مفاهیم جدیدی در وب ۲ هستند. لایه ها می توانند به عنوان بخش[1]های صفحه HTML دیده شوند و CSS می تواند برای نمایش لایه های متفاوت در یک صفحه در نظر گرفته شود. هرزنامه نویسان این لایه را به Invisible تغییر می دهند، در حالیکه محتویات این لایه هرزنامه است. کاربرعادی نمی تواند محتویات هرزنامه را ببیند اما ربات های موتورهای جستجو می توانند مشاهده کنند [۳۳].

سایر انواع هرزنامه ها :

هرزنامه های کپی:

هرزنامه نویسان محتویات وب سایت های شناخته شده را به صفحه خود کپی می کنند.برای مثال هرزنامه نویسان محتویات را از dmoz یا ویکی پدیا کپی می کنند و سپس لینک هایی را به سایت های اسپم در این صفحات قرار می دهند یا فقط تبلیغات و آگهی قرار می دهند [۳۳].

هرزنامه پرس و جو:

هرزنامه نویسان به ربات های پرس و جو اجازه جستجو می دهند لاگ فایل های پرس و جوی خود را آلوده کنند [۳۳].

[3] Segment [1]

هرزنامه نوار ابزار:
هرزنامه نویسان به موتورهای جستجو اجازه جستجوی داده های نوار ابزار باگ را میدهند که منجر به آلوده شدن لاگ فایل های مربوطه می شود [۳۳].

۵-۲- یادگیری ماشین:

شاخه ای برای طراحی و توسعه الگوریتم هایی که یاد می گیرند. در اینجا "یادگیری" به این معناست که برای یک وظیفه، الگوریتم می تواند کارایی را در طی زمان به وسیله داده های بیشتر بهبود دهد. یادگیری ماشین می تواند جستجو برای مدلهایی که تقریب خوبی از یک تابع ناشناخته که سیستم در حال پرسش را توصیف می کنند، باشد.
فرض بر این است که یک تابع y=f(x) داریم که ورودی x را به خروجی y نگاشت می کند. ما f واقعی را نمی دانیم اما ما مشاهداتی روی ورودی، گاهی روی خروجی و داده های آموزشی داریم. یادگیری فرآیند جستجو برای یک f' است که منطبق بر مشاهدات است.تابع f' اغلب به عنوان مدل است و مجموعه ای از توابع که تابع از آن انتخاب شده به عنوان فضای فرضیه در نظر گرفته می شود.
مجموعه داده استفاده شده برای یادگیری، داده های آموزشی[۲]است. بعد از اینکه یک مدل از داده های آموزشی به وسیله الگوریتم های یادگیری ساخته می شود، به وسیله داده تست[۳]داده های مشاهده نشده) برای دستیابی به دقت مدل ارزیابی می شود. دقت یک مدل طبقه بندی روی مجموعه تست به صورت زیر تعریف می شود:

دقت= تعداد نمونه های درست طبقه بندی شده / کل موارد تست

روشهای یاگیری ماشین می تواند مبتنی بر ماهیت داده های آموزشی و شیوه ی استفاده، به عنوان مختلف تقسیم شود. در یادگیری نظارت شده یک مجموعه با پاسخ درست داده می شود. برای هر ورودی x در داده های آموزشی، ما خروجی صحیح y=f(x) را می دانیم. وظیفه، پیش بینی پاسخ برای ورودی های بعدی است. مدل، بعد از پردازش داده های آموزشی، نهایی می شود و اصلاحی در طی پیش بینی برای داده های دیده نشده صورت نمی گیرد.
در یادگیری بدون نظارت، پاسخ صحیح داده نمی شود. وظیفه، شرح داده ها در تعدادی کلاس یا پارامتر است. الگوریتم کشف می کند که چگونه داده ها سازماندهی شده اند و کلاس ها را شناسایی می کند. شکل شناخته شده یادگیری بدون نظارت، خوشه بندی است که داده ها از گروههایی (خوشه) تشکیل شده اند. داده های در یک گروه باید مشابه باشند و در گروه های مختلف مبتنی بر یک اندازه داده شده، با هم متفاوت باشند. وظیفه خوشه بندی، تعیین خوشه مناسب برای هر نقطه داده است.
یادگیری نیمه نظارت شده که ترکیبی از یادگیری با ناظر و بدون ناظر است. یک مجموعه داده می شود اما پاسخ تنها برای کسری از آنها داده می شود. این روش معمولاً یک قاعده ویژه در داده های بدون برچسب را شناسایی می کنند و داده های برچسب دار را بر روی آن استفاده می کنند.
نتایج نشان داده است که استفاده از داده های بدون برچسب و تست، یادگیری را بهبود می دهد. با استفاده از یادگیری نیمه نظارت شده، هزینه زیاد داده های برچسب دار می تواند کاهش یابد [۲۴].

---

[2] Training Set
[3] Test Set

مختصراً تعدادی از تکنیکهای یادگیری ماشین را شرح می دهیم:

۱-۵-۲ -Naïve Bayes:

طبقه بندی کننده Naïve Bayes یک رویکرد مبتنی بر استنتاج بیز است. یک نمونه(x_k^1 =x_k ,…….x_k^m) و یک کلاس H_i داده شده است. روش Naïve Bayes، احتمال تعلق x_k را به H_i محاسبه می کند، به طوری که [۳۴]:

$$P(H_i|x_k^1,\ldots\ldots,x_k^m)=\frac{P(x_k^1,\ldots\ldots,x_k^m|Hi)P(Hi)}{P(x_k^1,\ldots\ldots\ldots,x_k^m)}$$

حال برای هر x_i، ما احتیاج به تعیین کلاس H_i با بالاترین احتمال داریم :

برای همه i:j که $i\neq j$ $P(H_i|x_k)>P(H_L|x_k)$

از آنجایی که مخرج در معادله بالا ثابت است، داریم:

$$P(H_i|x_k^1,\ldots\ldots,x_k^m)=Z.\ P(H_i|x_k^1,\ldots\ldots,x_k^m)P(H_i)$$

که Z یک مقدار ثابت مستقل از H_i است.

قاعده بیز، فرض می کند که تمام ویژگی های $x_k^1 x_k^m$,……, مستقل هستند، بنابراین توزیع شرطی می تواند به این صورت محاسبه می شود:

$$P(H_i|x_k^1,\ldots\ldots,x_k^m)=Z.P(H_i)\prod_{j=1}P(x_k^j|Hi)$$

یک برچسب کلاس $H^*=H_i$ به هر نمونه کلاس x_k با یک قانون تصمیم که بیشترین احتمال را بر می دارد، اختصاص می یابد.

قاعده بیز، یک روش کاملاً ساده و سریع است. قاعده بیز برای طبقه بندی کردن اسناد متنی با استفاده از کلمات به عنوان ویژگی ها به خوبی عمل می کند [۳۴].

۲-۵-۲- درخت تصمیم:

درخت تصمیم یک طبقه بندی کننده است که در شکل یک درخت دودویی ارائه شده که هر گره آن مربوط به یک متغییر است و به ما احتمال تحقق آن متغییر را نشان می دهد. برای یک نمونه داده (x_k^1 …….,x_k^m) =x_k، گره های برگ مربوط به احتمال کلاس H هستند.

هدف اصلی درخت تصمیم، ساخت فرضیه های کلاس مبتنی بر ویژگی های مشاهده شده از داده های آموزشی است. خروجی درخت تصمیم می تواند برای تعیین برچسب کلاسِ یک نمونه کلاس بندی نشده با در نظر گرفتن تحقق ویژگی های توصیفی آن استفاده شود [۲۴].

یادگیری درخت اغلب به کمک استراتژی تقسیم و غلبه انجام می گیرد که تقسیمات داده، درخت را به صورت بازگشتی تولید می کنند. در ابتدا همه نمونه ها در ریشه هستند، همانطور که درخت رشد می کند نمونه ها نیز به صورت بازگشتی تقسیم می شوند.

در یادگیری درخت، هر بازگشتی، بهترین خصیصه را برای قسمت بندی داده در گره جاری مطابق با مقدار خصیصه انتخاب می کند.

بهترین خصیصه براساس یک تابع انتخاب می شود که ناخالصی را بعد از تقسیم حداقل می کند، پس نکته کلیدی در درخت تصمیم گیری انتخاب تابع ناخالصی است [۳۵].

تابع ناخالصی که اغلب در یادگیری درخت تصمیم مورد استفاده قرار می گیرد information gain می باشد که در C4.5 مورد استفاده می باشد.

Information gain مبتنی بر تابع انتروپی می باشد [۳۵].

$$\text{entropy(D)}=-\sum_{j=1}^{|c|} pr(c_j)\log_2 pr(c_j) \quad (۲-۹)$$
$$)=1\sum_{j=1}^{|c|} pr(c_j$$

$pr(c_j)$، احتمال کلاس c_j در مجموعه داده D است. واحد انتروپی بیت است.روند به این صورت است که وقتی داده ها خالص تر و خالص تر می شوند، مقدار انتروپی کوچک و کوچکتر می شود. در واقع روشن است که اندازه انتروپی، مقدار ناخالصی در داده را نشان میدهد. همان چیزی که ما در یادگیری درخت تصمیم احتیاج داریم.

Information gain:

۱- در مجموعه داده D، نخست از تابع انتروپی برای محاسبه مقدار ناخالصی D استفاده می کنیم، همان entropy (D).

۲- سپس نیاز داریم که بدانیم کدام خصیصه میزان ناخالصی را کاهش می دهد. برای پیدا کردن آن، هر خصیصه مورد ارزیابی قرار می گیرد.
انتروپی بعد از تقسیم

$$\text{entropy}_{Ai}(D)=\sum_{j=1}^{v}\frac{|Dj|}{|D|}\times \text{entropy}(D_j)$$

اگر ما قصد داریم از خصیصه A_i استفاده کنیم، D به زیر مجموعه های D_1، D_2، ..... D_v تقسیم می شود.

۳- Information gain از فرمول زیر محاسبه می شود[۳۶]:

$$\text{Gain}(D,A_i)= \text{entropy}(D)-\text{entropy}_{Ai}(D) \quad (۲-۱۰)$$

دو نوع شاخص دیگر که اغلب به طور گسترده برای ارزیابی اینکه یک گره باید تقسیم شود یا نه استفاده

می شود، شاخص های Gini index و انحراف انتروپی است.

Gini index به صورت زیر تعریف می شود [۳۶]:

$$\text{Gini (T)}=1-\sum pj^2 \quad (۲-۱۱)$$

و انحراف انتروپی به صورت زیر تعریف می شود [۳۶]:

$$\text{entropy}(T) = -\sum pj \log_2 pj \qquad (۲-۱۲)$$

که p_j فراوانی نسبی کلاس j در درخت T می باشد.

اگرچه درختان تصمیم از روش های یادگیری محبوب هستند اما آنها مشکل overfitting دارند، پیش بینی با کیفیت بالا روی داده های آموزشی در حالی که داده های تست دیده نشده، دارای این کیفیت نیست.

۳-۵-۲- ماشین بردار پشتیبان (SVM):

ماشین های بردار پشتیبان نوع دیگری از سیستم های یادگیری هستند که دارای ویژگی های مطلوب بسیاری است که آن را یکی از محبوبترین الگوریتم ها نموده است. این الگوریتم در داده های با ابعاد بالا کاربرد دارد. برای نمونه، محققان زیادی نشان داده اند که ماشین بردار پشتیبان شاید دقیق ترین الگوریتم برای طبقه بندی متن باشد و هم چنین در طبقه بندی وب و کاربردهای بیوانفورماتیک به طور گسترده ای استفاده می شود. به طور کلی می توان گفت که ماشین بردار پشتیبان، یک سیستم یادگیری خطی است که طبقه بندی کننده با دو کلاس می سازد.

اگر نمونه آموزشی D به صورت زیر باشد:

$$\{(x_1,y_1), (x_2,y_2),\ldots\ldots (x_n,y_n)\}$$

که $x_i=(x_{i1},x_{i2},\ldots\ldots x_{ir})$ یک بردار ورودی r بعدی در فضای مقادیر حقیقی $x\subseteq R^2$; y_i برچسب کلاس و $y_i=\{1,-1\}$، ۱ کلاس مثبت و ۱- کلاس منفی را نشان می دهد. هر نمونه داده یک بردار ورودی خوانده می شود و با حروف تیره بزرگ نمایش داده می شود. در زیر ما از حروف تیره بزرگ برای همه بردارها استفاده کرده ایم.

برای ساخت یک طبقه بندی کننده، ماشین بردار پشتیبان یک تابع خطی به شکل زیر پیدا می کند.

$$f(x)=<\mathbf{W}.\mathbf{X}>+b \quad (*)$$

بنابراین اگر $f(x)\geq 0$ یک بردار ورودی $\mathbf{X_i}$، به کلاس مثبت اختصاص یافته است و برای کلاس منفی برعکس.

$$y_i = \begin{cases} ۱ & \text{اگر } <\mathbf{W}.\mathbf{X_i}>+b \geq 0 \\ -۱ & \text{اگر } <\mathbf{W}.\mathbf{X_i}>+b \leq 0 \end{cases}$$

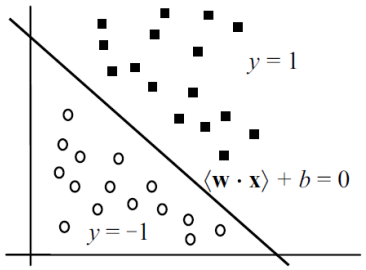
بنابراین $f(x)$ یک تابع حقیقی است، $f: x\subseteq R^2 \longrightarrow R$. $\mathbf{W=(w_1,w_2,\ldots..,w_r)}\in R^2$ که بردار وزن خوانده می شود و $b\in R$ پایه خوانده می شود.

بدون بردار معادله * به صورت زیر نوشته می شود:

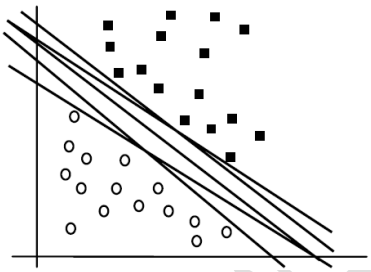
$$f(x_1,x_2,\ldots\ldots\ldots\ldots\ldots,x_r)=w_1x_1+w_2x_2+\ldots\ldots\ldots\ldots\ldots+w_rx_r+b$$

که x_i متغیر بیان کننده i امین مختصات بردار x می باشد. ماشین بردار پشتیبان یک ابر صفحه پیدا می کند($<w.x>+b=0$) که نمونه های مثبت و منفی را جدا می کند. این ابر صفحه، مرز تصمیم گیری یا صفحه تصمیم گیری نامیده می شود [۳۷].

ابر صفحه در فضای دوبعدی معمولاً خط و در فضای سه بعدی معمولاً صفحه است.طرح زیر یک فضای دو بعدی را نشان می دهد.



الف



ب

شکل۲-۵: الف: یک مجموعه تفکیک شده خطی شکل ب: مرزهای تصمیم گیری ممکن [۳۷]

در این طرح خطوط ضخیم در میانه، مرز تصمیم می باشند(در اینجا یک خط)، که نقاط مثبت و منفی را از هم جدا می کند. تعداد زیادی خط وجود دارد که نقاط مثبت و منفی را جدا می کند، ماشین بردار پشتیبان ابرصفحه ای را انتخاب می کند که حاشیه ها میان نقاط مثبت و منفی را بیشینه می کند. ماشین بردار پشتیبان در مورد داده های غیرخطی از توابع کرنل استفاده می کند.

برای پیدا کردن ابر صفحه بهینه نقاط $(x^+,1)$ و $(x^-,-1)$ را در نظر می گیریم و در ابر صفحه H^+ و H^- را تعریف می کنیم که از این دو نقطه بگذرند و موازی با صفحه $<\mathbf{w.x}>+b=0$ باشند و سپس حاشیه بین این دو ابر صفحه موازی را محاسبه می کنیم. با توجه به جبر خطی داریم [۲۴]:

$\frac{<\mathbf{W}.\mathbf{X}i>+b}{||\mathbf{W}||}$ که $||\mathbf{W}||$ برابر است با :

$$||\mathbf{W}||=\sqrt{<\mathbf{W}.\mathbf{W}>}=\sqrt{{w_1}^2+{w_2}^2+\cdots {w_r}^2}$$

حال فاصله ی نقطه ی x_s روی $<w.x>+b=0$ از صفحه $<w.x^+>+b=1$ محاسبه می کنیم. در نتیجه حاشیه برابر است با:

$$=\frac{1}{||\mathbf{W}||}\quad\frac{|<\mathbf{w}.xs>+b-1}{||\mathbf{W}||}$$

و به همین ترتیب حاشیه برای نقاط منفی محاسبه می شود ، بنابراین حاشیه برابر $\frac{2}{||\mathbf{W}||}$ خواهد بود. بیشینه سازی حاشیه یک مسئله بهینه سازی است و بیشینه کردن حاشیه همان کمینه کردن $\frac{||\mathbf{W}^2||}{2}$ است [۲۴].

۶-۲- ترکیب طبقه بندی کننده ها:

در خیلی از موارد، می توان طبقه بندی کننده ها را با هم ترکیب کنیم. دو تکنیک خوب و شناخته شده در این زمینه وجود دارد، bagging و boosting.

در هر دوی این روش ها، تعدادی طبقه بندی کننده ساخته می شود و طبقه بندی کننده تصمیم گیر نهایی برای هر نمونه تست، مبتنی بر انواع رای گیری می باشد.

۱-۶-۲- **Bagging**:

یک مجموعه آموزشی D با n نمونه داده شده و یک الگوریتم یادگیری پایه، bagging مطابق با آنچه در زیر آمده کار می کند:

یادگیری:

۱- k نمونه bootstrap، s_1، s_2 و s_k را خلق می کند. هر نمونه به صورت تصادفی از D با جایگزینی تولید می شود. هر نمونه s_i، ۶۳%/۲ نمونه های اصلی درD به همراه تعدادی نمونه ها که چندین بار آشکار می شوند را شامل می شود.

۲-طبقه بندی کننده مبتنی بر هر نمونه s_i ساخته می شود. این کار به ما k طبقه بندی کننده می دهد. همه ی طبقه بندی کننده ها با استفاده از الگوریتم یادگیری مشابه ساخته می شوند.

تست:
طبقه بندی کردن هر نمونه تست، با رای گیری K طبقه بندی کننده می باشد. کلاس اکثریت به عنوان کلاس نمونه اختصاص می یابد. bagging می تواند دقت را برای الگوریتم های یادگیری ناپایدار بهبود دهد.درخت تصمیم نمونه ای از روش های یادگیری ناپایدار هستند. روش های k-nearest و NaÏve Bayes نمونه ای از روش های پایدار هستند. برای طبقه بندی کننده پایدار، bagging می تواند گاهی دقت را کاهش دهد [۳۸].

۲-۶-۲- **Boosting**:

Boosting نیز مشابه bagging است، نمونه های یادگیری را دستکاری کرده و طبقه بندی کننده های چندگانه پایدار به منظور بهبود دقت تولید می کند [۳۹]. در اینجا ما الگوریتم مشهور Adaboost را شرح داده ایم.برخلاف bagging، Adaboost یک وزن را به هر نمونه آموزشی اختصاص میدهد [۴۰].
یادگیری :
Adaboost، دنباله ای از طبقه بندی کننده ها را تولید می کند(با استفاده از طبقه بندی کننده پایه). هر طبقه بندی کننده به قبلی وابسته است و ما روی خطاهای قبلی تمرکز می کنیم.
نمونه های آموزشی که به وسیله ی طبقه بندی کننده های قبلی به طور ناصحیح طبقه بندی شدند، وزن های بالاتری به آنها نسبت داده شده است.

مجموعه داده D را در نظر بگیرید$\{(x_1,y_1), (x_2,y_2),\ldots\ldots (x_n,y_n)\}$ که x_i بردار ورودی می باشد و y_i برچسب کلاس و$y_i \in Y$، یک وزن به هر نمونه اختصاص داده شده است، ما داریم $\{(x_1,y_1,w_1), (x_2,y_2,w_2),\ldots\ldots (x_n,y_n,w_n)\}$ و$\sum_i w_i = 1$. الگوریتم در طرح زیر داده شده است.

AdaBoost(D, Y, BaseLeaner, k)

1. Initialize $D_1(w_i) \leftarrow 1/n$ for all i; // initialize the weights
2. **for** $t = 1$ to k **do**
3. $f_t \leftarrow$ BaseLearner(D_t); // build a new classifier f_t
4. $e_t \leftarrow \sum_{i:f_t(D_t(\mathbf{x}_i))\neq y_i} D_t(w_i)$; // compute the error of f_t
5. **if** $e_t > ½$ **then** // if the error is too large,
6. $k \leftarrow k - 1$; // remove the iteration and
7. exit-loop // exit
8. **else**
9. $\beta_t \leftarrow e_t / (1 - e_t)$;
10. $D_{t+1}(w_i) \leftarrow D_t(w_i) \times \begin{cases} \beta_t & \text{if } f_t(D_t(\mathbf{x}_i)) = y_i \\ 1 & \text{otherwise} \end{cases}$; // update the weights
11. $D_{t+1}(w_i) \leftarrow \frac{D_{t+1}(w_i)}{\sum_{i=1}^{n} D_{t+1}(w_i)}$ // normalize the weights
12. **endif**
13. **endfor**
14. $f_{final}(\mathbf{x}) \leftarrow \underset{y\in Y}{\text{argmax}} \sum_{t:f_t(\mathbf{x})=y} \log\frac{1}{\beta_t}$ // the final output classifier

شکل۲-۶: الگوریتم Adaboost [۴۰]

الگوریتم با استفاده از یادگیر پایه دنباله ای از k طبقه بندی کننده می سازد(k به وسیله کاربر تعیین میشود)، در خط ۳ که base learner فراخوانی شده، در ابتدا وزن برای هر نمونه داده 1/n هست. در هر تکرار، مجموعه آموزشی D2 می شود که مشابه با D است اما با وزن های متفاوت. هر تکرار یک طبقه بندی کننده جدید f_t را در خط ۳ می سازد. خطای f_t در خط ۴ محاسبه می شود. اگر خطا خیلی بزرگ باشد، تکرار پاک می شود و خارج می شود (خطوط ۷-۵). خطوط ۱۱-۹ بروزرسانی و نرمال سازی وزن ها برای ساختن طبقه بندی کننده های بعدی می باشد.

تست:

برای هر نمونه تست، نتایج دسته طبقه بندی کننده ها برای تعیین کلاس نهایی نمونه تست ترکیب می شوند که در خط ۱۴ نشان داده شده است(رای گیری وزن دار).

Boosting در اغلب موارد بهتر از bagging کار می کند و تمایل به بهبود کارایی دارد [۴۱].

۲-۷- روش های ارزیابی:

در این قسمت ما اندازه های استاندارد بازیابی اطلاعات را شرح می دهیم.

۱-۷-۲-ارزیابی متقاطع[۳]:

وقتی که مجموعه داده کوچک باشد، ارزیابی متقاطعn-fold استفاده می شود، در این روش داده های موجود به n زیرمجموعه با اندازه مساوی تقسیم می شود. هر زیرمجموعه به عنوان مجموعه تست استفاده می شود و n-1 زیرمجموعه دیگر باقی می ماند که به عنوان مجموعه آموزشی برای یادگیری طبقه بندی کننده ترکیب می شوند. این روند n بار تکرار می شود و n دقت بدست می دهد. دقت نهایی تخمین زده شده از مجموعه داده، میانگین n دقت می باشد. اغلب از "10-fold" و "5-fold" استفاده می شود.

[3] Cross-Validation [4]

یک نوع خاص از ارزیابی متقاطع ، ارزیابی متقاطع leave-one-out است. در این روش هر fold فقط یک نمونه تست واحد دارد و همه ی بقیه داده ها به عنوان داده ی آموزشی استفاده می شوند. اگر داده اصلی m نمونه دارد، آن "m-fold" است. این روش زمانی استفاده می شود که داده موجود بسیار کوچک باشد و برای داده های زیاد کارا نیست [۴۲].

۲-۷-۲- دقت و فراخوانی:

فرض کنید که ما یک طبقه بندی کننده C و یک مجموعه اسناد D را داریم. طبقه بندی کننده C یک برچسب برای هر P∈D محاسبه می کند. C(P)=1 اگر P به عنوان هرزنامه طبقه بندی شده باشد و در بقیه موارد C(P)=0، الگوریتم هر سند در D به عنوان هرزنامه یا صفحه ی نرمال طبقه بندی می کند.

فرض کنید که ما متریک های زیر را تعریف کرده ایم:

مثبت درست(TP): اسناد هرزنامه ای که به عنوان هرزنامه تشخیص داده شده اند.

منفی درست(TN): اسناد نرمال که به عنوان نرمال تشخیص داده شده اند.

مثبت غلط(FP): اسناد نرمال که به صورت نادرست به عنوان هرزنامه طبقه بندی شده اند.

منفی غلط(FN): اسناد هرزنامه ای که به صورت نادرست به عنوان نرمال طبقه بندی شده اند.

	مثبت پیش بینی شده	**منفی پیش بینی شده**
مثبت	مثبت درست	منفی درست
منفی	مثبت غلط	منفی غلط

دقت :

کسری از اسناد به درستی تشخیص داده شده در مجموعه ای از اسناد که به عنوان هرزنامه تشخیص داده شده اند.

$$P=\frac{TP}{TP+FP} \qquad (۲-۱۳)$$

فراخوانی :

نمونه های مثبت درست طبقه بندی شده تقسیم بر کل نمونه های مثبت واقعی.

$$R=\frac{TP}{TP+FN} \qquad (۲-۱٤)$$

F-measure:

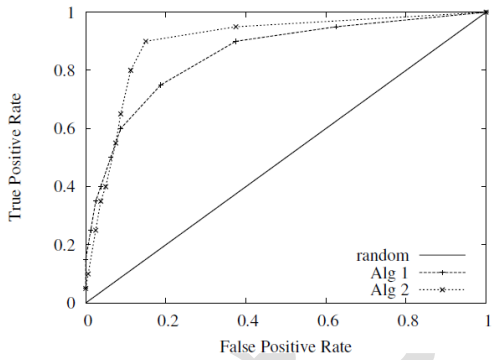
ترکیب دقت و فراخوانی است.

$$F=\frac{2PR}{P+R} \qquad (۲-۱۵)$$

۳-۷-۲- منحنی ROC:

در بیشتر موارد طبقه بندی کننده باینری نیست اما مقدار یا یک احتمال هرزنامه بودن را پیش بینی می کند. در اینگونه موارد می توانیم یک آستانه را به کار ببریم و در صورتی که مقدار آن از یک حد آستانه بیشتر باشد، برچسب هرزنامه را به یک صفحه بدهیم و مابقی صفحات به عنوان نرمال در نظر گرفته می شود.

روش دیگر بصری کردن کارایی وابسته به آستانه، منحنی ROC است. نام ROC از تئوری کشف سیگنال نشات گرفته است و از "مشخصه های عامل های گیرنده" می آید. این منحنی نرخ مثبت غلط ها را به عنوان تابعی از فراخوانی (میزان مثبت درست) به تصویر می کشد [۴۲].



شکل ۲-۷: منحنی ROC [۴۲]

۲-۸ – جمع بندی:

در این فصل ساختار وب مورد بررسی قرار گرفت. همچنین معماری موتورهای جستجو و رتبه بندی مبتنی بر محتوا و رتبه بندی مبتنی بر لینک بیان گردید. انواع هرزنامه ها بررسی شده و همچنین به تکنیک های یادگیری ماشین از قبیل درخت تصمیم، ماشین بردار پشتیبان و .... پرداخته شد.

فصل سوم
پیشینه تحقیق

۱-۳- مجموعه داده[۳۵]های مورد استفاده توسط محققین:

۱-۱-۳- UK2006:

این مجموعه هرزنامه وب که بصورت عمومی در دسترس است مبتنی بر جستجوی دامنه uk است، که در می ۲۰۰۶ توسط محققین آزمایشگاه الگوریتمهای و ب[۳۶] در یونیورسیت دگلی اِستادی دیمیلانو[۳۷] جمع آوری شده است.

این مجموعه داده با استفاده از نرم افزار UbiCrawler و توسط جستجوی اول پهنا[۳۸] به دست آمده است. پیمایش از یک مجموعه بزرگ از صفحات اولیه موجود در پروژه فهرست راهنمای باز[۳۹] شروع شده و مجموعه اولیه[۴۰] شامل بیش از ۱۹۰۰۰۰ URL در حدود ۱۵۰۰۰۰ میزبان بود. به‌عنوان یک نتیجه ۷۷/۹ میلیون صفحه متناظر با تقریباً ۱۱۴۰۰ میزبان جمع‌آوری شده بودند [۴۴].

این مجموعه مرجع در سطح میزبان توسط گروهی از داوطلبین برچسب گذاری شد. ارزیاب‌ها در گروه‌های دونفره میزبان‌ها را بصورت «عادی»، «مرزی»، «هرزنامه» یا «غیرقابل طبقه بندی» برچسب‌گذاری کرده در نتیجه هر میزبان توسط دو فرد و بصورت مستقل نامگذاری شد. ۶۵۵۲ ارزیابی بدست آمد. توزیع عناوین تخصیص داده شده توسط داورها در جدول زیر نشان داده‌شده است. بیشترین عنوان «عادی»، پس از آن «هرزنامه»، و سپس «مرزی» بود. تنها میزبان‌هایی مورد استفاده قرار داده شدند که ارزیاب‌ها در مورد آنها توافق داشتند، به‌علاوه میزبان‌هایی که در مجموعه به‌عنوان غیرهرزنامه علامتگذاری شده بودند، چرا که آنها متعلق به دامنه‌های خاصی مانند police.uk یا gov.uk بودند [۴۵].

جدول ۳-۱ :توزیع تعداد صفحات مرورشده توسط هر ارزیاب [۴۵]

درصد	تکرار	برچسب
۶۱/۷۵	۴۰۴۶	عادی
۱۰/۸۲	۷۰۹	مرزی
۲۲/۰۸	۱۴۴۷	هرزنامه
۵/۳۴	۳۵۰	غیرقابل طبقه بندی

۲-۱-۳- UK2007:

این مجموعه داده یک مجموعه بزرگ از میزبانهای هرزنامه و غیرهرزنامه است که مبتنی بر کاوش روی دامنه های uk می باشد که در می ۲۰۰۷ انجام گرفته است. مجموعه Web Spam UK2007، ۴۷۷۵ میزبان دارد که ۴۵۹۳ تا به عنوان عادی و ۱۸۲ تا به عنوان هرزنامه برچسب گذاری شده اند.

یکسری از ویژگی های از پیش محاسبه شده روی این میزبانها وجود دارد، بطورکلی UK2007، ۳۰۵ ویژگی را در برمی گیرد که به سه دسته متفاوت تقسیم می شوند ازجمله:

---

3 Data set
3 the Laboratory of Web Algorithmics
3 Universit`a degli Studi di Milano
3 BFS
3 Open Directory Project
4 Seed set

ویژگی های مستقیم:

این ویژگی ها فایل گراف محاسبه شده است و دو ویژگی را شامل می شود:

۱- تعداد صفحات در میزبان
۲- تعداد کاراکترها در نام میزبان

ویژگی های مبتنی بر لینک:

Feature set 2a: این مجموعه ویژگی های مبتنی بر لینک برای میزبانها را در بر می گیرد و در هر دو مورد صفحه خانگی و صفحه با بیشینه رتبه صفحه در هر میزبان اندازه گیری شده است. درجه ورودی، درجه خروجی، رتبه صفحه، لبه های متقابل،Trust rank ، truncated pagerank، تخمین پشتیبان ها[۱]و غیره،آنها ۵۸ ویژگی هستند.

Feature set 2b: ویژگی های مبتنی بر لینک تحول یافته که تحولات عددی ساده ویژگی های مبتنی بر لینک برای میزبانها هستند. این تحولات در عمل باعث بهتر شدن طبقه بندی شده اند نسبت به ویژگی های مبتنی بر لینک خام. اغلب نسبتهایی مابین ویژگی ها هستند، درجه ورودی/رتبه صفحه یا Trustrank/pagerank و لگاریتم ویژگی های متفاوت. این ویژگی ها ۱۴۹ مورد می باشند.

ویژگی های مبتنی بر محتوا:

که تعداد کلمات در صفحه اصلی، میانگین طول کلمات، میانگین طول عنوان و غیره را برای یک نمونه از صفحه روی هاست در بر می گیرد. آنها ۹۶ ویژگی هستند [۴۶].

۳-۱-۳- مجموعه داده جمع‌آوری شده با استفاده از جستجوی MSN:

یک مجموعه از ۱۰۵۴۸۴۴۴۶ صفحات وب که از جستجوی MSN جمع‌آوری شده‌اند که به‌عنوان یک پروکسی برای وب ارائه می‌شود. این صفحات، در طول آگوست ۲۰۰۴ جمع‌آوری شده‌اند و به طور اختیاری از کاوش جستجوی کامل MSN حاصل شده‌اند.

کاوش جستجوگر MSN، صفحات جدید را با استفاده از یک سیاست بررسی اول-عرض[۲] تشخیص می‌دهد و از تخمین‌های مختلف مهمی برای زمانبندی پیمایش مجدد صفحاتی که تاکنون تشخیص داده شده‌اند، استفاده می‌کند.

بنابراین صفحاتی که با استفاده از چنین سیاستی پیمایش شده‌اند، ممکن است یک توزیع تصادفی یکنواخت را دنبال نکنند؛ کاوش جستجوی MSN، به سمت صفحات با اتصال خوب، مهم و با کیفیت بالا، گرایش دارد. به‌علاوه، کاوش جستجوی MSN، از هیوریستیک‌های زیادی جهت تشخیص هرزنامه‌ی استفاده می‌کند.

در ابتدا، گرچه این پیمایشگر بر صفحات با اتصال خوب و مهم تمرکز دارد، این صفحات به‌طور معمول، توسط موتورهای جستجو، رتبه‌بندی بالایی دارند. بنابراین، تعداد هرزنامه ای که گزارش می-شود به‌طور تقریبی به چیزی که در نهایت توسط کاربران موتورهای جستجو درک می‌شود، نزدیک می‌شوند [۴۹].

[1] Supporter
[2] BFS

۴-۱-۳-DC2010[۴۳]: DC2010 یک مجموعه بزرگ از میزبان های وب برچسب گذاری شده به وسیله آکادمی علوم مجارستان (اسناد انگلیسی)، بنیاد حافظه اینترنت[۴۴](به زبان فرانسوی) و L3S هانوفر به زبان آلمانی است، پایه و اساس آن یک مجموعه 23M از صفحات است در 190 میزبان در دامنه های eu. که اوایل سال ۲۰۱۰ به وسیله بنیاد حافظه اینترنت پیمایش شده است. برچسب ها حوزه مجموعه داده های قبلی روی هرزنامه وب را گسترش داده اند. به علاوه برای سایت هایی دارای برچسب هرزنامه، طبقه بندی دستی برای نوع و کیفیت قرار داده اند.

انگیزه پشت این روش برچسب گذاری نیاز به یک آرشیو اینترنتی ساختگی است که ممکن است بخواهیم و یا نخواهیم کاملاً هرزنامه را حذف کنیم اما نوع خاصی از محتوا نظیر اخبار یا آموزشی را ماورای سایتهای تجاری ترجیح می دهیم.
هم چنین آنها ممکن است یک اولویت بالاتر به محتوای قابل اعتماد، واقعی و بی طرفانه بدهند که با یک نمره سودمندی[۴۵]ترکیب می شود.
DC2010 میزبان های برچسب گذاری شده با ویژگیهای متفاوت را دربرمی گیرد، اعتماد، واقعی و بی طرفانه بودن، ۵ نوع برای استفاده در طبقه بندی انتخاب شدند. از آنجایی که هیچ برچسب دیگری برای میزبان های هرزنامه ساخته نشده، ویژگیهای دیگر و به ویژه ۵ نوع تحریریه، تجاری، آموزشی، بحث و گفتگو و شخصی غیرمنحصر به فرد هستند و بنابراین مسئله دسته بندی باینری را مطرح کرده اند.
ارزیاب ها نخست به بررسی این مورد می پردازند که چرا میزبان ها نمونه را اصلاً در بر نمی گیرد از جمله بزرگسالان، آمیخته[۴۶]؛ سایت های طبقه بندی نشده زبانی. سپس هرزنامه وب براساس تعریف کلی "هر عملی که منجر به رتبه بندی نابجا شود، با توجه به ارزش درست صفحه" مورد شناسایی قرار گرفت. ارزیاب ها با مطالعه راهنمای Web Spam UK آموزش داده شدند. در DC2010 سه زبان انگلیسی، آلمانی و فرانسوی برچسب گذاری شده اند، اگرچه که زبان لهستانی و هلندی کسر بزرگتری از زبان فرانسه را در بر می گیرد [۴۸].

در جدول ۳-۲ مقدار هرزنامه DC2010 را در مقایسه با Web Spam UK2006 و Web Spam UK2007 خلاصه شده است.

جدول ۳-۲: کسری از هرزنامه ها در DC2010 و Web-spam –UK2006 [۴۸]

	UK2006	UK2007	DC2010			
			en	de	fr	All
Hosts	**10660**	**114529**	**61703**	**29758**	**7888**	**190000**
Spam	**19.8%**	**5.3%**	**8.5% of valid labels;5% of all in large domains**			

فراتر از هرزنامه، میزبان ها باید به وسیله نوعشان به دسته های زیر برچسب گذاری شوند، یک فهرست تنظیم شده براساس تست های ارزیاب ها:

۱. محتوای تحریریه یا خبری[۴۷]آشکارسازی پستها، اعلان، انتشار اخبار، انتشار متنهای واقعی روی یک وضعیت از امور مانند اخبار روز (از جمله ورزشی) و گزارش های پلیس. ارسال پستها، آنالیز کردن یک موضوع اقتصادی، تکنولوژیکی، محیطی و اجتماعی از جمله آگهی های تبلیغاتی، سیاسی.

3 Discover Challenge
4 Internet Memory Foundation
5 utility score
6 mixed
7 news

۲. محتوای تجاری[۴۸]: بررسی محصول، فروشگاه برخط، کاتالوگ محصول، کاتالوگ سرویس ها، محصولات مرتبط با آن، پرسش های متداول و آموزشی.
۳. محتوای آموزشی[۴۹]: پژوهشی: آموزش ها، کتاب های راهنما، چگونگی راهنمایی، مواد آموزشی، موارد دستوری، مقالات پژوهشی، کتاب، فهرست ها، واژه نامه ها، همایش ها، موسسات، صفحات پروژه، بهداشت و درمان هم چنین تعلق به این قسمت دارند.
۴. فضاهای بحث[۵۰]: شامل انجمن های اختصاصی، فضاهای چت، بلاگ ها و غیره، کامنت های استاندارد را به حساب نمی آوریم.
۵. شخصی[۵۱] یا اوقات فراغت: هنر، موزیک، خانه و خانواده و بچه ها، بازی ها، طالع بینی و غیره، یک بلاگ شخصی برای نمونه هم به " بحث" و هم به این قسمت تعلق دارد.
۶. رسانه ها[۵۲]: تصویری، صوتی و غیره. سایتی که محتوای اصلی آن متن نیست اما رسانه هست. برای نمونه سایت درباره موزیک شاید جزء اوقات فراغت است و نه رسانه.
۷. پایگاه داده[۵۳]: یک سایت " وب عمیق " که محتویات آن تنها با پرسش از یک پایگاه داده قابل بازیابی است، سایت های ارائه دهنده فرم ها در این دسته قرار می گیرند.

در نهایت ویژگیهای کلی مرتبط با اعتماد، بی طرفی با سه مقیاس برچسب گذاری شدند:

۱. معتبر و موثق بودن[۵۴]: من نمی توانم اعتماد کنم – جنبه هایی در این سایت وجود دارد که این منابع را برای من بی اعتماد می کند. من به این حواشی اعتماد دارم – یک منبع معتبر به نظر میرسد اما مالکیت آن نامشخص است. من کاملاً اعتماد دارم ، این یک منبع معتبر معروف (یک روزنامه ، کمپانی ، سازمان معروف)
۲. تبعیض قائل شدن[۵۵]: حقایق – من فکر میکنم که این عمدتاً حقایق هستند. حقایق و نظرات – من فکر میکنم حقایق و نظرات هستند، حقایق در سایت گنجانده شده اند یا از منابع خارجی مورد ارجاع قرار گرفته اند. نظر و عقیده – من فکر میکنم این بیشتر عقیده و نظر به نظر می رسد که ممکن است به وسیله حقایق پشتیبانی شود یا نشود اما حقایق اندکی یا هیچ حقیقتی را در بر نگرفته و یا مورد ارجاع قرار نگرفته است.
۳. بی طرفی[۵۶]: تعریف ما از ویکی پدیا[۵۷] اقتباس شده است [۴۸].

---

[4] Commercial [48]
[4] Educational [49]
[5] Discussion [50]
[5] Personal [51]
[5] Media [52]
[5] Database [53]
[5] Trustworthiness [54]
[5] Neutrality [55]
[5] Bias [56]
[5] Wikipedia [57]

جدول ۳-۳ : توزیع برچسب ها در مجموعه داده DC2010 [۴۸]

Label	Yes	Maybe	No
Spam	423		4982
New/Editorial	191		4791
Commercial	2064		2918
Educational	1791		3191
Discussion	259		4724
Personal-Leisure	1118		3864
Non-Neutrality	19	216	3778
Bias	62		3880
Dis-Trustiness	26	201	3786
Confidence	4933		49
Media	74		4908
Database	185		4797
Readability-Visual	37		4945
Readability-Language	4		4978

۲-۳-مطالعات مبتنی بر محتوا:

کار اصلی روی الگوریتم های ضد هرزنامه مبتنی بر محتوا توسط فترلی و همکاران انجام شده است. آنها پیشنهاد می کنند که صفحات وب توسط آنالیز آماری تشخیص داده شوند. محققان دریافته اند URL های صفحات هرزنامه تعداد نقطه، خط تیره، ارقام و طول استثنایی دارند. آنها گزارش داده اند که ۸۰ مورد از ۱۰۰ مورد طولانی ترین نام های میزبان به وب سایت های بزرگسالان ارجاع شده است. آنها هم چنین نشان داده اند که صفحات خودشان ماهیت تکثیر دارند. بیشتر صفحات هرزنامه که روی همان میزبان قرار دارند، مغایرت خیلی کمی در تعداد کلمات دارند. یکی از مشاهدات جالب این است که صفحات هرزنامه محتوایشان به سرعت در حال تغییر است. به طور خاص آنها تغییرات هفته به هفته روی تمام صفحات وب روی یک میزبان را مورد مطالعه قرار داده اند و متوجه شده اند که میزبان های هرزنامه تا ۹۷ درصد با این ویژگی قابل کشف هستند [۴۷].

در کار دیگر آنها از یک مجموعه داده منتخب که عمده‌ی صفحات (حدود ۵۴%) به زبان انگلیسی نوشته شده‌اند که توسط کاوشگر نوشته‌شده توسط جستجوی MSN تعیین شده‌است، استفاده کرده اند. در این مجموعه داده ۲۳۶۴ صفحه، (۱۳/۸%) به عنوان هرزنامه برچسب خوردند در حالی که ۱۴۸۰۴(۸۶/۲%) به‌عنوان غیرهرزنامه برچسب خوردند.

به هیوریستیک‌های مختلفی جهت تشخیص هرزنامه توسط فترلی و همکاران اشاره شده است از جمله: تعداد کلمات در صفحه، تعداد کلمات در عنوان صفحه، طول میانگین کلمات، مقدار متن Anchor، بخشی از محتوای قابل مشاهده(برای مثال مؤلفه‌هایی نظیر نظرات در بدنه صفحه، یا ویژگی ALT

است که به تصاویر اختصاص دادهشدهاست یا برچسبهای META در سرآیند)، قابلیت تراکمپذیری، بخشی از صفحه برگرفته از لغات عمومی کلی، کسری از کلمات محبوب عمومی، احتمال استقلال n-gram، احتمال شرطی n-grams اشاره شده است. نهایتاً از درخت C4.5 به منظور طبقه بندی استفاده شده است و به منظور افزایش دقت از bagging و boosting استفاده شده است [۴۹].

در کار دیگر در آنها محتویات تکراری را مورد مطالعه قرار دادند [۵۱,۵۰]. متوجه شدند که بزرگترین خوشه ها با محتوای تکراری هرزنامه هستند، برای پیدا کردن خوشه ها ومحتویات تکراری آنها روش shingling مبتنی بر rabin fingerprint را به کار برده اند [۵۳,۵۲].

آنها نخست هر کدام از n کلمه را با استفاده از یک چندجمله ای P_A، fingerprint می کنند و سپس هر کدام از نشانه های مرحله نخست را با یک چند جمله ای متفاوت P_B با استفاده از حذف پیشوندها و توسعه تغییرات، fingerprint می کنند، در سومین مرحله آنها m تابع fingerprint متفاوت را برای هر رشته از مرحله اول به کار می برند و کوچکترین مقادیر نتیجه را برای هر کدام از m تابع fingerprint حفظ می کنند. آنها همچنین فهرستی از عبارات محبوب با مرتب کردن سه تایی های (i,s,d) استخراج می کنند و اجراهای طولانی از سه تایی ها با تطبیق دادن مقادیر i و s می گیرند. در [۴۹] آنها مطالعات خود را ادامه داده اند و تعدادی از ویژگی های مبتنی بر محتوای دیگر را هم فراهم کرده اند. نهایتاً همه این ویژگی ها در یک مدل طبقه بندی با C4.5، boosting و bagging ترکیب شده اند. مطالعاتی نیز شرح می دهد که چگونه ویژگی های مختلف و مدلهای یادگیری ماشین به منظور کیفیت الگوریتم های کشف هرزنامه با هم ترکیب می شود [۴۸].

گروه دیگری به معرفی ویژگی های مبتنی بر ساختار صفحه HTML به منظور شناسایی اسکریپت های تولید شده صفحات هرزنامه پرداخته اند [۵۵]. ایده اساسی که صفحات اسپم تولید شده ماشین هستند در [۵۰, ۵۱] بحث شده است. هر چند که محققین یک گام پیش پردازش را با برداشتن محتوا و نگه داشتن فقط طرح صفحه هم اضافه نموده اند. بنابراین آنها تکرار صفحه را با آنالیز کردن طرح و نه محتوا مطالعه کرده اند.آنها برای پیدا کردن گروه صفحات اسپم تکراری تکنیک انگشت نگاری را به همراه خوشه بندی به کار برده اند [۵۳,۵۲].

در کاری دیگر از مدل های زبان[5]برای کشف هرزنامه استفاده شده است. شاخه ای از کشف هرزنامه در بلاگ ها با مقایسه مدلهای زبانی برای نظرات بلاگ و صفحات ارائه شده اند [۵۷,۵۶]. آنها از واگرایی KL به عنوان یک معیار از اختلاف بین دو مدل زبانی (توزیع احتمال) استفاده می کنند [۹۴].

$$KL(\ominus 1 || \ominus 2)=\sum_w p(w|\ominus 1) \log \frac{p(w|\ominus 1)}{p(w|\ominus 2)} \qquad (۱-۳)$$

ویژگی مفید این روش آن است که نیازی به داده های آموزشی ندارد. آنالیزی از ویژگی های زبانی با در نظر گرفتن اعتبار واژگانی، تنوع واژگان و محتوا، تنوع نحوی و آنتروپی، استفاده از صداهای فعال و غیرفعال و سایر ویژگی های زبان های طبیعی (NLP) نیز توسعه داده شده است [۵۹,۵۸].

[5] Language Model [8]

ویژگی هایی بر اساس وقوع کلمات کلیدی روی یک صفحه که ارزش های تبلیغاتی بالا دارند را پیشنهاد می دهد [۶۰]. در موتورهای جستجو لاگ فایل های پرس و جو و لاگ فایل های کلیک روی تبلیغات آنلاین با توجه به محبوبیت پرس و جو مورد آنالیز قرار گرفته اند [۶۱].

استفاده از الگوریتم های یادگیری ماشین برای تشخیص هرزنامه مبتنی بر محتوای در صفحات عربی

با توجه به اینکه تعدادی از وب سایتها که در Web Spam uk2007 آمده موجود نیست و هم چنین بنا به ضرورت محاسبه ویژگیهای جدید، مجموعه داده جدیدی به نام uk2011 ساخته شده و به عنوان یک مجموعه جدید جایگزین شده است. مجموعه داده جدید، ۳۷۰۰ صفحه ی وب را در بر می گیرد [۶۷]. هم چنین با توجه به مجموعه داده Wahshah و همکاران که ۴۰۰ صفحه ی وب را در بر میگیرد، اقدام به توسعه ی یک مجموعه داده عربی با ۱۰۰۰۰ صفحه وب عربی گردیده و ویژگی های جدید استخراج شده است. صفحات به صورت دستی برچسب گذاری شده اند [۶۷].

ویژگی های مورد استفاده:

از برخی ویژگی های مورد استفاده مطالعات قبلی بهره گرفته و علاوه بر آنها سه ویژگی جدید نیز ارائه گردیده است [۶۲,۶۳, ۶۴, ۶۵, ۶۶]. ویژگی های جدید ارائه شده عبارتند از:

۱) تعداد کل کلمات در برچسب <Meta>: keystuffing در عمل کلمات کلیدی در عناصر html هستند که به تعداد دفعات زیاد تکرار می شوند. هرزنامه نویسان از stuffing در برچسب متا استفاده می کنند که هدف آن جاسازی محتوای صفحات وب عربی با کلمات محبوب است.
۲) کمینه یا بیشینه طول کلمه در صفحه وب : هرزنامه نویسان سعی می کنند طول کلمات کلیدی مهم یا محبوب را در صفحه ی وب افزایش دهند. به منظور شناسایی این ویژگی ها نیاز به دانستن کمینه یا میانگین طول کلمات در صفحات غیرهرزنامه داریم.
۳) تعداد کل تصاویر در صفحه ی وب. افزایش تعداد تصاویر در صفحه ی وب می تواند منجر به جذب کاربران بیشتر شود و رتبه صفحه را در نتایج جستجو بهبود دهد.
۴) متدلوژی مورد استفاده:

از دو الگوریتم یادگیری ماشین (درخت تصمیم و Naïve Bayes) استفاده شده است. یک مجموعه داده از صفحات وب عربی شامل ۱۰۰۰۰ صفحه ی ساخته شده است. به علاوه از نسخه بروزرسانی شده UK2007 که UK2011 نامیده شده و شامل ۳۷۰۰ صفحه می باشد استفاده شده است.

۵) تقسیم بندی کار به چهار قسمت:

۱-محاسبه ویژگی ها برای تشخیص هرزنامه وب

۲- برای مقایسه Extended Arabic 2011 و UK2011 از ویژگی های ارائه شده در [۶۲] استفاده شده است.

۳-هم چنین دو مجموعه داده Extended Arabic 2011 و UK2011 برای ویژگی های جدید مقایسه شدند.

۳- نهایتاً همه ویژگی ها در یک گروه ادغام و این دو مجموعه داده روی این ویژگی ها مقایسه شدند.

۶) نتایج :

الگوریتم درخت تصمیم j48 و NB به کار برده شده است. الگوریتم درخت تصمیم نسبت به NB برای تشخیص هرزنامه کاراتر بوده و مجموعه داده Extended Arabic-2011 نسبت به Uk-2011 بهتر عمل می کند [۶۷].

۳-۳-روش های مبتنی بر لینک:

تمام الگوریتم‌های کشف هرزنامه مبتنی بر لینک می‌توانند به ۴ گروه تقسیم شوند. گروه اول از رابطه مکانی (فاصله، co-citation، تشابه) بین صفحات وب و مجموعه‌ای از صفحات برای برچسب های شناخته شده استفاده می‌کنند. گروه دوم الگوریتم‌ها، روی شناسایی گره‌ها و لینک‌های مشکوک و پایین آوردن وزن آن‌ها تمرکز می‌کنند. گروه سوم، بوسیله استخراج ویژگی‌های مبتنی بر لینک برای هر گره عمل می‌کند و الگوریتم‌های یادگیری ماشین متنوع را برای کشف هرزنامه به کار می‌برند. گروه چهارم الگوریتم‌های مبتنی بر لینک، از ایده‌ی پالایش برچسب‌ها مبتنی بر پیکربندی گراف وب استفاده می‌کنند، که برچسب های پیش بینی شده به وسیله الگوریتم پایه با استفاده از انتشار از طریق گراف فوق پیوند اصلاح می شوند [۶۸].

۱-۳-۳-الگوریتم‌های مبتنی بر انتشار برچسب‌ها:

ایده اصلی این الگوریتم این است که یک مجموعه از صفحات با برچسب های شناخته شده را در نظر گرفته و برچسب های گره های دیگر براساس قوانین انتشار محاسبه شود.

یکی از اولین الگوریتم ها در این دسته Trustrank است که اعتماد را از مجموعه صفحات خوب از راه رتبه بندی شخصی انتشار می دهد. در واقع ایده اصلی این الگوریتم این است که صفحات قابل اعتماد عمدتاً به صفحات قابل اعتماد دیگر لینک می شوند و به ندرت به صفحات هرزنامه لینک می شوند [۶۸].

الگوریتم TrustRank**:**

برای انتخاب یک مجموعه صفحات خوب استفاده از رتبه شخصی معکوس پیشنهاد می شود که به عنوان یک گراف با لبه های برعکس عمل می کند. با داشتن امتیاز رتبه صفحه معکوس محاسبه شده برای تمام صفحات روی وب، K صفحه بالایی را در نظر گرفته می‌شود و به کاربران اجازه داده می-شود تا در مورد اعتبار این صفحات قضاوت کنند. سپس یک بردار رتبه صفحه شخصی ایجاد کردند که اجزا فقط به صفحات قضاوت شده معتبر مطابقت دارند که غیر صفر هستند. در نهایت، رتبه صفحه شخصی شده محاسبه شده است. TrustRank ویژگی های بهتری را نسبت به رتبه صفحه برای کاهش رتبه هرزنامه وب نشان می‌دهد.

این الگوریتم نمره اعتماد را به هر صفحه تخصیص می دهد، مجموعه فرزند D یک نمره اعتماد مثبت اولیه دارد و به همه صفحات دیگر نمره صفر تخصیص داده شده است.

$$t_{0,i} = \begin{cases} 1/|D| & i \in D \text{ اگر} \\ 0 & \text{در بقیه موارد} \end{cases}$$

$$t_{i+1} = \alpha\beta t_i + (1-\alpha)t_0$$

t_i بردار نمرات اعتماد در تکرار i است ، B ماتریس انتقال و α فاکتور تنزل است .t_0 بردار شخصی شده است .سخت ترین قسمت الگوریتم انتخاب مجموعه فرزند است ، روش های متعددی برای این کار استفاده می شود ، یکی از آنها محاسبه رتبه معکوس روی گراف وب به منظور شناسایی سایت هایی است که از تعداد زیادی از سایت های دیگر در دسترس هستند [۶۹,۶۸].

الگوریتم Anti-Trustrank:

مقابل TrustRank، انتشار بی اعتمادی از یک مجموعه صفحات هرزنامه شناخته شده روی یک گراف معکوس مورد توجه قرار گرفت. یک مجموعه اولیه بین صفحات با مقادیر رتبه صفحه بالا انتخاب شده است.Anti-TrustRank بر این اساس است که به ندرت اتفاق می‌افتد صفحه‌ی خوبی به صفحه‌ی بدی اشاره کند. همچنین این اصل می‌رساند که صفحاتی که به هرزنامه‌ها اشاره دارند خودشان به احتمال زیاد هرزنامه هستند.

الگوریتم Anti–Trust Rank از مجموعه بنیادی صفحات حاوی هرزنامه که به طور دستی برچسب-گذاری شده‌اند، آغاز شده و در جهت معکوس در امتداد پیوندهای ورودی انتشار می‌یابد. صفحه‌ای را که دارای رتبه ارزش TrustRank بیشتر از ارزش آستانه مشخص شده بود، به عنوان صفحه‌ی حاوی هرزنامه طبقه بندی می‌شود. اصل تقریبی جداسازی به طور کل ما را قادر به تشخیص صفحات خوب از صفحات نه چندان خوب می‌کند [۶۸].

همچنین ایده انتشار توسط تحلیل این که چگونه استراتژی های گسترش اعتماد و بی‌اعتمادی می‌توانند با هم کار کنند، مورد بررسی قرار گرفته است. اعتماد در الگوریتم TrustRank انتشار یافته است - هر فرزند یک قسمت مساوی از $c.\frac{TR(p)}{|Out(p)|}$ اعتماد و درستی پدر را به دست می‌آورد، دو استراتژی ارائه شده است:

- تقسیم پایدار[5]زمانی که هر فرزند همان قسمت کاهش یافته از امتیاز c.TR(p) درستی پدر را بدون توجه به تعداد فرزندان به دست می‌آورد.
- تقسیم لگاریتمی[6]زمانی که هر فرزند یک قسمت مساوی از امتیاز پدر را که توسط الگوریتم تعداد فرزندان $c.\frac{TR(p)}{log(1+|Out(p)|)}$نرمال شده است را به دست می‌آورد [۴۳].

یکی‌دیگر از استراتژی‌ها، استراتژی تجمیع اعتماد جزیی مختلف است، در حالی‌که TrustRankدر واقع مجموع مقادیر اعتماد از هریک از والدین است. به طور خاص، در اینجا استراتژی سهم حداکثر در نظر گرفته شده است، زمانی که مقدار حداکثر ارسال شده توسط والدین، استفاده شده

[5] Constant splitting
[6] Logarithmic splitting

است؛ و استراتژی والد حداکثر، هنگامی‌که انتشار انجام شده است برای ضمانت این است که امتیاز فرزند از حداکثر نمره والدین تجاوز نکند. در نهایت، از یک ترکیب خطی از مقادیر اعتماد و بی-اعتمادی استفاده می‌شود [۴۸]:

$$TotalScore(p)= \eta.TR(p) – \beta . AntiTR(p)$$

که $\beta, \epsilon\eta(0,1)$ است. براساس آزمایشات انجام شده، ترکیبی ازهردو استراتژی انتشار درتنزل رتبه هرزنامه نتیجه بهتری می دهد(۸۰ درصد ازسایتهای هرزنامه از ۱۰ تای بالادرمقایسه TrustRank و PageRank ناپدید می‌شوند)، سهم حداکثر با تقسیم لگاریتمی (Logarithmic splitting) بهترین راه برای محاسبه مقادیر اعتماد و بی‌اعتمادی است.

تعدادی از الگوریتم‌ها ویژگی تجزیه رتبه صفحه را مورد استفاده قرار می‌دهند تا مقدار رتبه صفحه نابجا را که از گره‌های مشکوک می‌آید، برآورد کنند [۷۱,۷۰]. یکی از این الگوریتم SpamRank است؛ که پشتیبان ها را برای یک صفحه با استفاده از شبیه‌سازی‌های مونت کارلو می‌یابد، با تحلیل این که آیا امتیاز *رتبه صفحه* شخصی $PPR(\overrightarrow{X_j})_i$ با بایاس گره‌های مشکوک توزیع شده است، یک نمره پنالتی به هر صفحه اختصاص می‌دهد و در نهایت SpamRank را برای هر صفحه به عنوان یک PPR با بردارهای شخصی مقداردهی اولیه شده با نمره پنالتی محاسبه می‌کند. ماهیت الگوریتم در اختصاص نمرات پنالتی است. تمام حمایت‌کننده‌های یک صفحه را توسط امتیازات رتبه صفحه آن‌ها با استفاده از ذخیره کردن با افزایش پهنا به طور نمایی، محاسبه همبستگی بین شاخص bin و لگاریتم تعداد دفعات مشاهده شده در bin، و سپس اختصاص امتیاز پنالتی به پشتیبان ها به وسیله مجموع امتیازات صفحاتی که آن‌ها پشتیبانی می‌کنند،تفکیک می‌شوند.

مفهوم توده هرزنامه[1]مقدار را که از صفحات هرزنامه می‌آید اندازه گیری می‌کند [۷۲]. مشابه TrustRank به هسته صفحات خوب شناخته شده احتیاج دارد تا مقدار رتبه صفحه را که از صفحه های هرزنامه می‌آید تخمین بزند. این الگوریتم در دو مرحله کار می‌کند. ابتدا، بردارهای $\vec{\pi}$ PageRank و $\vec{\pi}$TrustRankرا محاسبه می‌کند و مقدار توده هرزنامه هر صفحه را با استفاده از فرمول $=\frac{\vec{\pi}-\vec{\pi}}{\vec{\pi}}\vec{m}$ محاسبه می‌کند. سپس، حد آستانه، که به مقدار توده هرزنامه وابسته است، ایجاد می‌شود. شایان ذکر است که الگوریتم می‌تواند دانش را در مورد صفحات بد به طور مؤثری به کار ببرد.

الگوریتم دیگری برای تشخیص هرزنامه براساس تحلیل لینک مبتنی بر اعتبار است. در این جا مفهوم k-ScopedCredibility برای هر صفحه تعریف می‌شود و توسط چندین روش تخمین زده می‌شود. ابتدا مفهوم BadPath تعریف می‌شود، یک قدم زدن تصادفی k جهشی[2]با شروع از یک صفحه درست و پایان در یک صفحه هرزنامه، و سپس محاسبه نمره tuned k-Scoped Credibility به صورت زیر [۷۳]:

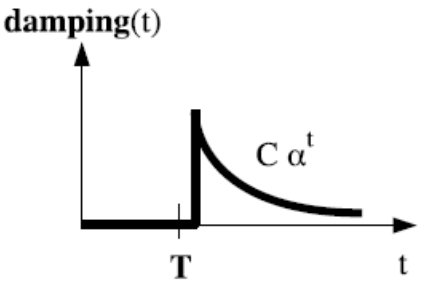
$$C_K(\mathrm{p}) = \left\{1 - \sum_{l=1}^{k}\left[\sum_{path_l(p)\in Badpath_l(p)} P(path_l(p))\right]\right\}(\mathrm{p})\gamma \qquad (۲-۳)$$

[1] Spam mass
[2] k-hop random walk

که k پارامتر تعیین کننده طول یک قدم زدن تصادفی است، $\gamma(p)$ یک فاکتور منفی اعتبار است که فقط به مقدار دانش جزئی از کل صفحات هرزنامه روی وب نیاز داشته است، و $P(path_l(p))=\prod_{i=0}^{l-1}\omega_{ii}+1$. نمره اعتبار می‌تواند برای پایین آوردن وزن یا هرس کردن لینک‌های با اعتبار پایین قبل از رتبه‌بندی مبتنی بر لینک یا برای تغییر بردار شخصی در PPR، TrustRank، یا Anti-TrustRankاستفاده شود [۷۳].

۲-۳-۳- رتبه بندی تابعی:

کار اصلی به وسیله بایزا-یتز[۳]و همکاران انجام شد. آنها مفهوم رتبه بندی تابعی را بیان کردند که تعمیم PageRank به وسیله توابع تعدیل مختلف است [۷۴]. تحقیقات نشان داده‌اند که صفحات هرزنامه باید به تغییرات در ضریب تعدیل محاسبه رتبه صفحه بسیار حساس باشند. یک راه کاهش رتبه صفحات هرزنامه در نظر گرفتن ضریب تعدیلی است که مشارکت مستقیم نخستین سطح از لینک‌ها را حذف می‌کند.



$$\text{Damping}(t)=\begin{cases}0 & t \le T\\ C\alpha^{t} & t > T\end{cases}$$

تابع رتبه صفحه تعدیل شده که C یک ثابت نرمال‌سازی و α ضریب تعدیل مورد استفاده برای رتبه صفحه ، t طول مسیرها است. این تابع صفحاتی که سهم بزرگی از رتبه خود را از چند سطح نخست لینک‌ها بدست می‌آورند جریمه می‌کند. یک روش خیلی سریع برای محاسبه رتبه صفحه تعدیل شده وجود دارد. با توجه به محاسبه رتبه صفحه، تصاویر لحظه‌ای از مقادیر رتبه صفحه در تکرارهای مختلف را ذخیره کرده و سپس تفاوت را محاسبه کرده و مقادیر نهایی محاسبه رتبه صفحه نرمال‌سازی می شود. ضرورتاً این بدین معنی است که رتبه صفحه تعدیل شده را می‌توان بدون هزینه در حین تکرارهای رتبه صفحه محاسبه نمود.

باید توجه نمود که به تعداد همسایه‌های غیرمستقیم به تعداد همسایه‌های مستقیم نیز بستگی دارد، کاهش مشارکت سطح نخست لینک‌ها توسط این روش بدین معنی نیست که چیزی کاملاً متفاوت از رتبه صفحه محاسبه می‌شود. در واقع، برای بیشتر صفحات، هر دو معیار همبستگی شدیدی با هم دارد. در عمل، مشاهده می‌شود که برای میزبان‌های هرزنامه، رتبه صفحه تعدیل شده کمتر از رتبه صفحه است.

هم چنین محققین از توابع تعدیل عمومی برای کشف هرزنامه استفاده کرده و الگوریتم truncatedPageRank را ارائه داده اند که از مدل نمایی کوتاه شده استفاده می‌کند [۷۵]. این الگوریتم بر

[6] Baeza-Yates [3]

این اساس است که صفحات هرزنامه تعداد زیادی پشتیبان در فواصل نزدیک دارند، در حالی‌که این تعداد کمتر از تعداد مورد انتظار در فواصل بیشتر است. به این دلیل، توابع تعدیل برای آن که سهم مستقیم اولین سطوح از لینک‌های داخلی را نادیده بگیرد، استفاده می‌شود.

$$\text{damping(j)} = \begin{cases} 0 & if\ j \le J, \\ D(1-c)^{j} & otherwise \end{cases}$$

در این روش از یک الگوریتم شمارش استفاده می‌شود تا تعداد پشتیبان‌های یک صفحه به طور مؤثر و کارآمد تخمین زده شود.

۳-۳-۳- الگوریتم‌های هرس لینک و وزن‌دهی دوباره

الگوریتم‌های متعلق به این طبقه برای یافتن لینک‌های نامطمئن و کاهش آن‌ها هستند. کار اصلی انجام شده در این الگوریتم‌ها این است که مشکلات موجود در الگوریتم HITS را مورد بررسی قرار می‌دهد، مثل تسلط روابط تقویت کننده دو طرفه و انحراف موضوع[4]گراف همسایه، و روش‌های پیشنهاد شده راه حل‌ها افزودن یک تحلیل لینک با یک تحلیل محتوا است. یکی از روش‌ها این است که اگر k صفحه از یک سایت وجود داشته باشد که به یک صفحه تنها روی یک سایت دیگر لینک شده است، به هر یال یک وزن معتبر $\frac{1}{k}$ نسبت می‌دهند و اگر یک صفحه تنها از اولین سایت به l صفحه روی سایت دیگر اشاره کند، وزن قطبیت $\frac{1}{l}$ را نسبت می‌دهند. برای مقابله در برابر انحراف موضوع، از گسترش پرس و جو به وسیله گرفتن k کلمه بالایی تکرار شونده از هر قسمت ابتدایی صفحه بازیابی شده و هرس کردن مجموعه صفحات کاندید، با در نظر گرفتن ارتباط صفحات به عنوان یک فاکتور در محاسبات HITS استفاده شده است [۷۶].

روش دیگر، روشی برای محاسبه میزان اعتبار است که قسمت بردار مشخصه الگوریتم HITS را به روش زیر اصلاح می‌کند. به جای محاسبه یک بردار مشخصه اصلی AA^{T}، تمام بردارهای مشخصه ماتریس محاسبه می‌شوند و سپس بردار مشخصه روی مجموعه ریشه در نظر گرفته می‌شود (مجموعه‌ای از صفحاتی که در آغاز توسط کلمات کلیدی موتور جستجو بازیابی شده‌اند، همان‌طوری که در HITS است)، سرانجام نمره اعتبار به عنوان اجزا و مؤلفه‌های مشابه این بردار مشخصه معرفی می‌شوند [۷۷].

الگوریتم دیگر، الگوریتم SALSA است که مفهوم ارتباط محکم[5](TKC) را معرفی می‌کند، که دو قدم زدن تصادفی را برای تخمین اعتبار و نمره قطبیت را برای صفحات در یک زیر گراف ابتدایی بازیابی شده به وسیله جستجوی مبتنی بر کلمه کلیدی، اجرا می‌کند. زیر گراف‌های اصلی و معکوس برای به دست آوردن دو امتیاز مختلف مطرح شده‌اند [۷۸]. ادامه این روند شامل ساختار خوشه‌بندی روی صفحات و الگوهای پیوسته آن‌ها برای پایین آوردن میزان لینک‌های بد است [۷۹]. روش کلیدی شمردن تعداد خوشه‌های اشاره‌گر به یک صفحه به جای تعداد گرههای انفرادی و تک است. در این مورد اعتبار یک صفحه به صورت رابطه‌ی زیر تعریف می‌شود [۴۸]:

$$a_j = \sum_{k:j\in l(k)} \frac{1}{\sum_{i:j\in l(i)} S_{ik}} \qquad (۳-۳)$$

[4] Topic drift
[5] tightly-knit community

که $S_{ik}=\frac{|l(i)\cap l(K)|}{|l(i)\cup l(k)|}$ و l(i) یک مجموعه از صفحات لینک شده از صفحه p_i است.

روش دیگر مفهوم لینک‌های"neponistic" را معرفی می‌کند – لینک‌هایی که به دلایلی بیشتر از حد استحقاق خود قرار می‌گیرند، برای مثال، لینک‌های هادی[۶]وی یک وبسایت یا لینک‌های بین صفحات در یک مزرعه لینک. سپس الگوریتم C4.5 برای تشخیص لینک‌های neponistic با استفاده از ۷۵ ویژگی باینری مختلف مثل IsSimilarHeaders,IsSimilarHost اجرا میشود. در نهایت پیشنهاد هرس کردن یا کاهش وزن لینک‌های neponistic داده می‌شود [۸۰].

۴-۳-۳ ـ الگوریتم‌های مبتنی بر پالایش برچسب‌ها:

ایده پالایش برچسب‌ها در تحقیقات مربوط به یادگیری ماشین برای مسائل طبقه بندی برای مدتی طولانی مورد مطالعه قرار گرفت. در این قسمت الگوریتمی را معرفی می‌کنیم که این ایده را برای کشف هرزنامه وب اجرا می‌کند.یکی ازالگوریتم‌های این گروه به صورت زیر است [۴۴] .

این الگوریتم در دو مرحله کار می‌کند. ابتدا برچسب‌ها با استفاده از یک الگوریتم کشف هرزنامه که در[۸۱] اشاره شده، اختصاص داده می‌شوند. سپس، در مرحله دوم برچسب‌ها به یکی از سه روش زیر پالایش می‌شوند. یک روش انجام خوشه‌بندی گراف وب است. اگر اکثر صفحات موجود در یک خوشه هرزنامه پیش بینی شده باشند، آنگاه تمام صفحات موجود در خوشه را به عنوان هرزنامه مشخص می‌شوند. پیش بینی‌های الگوریتم پایه [۰,۱] هستند، سپس مقدار میانگین خوشه محاسبه و با یک حد آستانه مقایسه می‌شود. همان فرایند برای پیش‌بینی غیرهرزنامه انجام می‌شود.

روش‌های دیگر پالایش برچسب ها مبتنی بر انتشار با قدم زدن تصادفی است. قسمت مهم، مقداردهی اولیه بردار شخصی‌سازی $\vec{r}$ در PPR توسط نرمال‌سازی پیش بینی‌های الگوریتم مبنا است: $r_p = \frac{s(p)}{\sum_{p\in v} s(P)}$، که $s(p)$ یک پیش‌بینی الگوریتم مبنا و $r(p)$ مؤلفه بردار $\vec{r}$ متناظر با صفحه p است [۸۲].

در نهایت، روش سوم استفاده از یادگیری گرافیکی پشته ای[۷]است. این ایده برای گسترش دادن ویژگی‌های اصلی یک شی با ویژگی جدید است که یک پیش بینی میانگین برای صفحات وابسته در گراف است و یک الگوریتم یادگیری ماشین را دوباره اجرا می‌کند و محققین بعد از دو بار یادگیری ۳ درصد بهبود را نتیجه گرفته اند [۸۳].

۴-۳-روش های مبتنی بر محتوا و لینک:

۱-۴-۳- مطالعات مبتنی بر کاهش ویژگی:

تمرکز اصلی مطالعه یاری و همکاران بر کاهش ویژگی ها و سعی در حفظ کارایی کلی شناسایی هرزنامه وب است [۴۶].

یکی از وظایف زمانبر و مهم در سیستم های شناسایی هرزنامه وب، استخراج ویژگی است که با کوشش فراوان و طی فاز شاخص بندی انجام می شود، اگر تعداد کمتری ویژگی در شناسایی هرزنامه استفاده شود هزینه محاسباتی کمتر و بنابراین کارایی سیستم بیشتر خواهد شد.

---

6 navigational 6

6 Stacked graphical learning 7

محققین انواع روش های انتخاب ویژگی را مبتنی بر χ^2، SVM، IG و CFS انجام داده، بعد از اینکه ویژگی ها به وسیله روشهای انتخاب ویژگی انتخاب شدند، تاثیرگذاری آنها به وسیله الگوریتم های طبقه بندی برای ویژگی های انتخاب شده در برابر کل ویژگی ها مورد بررسی قرار گرفت.

اصل کار استفاده از تعداد کمتری ویژگی برای حصول به سطح عملکرد بالاتر است، براساس نتایج آزمایشات حاصله ویژگی هایی که با روش همبستگی[8]انتخاب می شوند، دارای تاثیر بیشتری در شناسایی هرزنامه وب هستند، در حالیکه بعد از به کارگیری LADTree مشخص شد تعدادی از ویژگی ها در شناسایی بسیار تاثیرگذارتر و متمایزتر هستند.

نهایتاً از ۹ ویژگی(HST-9,HST-17,AVG-53,AVG-55,AVG-64,AVG-66,STD-95,Neighbors-2-mp,outdegree-mp) به عنوان ویژگی های نهایی سیستم شناسایی هرزنامه استفاده نموده اند. لازم به ذکر است مبتنی بر نتایج حاصله، سهم این ۹ ویژگی تاثیرات جدی در شناسایی صفحات هرزنامه از غیرهرزنامه دارد.

در این مطالعه سپس به طبقه بندی با الگوریتم های متفاوت پرداخته است. نتایج این الگوریتم ها برای مقایسه تاثیر روشهای انتخاب ویژگی متفاوت کاربرد دارد.

برای وظایف طبقه بندی از الگوریتم های نظیر شبکه های عصبی، SVM، NaÏve Bayes، درخت تصمیم استفاده شده است. از آنجا که هدف از این تحقیق بیشینه کردن کارایی است، الگوریتمی انتخاب شده که کارایی را بیشینه کند. الگوریتم LADTree که نسبت به بقیه الگوریتم ها بهترین نتیجه را می دهد، بکار گرفته شده است.

به منظور ارزیابی الگوریتم های تشخیص هرزنامه، از مجموعه داده ای UK2007 استفاده شده و برای جلوگیری از Overfitting و اطمینان از صحت ارزیابی نهایی، ارزیابی متقاطع 10-fold روی داده های آموزشی و تست استفاده شده و مجموعه آموزشی T از تعدادی زیادی سند هرزنامه و غیرهرزنامه تشکیل شده است و هر سند با تعدادی ویژگی ارائه می شود و برای هر سند یک درجه هرزنامه گی[69] مشخص می شود.

از آنجا که ویژگی های ترکیبی به خوبی عمل می کنند ، آنها را با طبقه بندی کننده های متفاوت به کار برده و سپس تعدادی از آنها که دقت بالاتری داشته و نسبت به بقیه بهتر هستند اشاره شده است.

نتایج حاصل از آزمایشات با در نظر گرفتن ۱۴۰ ویژگی اولیه نشان می دهد که حداقل تعداد ویژگی ها با روش CFS انتخاب شدند. دقت بالا با Random Forest (با همه ۱۴۰ ویژگی) بدست آمد و با LADTree با ۲۶ ویژگی. دقت مشابه را در هر دو الگوریتم ببینیم اما در LADTree تعداد ویژگی ها کمتر است.

در عمل کاهش فضای ویژگی ها حتی در صورت از دست دادن اندک دقت امری ضروری است، بنابراین مدل ساخته شده با الگوریتم LADtree با ۲۶ ویژگی به مدل ساخته شده با Random Forest با همه ویژگی ها ارجحیت دارد .

با توجه به متریک F-measure نیز برای مدل SVM در زمان استفاده از روش انتخاب SVM کاهش ویژگی ها قابل توجه بوده (۹۳ ویژگی و اندازه ۷۰/۳ %) و همچنین کاهش ویژگی ها در مدل LADtree با روش انتخاب CFS قابل توجه است (۲۶ ویژگی و ۷۰/۷ %) و سرانجام الگوریتم

---

6 Correlation
6 Spamcity
8
9

LADtree انتخاب شده و از آنجا که درخت تصمیم می تواند ویژگی های مفید را در طی ساخت درخت ارائه کند ویژگی های نهایی مبتنی بر مدل LADtree انتخاب شده است.

و نهایتاً نتایج با استفاده از این ۱۰ ویژگی انتخابی و متریک ROC ارائه شده است.

جدول۳-۴: نتایج بدست آمده با ۱۰ ویژگی با اعمال الگوریتم های کاهش [۴۶]

ویژگی ها	LADtree	Neutral Network	SVM	Naïve Bayes	Random Froset
۱۰	۷۶/۸%	۶۸/۹%	۶۳/۷%	۷۲%	۷۶%

در نهایت تکنیک های متفاوتی برای بهبود طبقه بندی کننده انتخابی مورد آزمایش قرار گرفت از جمله bagging و boosting.

در این مورد از boosting استفاده شده و ترکیب طبقه بندی کننده ها نتیجه بهتری را بدست داده است.

جدول ۳-۵: نتایج بدست آمده با ۱۰ ویژگی با استفاده از boosting [۴۶]

ویژگی ها	دقت	بازیابی	F-measure	ROC
۱۰	۷۰/۳%	۶۹/۷%	۶۹/۵%	۷۷/۵%

۴-۳-۲- مطالعات مبتنی بر ترکیب طبقه بندی کننده ها:

در زمینه ترکیب طبقه بندی کننده ها برای کشف هرزنامه وب، انتخاب گروه[۷۰] هنوز کاربردی نشده است. توانایی ترکیب شمار زیادی از طبقه بندی کننده ها که از تناسب بیش از حد[۷۱] جلوگیری کند، انتخاب گروه را یک گزینه ایده آل برای طبقه بندی هرزنامه وب می کند. به این خاطر که به ما اجازه استفاده از شمار بزرگتری از ویژگیها را می دهد و جنبه های متفاوتی از داده های آموزشی در همان زمان آموزش داده می شود (یادگیری).

به جای تنظیم پارامترهای طبقه بندی کننده متفاوت، ما می توانیم روی پیدا کردن ویژگی های قوی و مدل های طبقه بندی اصلی که به اعتقاد ما قادر به ثبت تفاوت مابین کلاسهای تشخیص داده شده هستند تمرکز کنیم.

Erdélyi و همکاران برای پیاده سازی انتخاب گروه وکا را برای اجرای آزمایشات استفاده کرده اند. وکا از استراتژیهای اثبات شده برای جلوگیری از تناسب بیش ازحد نظیر bagging، جایگزینی با انتخاب، sort initialization، ارزیابی متقاطع پشتیبانی می کند. از ۵-fold در طی آموزش و ساختن گروه ها استفاده شده، AUC را به عنوان متریک هدف برای بهینه سازی تنظیم کرده و ۱۰۰ تکرار را برای الگوریتم تپه نوردی[۷۲] اجرا کرده است.

در این مطالعه از دو مجموعه داده Web Spam-UK2007 و DC2010 ایجاد شده برای Discovery ECML/PKDD Challenge 2010 روی کیفیت وب استفاده شده است.

[۷۰] Ensemble Selection
[۷۱] Overfitting
[۷۲] hillclimbing

Erdélyi و همکاران از انواع مدلهای زیر برای ساختن کتابخانه مدل برای انتخاب گروه استفاده کرده اند:

Random Forest ،Naïve bayes ،logistic regression ،Boosted decision tree ،Bagged decision tree

برای اغلب کلاس های ویژگیها همه ی طبقه بندی کننده ها بکار برده شده و اجازه انتخاب بهترین آنها را داده شده است.

برای ECML/PKDD Discovery، از سودمندی تجمعی نزولی نرمالیزه[۷۳](NDCG) برای ارزیابی استفاده شده است. معیار DCG با استفاده از یک مقیاس رتبه بندی شده ارتباط اسناد در مجموعه نتایج موتورهای جستجو، سودمندی یک سند را مبتنی بر مکان آن در فهرست نتایج موتور جستجو می دهد. سودمندی از بالای فهرست به پایین فهرست با کاهش رتبه تنزل پیدا می کند.

سودمندی تجمعی[۷۴]مکان نتیجه را مورد توجه قرار نمی دهد و در واقع مجموع مقادیر ارتباط رتبه بندی شده می باشد و در یک مکان با رتبه p به صورت زیر تعریف می شود [۴۸]:

$$CG_P=\sum_{i=1}^{p} rel_i \qquad (۳-٤)$$

توسط تابع سودمندی تجمعی با تغییرات در ترتیب نتایج جستجو تاثیرپذیر نیست بنابراین DCG معرفی شده است.

DCG بیان میکند که اسناد با ارتباط بالا که در فهرست موتورهای جستجو در مکان پایین تر آشکار می شوند باید جریمه شوند و مقدار ارتباط رتبه بندی شده باید متناسب با مکان آنها به صورت لگاریتمی کاهش یابد. DCG در یک مکان با رتبه p به صورت زیر محاسبه می شود [۴۸]:

$$DCGp=\sum_{i=1}^{p} \frac{2^{rel_i}-1}{\log_2 i+1} \qquad (۳-٥)$$

مقایسه کارایی موتورجستجو از یک پرسش تا پرسش دیگر نمی تواند تنها با DCG بدست آورده شود، بنابراین سودمندی تجمعی در هر مکان برای یک مقدار p انتخابی باید در طی پرسش ها نرمال سازی شود و این کار با مرتب سازی اسناد یک فهرست نتیجه به وسیله ارتباط، تولید بیشینه احتمال DCG تا مکان p، همچنین DCG ایده آل تا آن مکان انجام می گیرد. برای یک پرسش NDCG به صورت زیر محاسبه می شود[۴۸]:

$$nDCGp=\frac{DCGp}{IDCGp} \qquad (۳-٦)$$

7 Normalized discounted cumulative Gain
7 CG 4

در اینجا برای تاکید کارایی روی فهرست کامل ، تابع کاهشی از تعریف عادی به خطی تغییر یافته:

1-i/N

که در آن N سایز زیر مجموعه test است. برای توجیه عملکرد تابع کاهش، در نظر داشته باشید که یک آرشیو اینترنتی که ممکن است ۵۰ درصد و یا بیشتر دانه های میزبان شناسایی شده خزش شوند و هرزنامه ۱۰ تا ۲۰ درصد از همه میزبانها را تشکیل می دهد. فرمول نهایی ارزیابی [۴۸]:

$$NDCG=\frac{DCG}{Ideal\ DCG}$$

$$DCG=\sum_{rank=1}^{N} utility(rank).(\frac{rank}{N}) \qquad (۳-۷)$$

هدف ما کشف کارایی مجموعه ویژگیهای قابل محاسبه کم هزینه است و تشکیل مجموعه ویژگیها در زیر توضیح داده خواهد شد.

گروه فقط محتوا: سه گروه مختلف را روی ویژگی های فقط محتوا به منظور ارزیابی کارایی با حذف کامل اطلاعات لینک ساخته شده است. مجموعه ویژگی ها برای این گروه ها در زیر آمده اند.

(A): محتوای عمومی [۴۴,۴۹] ویژگیها بدون هر اطلاعات لینک. ویژگیها برای صفحه با رتبه صفحه بیشینه در میزبان برای صرفه جویی در محاسبات رتبه صفحه، استفاده نمی شود. دقت پیکره، کسری از کلمات در صفحه که فرکانس corpuswise و فراخوانی پیکره است، کسری از فرکانس اصطلاحات صفحه که زمانی استفاده می شوند که آنها نیاز به اطلاعات کلی از پیکره دارند.

(Aa): مجموعه کوچکی از ویژگی ها از ۲۴ ویژگی (A)، دقت پرسش و فراخوانی پرس و جو مشابه با دقت پیکره تعریف می شود. فراخوانی اما مبتنی بر اصطلاحات محبوب از یک لاگ فایل پرس و جو به جای پیکره کامل است. یک مجموعه ویژگی های قوی مبتنی بر شهود که هرزنامه نویسان اصطلاحاتی که پرسش های محبوب را آرایش می دهند استفاده می کنند.

(B): مجموعه محتوای عمومی کامل در برگیرنده ویژگیها برای صفحه با بیشینه رتبه صفحه میزبان

مجموعه ویژگی های +B: یک نماینده از کلمات مشتق شده از اصطلاحات BM25 طرح توزین اصطلاحات BM25 [۴۴,۸۴].

مقایسه کارایی گروه های ساخته شده با مجموعه ویژگی های بالا نشان می دهد که با تعداد ۱۰۰۹۶ ویژگی BM25 و B برای مجموعه داده UK2007 و مجموعه داده DC2010، متریک AUC به ترتیب مقادیر ۰/۸۹۳ و ۰/۸۹۱ بدست آمده و برای DC2010 متریک NDCG، مقدار ۰/۸۹۳ بدست آمده است که بهترین مقادیر هستند.

در کمال تعجب با مجموعه Aa با تعداد کمی ویژگی(۲۴ ویژگی) کارایی تنها ۱ درصد بدتر شده است (AUC=۰/۸۴۱)، با استفاده همه ویژگی های مبتنی بر محتوای موجود بدون اطلاعات پیوند، کارایی مشابه آنچه بهترین گزارشات روی مجموعه داده های مورد آزمایش تا کنون ارائه داده اند بدست آمده است (۱۰۲۷۳ ویژگی و برای UK2007 متریک AUC=۰/۹۰۲).

در این آزمایشات مشاهده شد که ویژگی های مبتنی بر لینک کارایی را افزایش نمی دهند.

با تصویرسازی روی ۱۰۰۰۰۰ میزبان Web Spam UK2007 و ۱۹۰۰۰۰ میزبان مجموعه داده ای DC2010، مبادله بین تولید ویژگی و دقت طبقه بندی هرزنامه را مورد مطالعه قرار گرفت و مشاهده شد که ویژگی های بیشتر کارایی بیشتری را باعث می شوند، هر چند که ویژگی های مبتنی بر لینک فقط کارایی حاشیه ای را بدست می دهند وتکنیک های یادگیری ماشین بهتر از خلق ویژگی های جدید پیچیده است. مولفین موفق به کامپایل یک مجموعه ویژگی های حداقل شده که می تواند به سرعت مورد محاسبه قرار گیرد تا رهگیری هرزنامه و زمان خزش مبتنی بر یک نمونه از یک وب سایت جدید را انجام دهد [۴۸].

۳-۴-۳- مطالعات مبتنی بر تست اهمیت ویژگی های متفاوت در تشخیص هرزنامه:

در مطالعه ای دیگر Egele و همکاران آزمایش های جامعی برای درک اثرات ویژگی های متفاوت در رتبه بندی موتورهای جستجو انجام داده اند. آنها سیستمی را توسعه داده اند که باعث کاهش ورودی های اسپم از نتایج موتورهای جستجو به وسیله پس پردازش آنها می شود.

انتخاب ویژگی:

نخست آنها به انتخاب ویژگی ها پرداخته اند. مهندسی معکوس برای تعیین ویژگی های متناسب برای رتبه بندی به کار برده شده است. برای پی بردن به اهمیت ویژگی، تست جعبه سیاه را روی موتورهای جستجو اجرا کرده اند. به طور دقیق تر مجموعه ای امتحانی متفاوت با ترکیبات متفاوت ویژگی ها را خلق نموده و رتبه آنها مشاهده نموده اند.

براساس گزارش از کمپانی های بهینه سازی موتورهای جستجو و مطالعه کارهای مرتبط ده ویژگی مهم صفحات را انتخاب کرده اند(کلمات کلیدی در برچسب عنوان، کلمات کلیدی در بدنه، کلمات کلیدی در برچسب H_1، لینک های خروجی به سایتها با کیفیت بالا، لینک های خروجی به سایتها با کیفیت پایین، تعداد لینک های ورودی، متن لنگر لینک های ورودی شامل کلمات کلیدی، مقدار متنهای شاخص پذیر، کلمات کلیدی در مسیر فایل URL، کلمات کلیدی در نام دامنه) [۸۵,۸۷,۸۶].

با توجه به ویژگی ها، نخست مکان های متفاوت روی صفحه که یک عبارت جستجو می تواند ذخیره شود مورد بررسی قرار داده و ویژگی های مبتنی بر محتوا نظیر برچسب های بدنه، عنوان و سرفصل مورد بررسی قرار گرفته اند که شاخص خوبی برای اطلاعاتی که می توان بر روی آن صفحه یافت، می باشد. به علاوه ویژگی های مبتنی بر لینک نیز بیان شده اند. معمولاً تعداد لینک های ورودی به یک صفحه نمی تواند به طور مستقیم تاثیرگذار باشد (برای مثال ویژگی in–link). به همراه این ویژگی ها که مستقیماً با محتوای صفحه مرتبط نیست (نظیر کلمات کلیدی در نام دامنه) گستره وسیعی از ویژگی ها پوشش داده شده که برای محاسبه رتبه بندی کاربرد دارند.

آماده سازی صفحات تست:

هنگامی که ویژگی ها انتخاب شدند، گام بعدی ایجاد مجموعه ای بزرگی از صفحات تست با مقادیر متفاوت و ترکیب های مختلف از این ویژگی ها می باشد. "gerridae plasmatron" به عنوان عبارت کلیدی برای بهینه سازی صفحات انتخاب شده است. هدف تخمین تاثیر ویژگی های صفحه در الگوریتم های رتبه یابی است و تعیین اینکه آیا صفحات آزمایشی بهتر از سایتهای قانونی هستند.

بنابراین صفحات قانونی و هرزنامه را به صورت دستی امتحان کرده و میانگین، فرکانس های تجربی را استخراج نموده اند. برای نمونه فرکانس کلمات کلیدی در بدنه متن تا ۱% به عنوان پایه در نظر گرفته شده و ۴% به عنوان بالا و ۱۰% به عنوان هرزنامه در نظر گرفته شده است.

یک زیرمجموعه ۳۰ تایی از ترکیب ویژگی ها را انتخاب کرده، هر شکل ترکیب یک گروه آزمایشی است که دربرگیرنده سه نمونه یکسان است که مقادیر ویژگیهای همانند را به اشتراک می گذارند.

برای این ۳۰ گروه مقادیر ویژگی ها به طریقی انتخاب شده که موارد عادی، متفاوت را بیان کند. مورد عادی یک سایت قانونی هست که به وسیله صفحه مرجع بیان می گردد. برای این صفحه مقادیر ویژگی ها به کلاس نرمال تعلق دارد. موارد دیگر کلمات کلیدی را در مکانهای مختلف صفحه در بر می گیرد (به عنوان مثال، بدنه، عنوان، سرفصل) یا مقادیر متفاوت از لینکهای ورودی یا خروجی.

با استفاده از این عبارت جستجو صفحات آماده شده و در پایان یک صفحه مرجع را دربرگیرنده اطلاعات درباره gerridae وplasmatron گردآوری شده از منابع مختلف است، خلق شده است.

در مرحله دوم این صفحه مرجع ۹۰ بار کپی شد. برای فرار از تشخیص های تکراری توسط موتورهای جستجو، در هر کدام از این ۹۰ صفحه، بسیاری از کلمات با شیوه ای شبیه [۸۸] جایگزین شدند.

نتایج :

هنگامی که ۳۰ گروه آزمایشی ایجاد شدند، آنها در ۹۰ دامنه ثبت شده قرار گرفتند که توسط ۴ ارائه دهنده خدمات میزبانی سرویس می گرفتند. به علاوه تعدادی دامنه ها در بخش وب سرور دپارتمان ما مستقر شدند.

هنگامی که سایتها مستقر شدند، اقدام به ثبت تصاویر لحظه ای[5]از نتایج موتور جستجو برای پرسش "plasmatron gerridae" شد. برای ۲۳۱۲ پرسش ارائه شده به گوگل و ۱۷۰۰ پرسش ارائه شده به یاهو، مشاهده شد که رتبه بندی نمی تواند در طول مدت طولانی پایدار بماند. در حقیقت ، طولانی ترین دوره برای یک رتبه بندی پایدار برای صفحات آزمایشی تنها ۶۸ ساعت برای گوگل و ۱۴۳ ساعت برای یاهو بود. گوگل صفحاتی که مسیر (URL) آنها شامل بیش از ۵ دایرکتوری هست شاخص سازی نمی کند. تعدادی از صفحات آزمایشی در هفته های اول از شاخص حذف شدند. برای گوگل جستجو برای"gerridae" نزدیک به ۵۵۰۰۰ نتیجه در برداشت. صفحات تست ما ۵ تا از ده اسلات بالای رتبه بندی را اشغال کردند. ۶ بالاترین مکان مشاهده شده برای پرسش"plasmatron" بود. برای یاهو برای هر دو کلمه کلیدی صفحات تست ما در مکان ۱ قرار گرفتند و برای دو هفته دارای همین رتبه بندی بودند .

بدلیل رتبه بندی متفاوت، موقعیت یک صفحه با میانگین گیری موقعیت آن در طی شش هفته تعیین شد. به این دلیل شش هفته تعیین شد که فاز ابتدایی آزمایش اشتباهاتی به دلیل کشف تکراری در بر می گرفت. هم چنین گنجانده شدن صفحات در شاخص زمان می گرفت. مشاهده شد که وقتی یک پرسش مشابه به گوگل یا یاهو داده میشود ، رتبه بندی های مختلفی تولید می کنند. این نشان می دهد که الگوریتم های به کار رفته متفاوت است. بنابراین وزن های ویژگی متفاوت برای گوگل ، یاهو استخراج شده است.

[7] Snapshot [5]

با دانستن ترکیبات مقادیر همه ویژگی ها برای یک صفحه k و مشاهده مکان آنها pos(k) در رتبه بندی، هدف تعیین کردن یک وزن بهینه برای هر ویژگی است که به بهترین وجه اهمیت ویژگی را برای الگوریتم رتبه بندی ثبت و ضبط کند. به عنوان اولین قدم، یک تابع نمره تعیین شده که این تابع به عنوان ورودی مجموعه ای از وزن ها و مقادیر ویژگی دریافت می دارد و یک نمره score(k) را برای صفحه page(k) محاسبه می کند [۸۹].

$$Score(k)=\sum_{i=1}^{n} f_i^k . w_i$$

N تعداد ویژگی ها ، w_i وزن ویژگی i و$w_i \epsilon$ [-1,1] ، f_i^k وجود ویژگی i در صفحه تست k می باشد.

این محاسبات برای همه صفحات تست تکرار می شود البته با وزن های مشابه. وقتی که همه نمرات محاسبه شدند، مجموعه صفحات تست بر طبق نمره شان مرتب شدند و این اجازه می دهد که یک رتبه بندی پیش بینی شده rank(k) به هر صفحه اختصاص داده شود. تفاوت مابین رتبه بندی پیش بینی شده و جایگاه واقعی برای همه صفحات محاسبه می شود. وقتی مجموع این تفاوت ها کمینه شود ، وزن ها بهینه هستند. این به یک تابع هدف مسئله برنامه ریزی خطی (LP) تبدیل می شود [۸۹].

$$\min: \sum_{k=1}^{m} \alpha_k |pos(k)-rank(k)|$$

فاکتور $\alpha(k) = m-pos(k)$ را به LP اضافه کرده، که اجازه می دهد صفحات تست با رتبه بندی بالاتر، تاثیر بیشتری را روی وزن های ویژگی اعمال کنند (m تعداد صفحات تست می باشد). موقعیت دقیق صفحات با رتبه بندی پایین در حال نوسان است. بنابراین باید راهی یافت که این تاثیر تصادفی را روی محاسبه وزن ها کاهش دهد. حل LP با الگوریتم سیمپلکس در وزن ها برای تمام ویژگی ها، فاصله میان مقادیر پیش بینی شده و واقعی را کاهش می دهد.

نهایتاً مشخص شد برای گوگل تعداد عبارات جستجو در عنوان[۶]و بدنه متن[۷]تاثیر مثبت و قوی در رتبه بندی دارد. هم چنین تعداد لینک های خروجی مهم بوده است. از سوی دیگر کلمات کلیدی که جزء مسیر فایل هستند تاثیر کمی در رتبه بندی دارند.

برای یاهو، ویژگی ها کاملاً متفاوت هستند. برای مثال کلمه کلیدی که در عنوان آشکار می شود تاثیر کمتری دارد و حتی با افزایش فرکانس این تاثیر کمتر نیز می شود. یاهو وزن را بیشتر روی تعداد لینک های ورودی و خروجی نسبت به گوگل قرار می دهد. به عبارت دیگر تعداد دفعاتی که کلمه کلیدی در متن آشکار می گردد چندان تاثیر مثبتی ندارد.

با توجه به نتایج گوگل ۷۸ صفحه از ۲۶ گروه آزمایشی در رتبه بندی فهرست شده اند. گروه های آزمایشی از دست رفته ما صفحاتی با سلسله مراتب سطح ۵ هستند و بنابراین توسط موتورهای جستجو شاخص سازی نشده اند. موقعیت برای شش گروه (۲۳%) با فاصله ۲ مورد پیش بینی قرار گرفت و برای یازده گروه(۴۲%) با فاصله ۵ یا کمتر پیش بینی شد. برای یاهو وقتی که گروه های آزمایشی با رتبه بندی مقایسه کردند، ۲۱ گروه در رتبه بندی آشکار شدند. سه تا از این گروه ها (۱۴%) با فاصله ۲ پیش بینی شدندو هشت گروه(۳۸%) با فاصله ۵ یا کمتر پیش بینی شدند .

در نگاه اول پیش بینی های چندان دقیق به نظر نمی رسد ، با این حال برای یاهو پیش بینی ها نزدیک به نتایج واقعی هستند .هر چند پیش بینی دقیق نیست اما روند کلی را مشخص می کند .

[7] Title [6]
[7] Body [7]

می توان نتیجه گرفت که ارزیابی کلی از اهمیت یک ویژگی درست است، اگرچه که مقادیر وزنی دقیق ممکن است متفاوت باشد. در اینجا تنها یک تابع رتبه بندی خطی در نظر گرفته شده در حالی که الگوریتم های رتبه بندی پیچیده تر می باشند.

ساخت طبقه بندی کننده:

رویکرد تعیین یک طبقه بندی کننده است که صفحات هرزنامه را از غیرهرزنامه مطابق با این ویژگی ها تشخیص دهد.

طبقه بندی ارائه شده برای موتور جستجوی گوگل توسعه داده شده است. بنابراین آن دسته از ویژگی ها که مربوط به گوگل هستند در نظر گرفته شده است. اینها تعداد کلمات کلیدی در متن، بدنه و نام دامنه هستند. به علاوه اطلاعات لینک نیز مورد توجه قرار گرفته است. تعداد لینک های خروجی بی اهمیت است و تعداد لینکهای ورودی نیز بی آسانی قابل تعیین نیست. اطلاعات لینک های ورودی که به یک صفحه اشاره می کنند توسط موتورهای جستجو موجود نیست. و به این علت ویژگیهای مربوطه را با لینک کمکی: پرسش ها تخمین زده شده است. گوگل و یاهو پرسش ها را درشکل لینک http://www.example.com پشتیبانی می کنند که یک فهرست از صفحات که به این سایت لینک می شوند را نتیجه می دهد. مشکل اینجاست که نه یاهو و نه گوگل همه صفحاتی که به صفحه پرسش پیوند می خورند را شامل نمی شود. بنابراین این تعداد تنها تقریبی از تعداد واقعی لینک ها که به یک سایت اشاره می کنند، هستند.

برای ساخت طبقه بندی کننده صفحات وب نیاز به یک مجموعه آموزشی برچسب گذاری شده است و همچنین مجموعه دیگری از داده ها به منظور بررسی مدل حاصل و ارزیابی کارآیی آن مورد نیاز است. برای ایجاد این مجموعه ها، ۱۲ پرسش به موتور جستجوی گوگل ارائه شد (درخواست برای عبارات جستجو شده رایج استخراج شده از فهرست پرسش های رایج گوگل). برای هر پرسش ۵۰ نتیجه اول به صورت دستی به عنوان قانونی/ هرزنامه طبقه بندی شدند. با دست کشیدن از صفحات غیر HTML (برای مثال pdf، ppt) یک مجموعه داده آموزشی شامل ۲۵۹ سایت به عنوان نتیجه حاصل شد (۱۹۴ تا قانونی و ۱۰۱ هرزنامه). مجموعه داده آزمایشی برای این مطالعه ۲۵۲ صفحه دارد (۱۹۳ تا قانونی و ۵۹ تا هرزنامه).

همه صفحات نتیجه دانلود شده و کدهای منبع HTML تجزیه می شود و مقادیر ویژگی برگردانده می شود. اگر پرسش شامل اصطلاحات چندگانه شود، استخراجگر ویژگی مستقل، در صورتی که پرسش به صورت کامل با ویژگی آنالیز شده مطابقت داشته باشد مقادیر بالاتری را گزارش می کند. منطق پشت این موضوع این است که یک برچسب Heading واحد که پرسش کامل را در بر می گیرد تطابق بهتری را نسبت به برچسب های Heading چند گانه که هر کدام دربرگیرنده یک قسمت از پرسش هستند نشان می دهد. استخراجگر ویژگی که از این رویکرد پیروی می کند، در فهرست زیر با (X) نشان داده شده که تمام ویژگی هایی مورد نظر را شمارش می کند.

عنوان: تعداد اصطلاحات پرس و جو از برچسب عنوان HTML

برچسب H_1: تعداد اصطلاحات پرس و جو در برچسب H_1 HTML

بدنه: تعداد اصلاحات پرس و جو در قسمت بدنه HTML

نام دامنه: تعداد اصطلاحات پرس و جو در نام دامنه
(برای مثال http://www.gerridae-plasmatron.com/index.php)

مسیر فایل: تعداد اصطلاحات پرس و جو در مسیر URL، برای مثال

(http://www.example.org/gerridae-plasmatron/index.php)

لینک های خروجی: تعداد کلی لینک های خروجی

کلمات کلیدی لینک های خروجی: تعداد لینک های خروجی که کلمات کلیدی را در بر می گیرند .

لینک های ورودی گوگل: تعداد لینک های ورودی گزارش شده گوگل

لینک های ورودی یاهو: تعداد لینک های ورودی گزارش شده یاهو

نشانه رتبه بندی: مقدار رتبه بندی گوگل برای URL که به وسیله نوار ابزار گوگل گزارش می شود .

دامنه رتبه بندی: مقدار رتبه بندی گوگل برای دامنه نیز به وسیله نوار ابزار گوگل گزارش می شود.

تعداد کلمه: تعداد کلی کلمات در متن

T فرکانس: فرکانس اصطلاحات پرس و جو آشکار شده در متن (تعداد کلمات پرس و جو / تعداد کلمات روی صفحه)

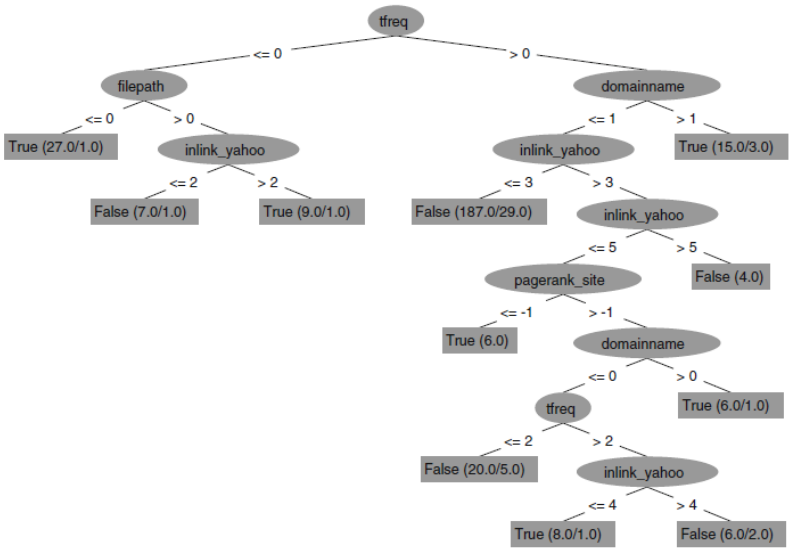
با استفاده از مجموعه آموزشی برچسب گذاری شده به عنوان پایه، یکسری الگوریتم ها برای آموزش مدل های مختلف طبقه بندی اجرا شد. در پایان از ابزار weka استفاده شد که از بسیاری مدل های کلاس بندی پشتیبانی می کند.

ارزیابی مدلهای طبقه بندی:

هشت مدل طبقه بندی متفاوت از ابزار وکا برای ارزیابی قابلیت اجرای هدف مورد بررسی قرار گرفته است.

نهایتاً یک درخت تصمیم گیری به عنوان طبقه بندی کننده انتخاب شده زیرا اهمیت ویژگی ها را تعیین می کند، ویژگیهای نزدیک به ریشه مهم تر هستند. اجرای j48 موجود در بسته وکا، احتمالات مختلفی را برای نتیجه نهایی ارائه می دهد. جالب ترین فاکتور، فاکتور اطمینان است که درجه هرس کردن را نشان می دهد و بنابراین دقت کلاس بندی را نشان می دهد.

مقدار ۰/۱ منجر به بهترین نتیجه برای مجموعه داده مورد آزمایش می شود. این درخت شامل ۲۱ گره می شود که ۱۱ تای آنها برگ هستند. ۵ ویژگی به وسیله الگوریتم انتخاب شده اند که می توانند به عنوان معیارهای تمایز بین سایت های هرزنامه و غیر هرزنامه مفید باشند. علاوه بر این وکا یک فاکتور اطمینان را برای هر برگ محاسبه می کند. مهم ترین ویژگی مربوط به وجود اصطلاحات مورد جستجو در صفحه است. ویژگیهای مهم دیگر نام دامنه، مسیر فایل، لینک های ورودی گزارش شده یاهو و ارزش رتبه بندی گزارش شده به وسیله نوار ابزار گوگل می باشند.



شکل ۳-۱:درخت j48 تولید شده توسط وکا [۸۹]

این درخت تصمیم برای ارزیابی داده های تست مورد استفاده قرار گرفت و ماتریس زیر را حاصل شد. طبقه بندی کننده نرخ مثبت غلط ۱۰/۸ % و نرخ منفی غلط ۶۴/۴ % را ارائه می دهد. همچنین نرخ تشخیص (مثبت درست) ۳۵/۶ درصد بدست آمده است.

جدول ۳-۶: نتایج حاصل از ارزیابی درخت j48 بر روی داده های تست [۸۹]

	طبقه بندی شده به عنوان هرزنامه	طبقه بندی شده به عنوان قانونی
هرزنامه	۲۱	۳۸
قانونی	۲۰	۱۷۳

تشخیص ۳۵ درصدی برای سیستم پیشنهادی مناسب می باشد اما نرخ مثبت غلط ۱۱ درصدی نامناسب می باشد. به منظور کاستن نرخ مثبت غلط، تصمیم گرفته شد فاکتور اطمینان برگ ها دخالت داده شود. با استفاده از این فاکتور اطمینان به عنوان آستانه، می توان سیستم را به نحوی تنظیم کرد که نرخ مثبت غلط ها کاهش یابد. به عنوان مثال با اطمینان ۰/۸۸، طبقه بندی کننده نرخ منفی غلط ۸۱/۴ درصد را ارائه می دهد و هیچ مثبت غلط ی را برای مجموعه داده تست مورد آزمایش نمی دهد و همچنین سیستم با این مقدار آستانه نرخ مثبت درست ۱۸/۴ درصد را ارائه می دهد.

در حالی که نرخ ۱۸ درصد کامل نیست و نیاز به بهبود دارد، اما سایتهای ناخواسته را در نتایج کم می کند. با توجه به اینکه اغلب کاربران ۱۰ یا ۲۰ نتیجه ی بالای موتور جستجو توجه می کنند، این ۱۸ درصد سبب ایجاد دو اسلات خالی در ۱۰ نتیجه بالای موتورهای جستجو می شود که بالقوه می تواند با صفحات جالب جایگزین شود [۸۹].

۴-۴-۳- مطالعات مبتنی بر پیکربندی[7] وب[8]:

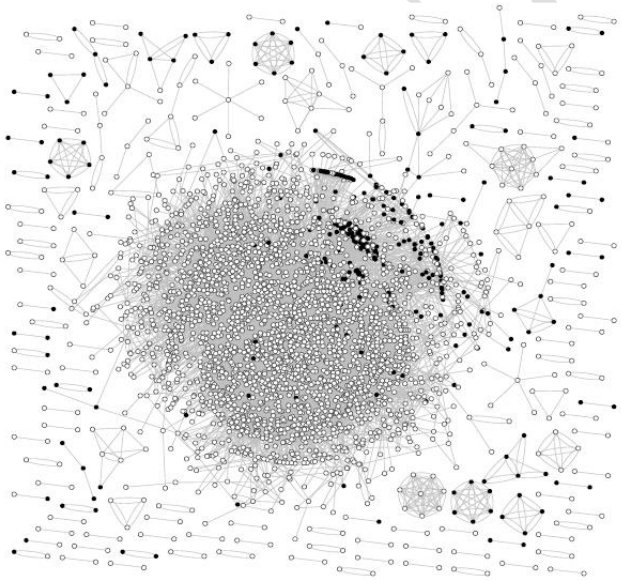
از پیکربندی گراف وب با استخراج وابسبرچسبی های لینک مابین صفحات استفاده می شود. آنها متوجه شده اند که میزبانهای لینک شده هر دو تمایل به کلاس های مشابه دارند : هر دو هرزنامه هستند یا هر دو غیرهرزنامه.

سه روش از پیکربندی های گراف وب استفاده شده در طبقه بندی کننده پایه:

i) خوشه بندی گراف میزبان و تخصیص برچسب همه ی میزبان ها در خوشه با رای حداکثر ii) انتشار برچسب های پیش بینی شده به همسایگان iii) استفاده از برچسب های پیش بینی شده میزبان های همسایه به عنوان ویژگی های جدید و بازآموزی طبقه بندی کننده .

در مورد وب وابستگی ما بین صفحات و میزبان ها وجود دارد. هرزنامه ها تمایل به خوشه شدن بر روی وب دارند. یک توضیح برای این رفتار این است که صفحات هرزنامه اغلب تکنیک های rank-hosting مبتنی بر لینک را اتخاذ کرده اند نظیر link-farming.

طرح ۱ تصویری از گراف میزبان را در مجموعه هرزنامه وب می دهد که مورد استفاده محققین بوده است. یک لبه بین دو میزبان زمانی نشان داده شده است که حداقل ۱۰۰ لینک بین دو میزبان وجود داشته باشد . در طرح، گرههای سیاه هرزنامه هستند و گرههای سفید غیرهرزنامه هستند.



شکل ۳-۲: طرح گراف میزبان [۴۴]

- استفاده از طبقه بندی کننده حساس به هزینه برای استخراج کردن برچسب های غیرمتقارن ذاتی
- بهبودهایی در دقت طبقه بندی با استفاده از وابستگی های برچسب های همسایگان میزبان در گراف وب.
- ترکیب این وابستگی ها به وسیله ی خوشه بندی و پیاده روی تصادفی

[7] Topology [8]

- استفاده از stacked graphical learning برای بهبود دقت طبقه بندی [۹۰].

از مجموعه داده Web Spam UK2006 استفاده شده است. برای آنالیز محتوا، خلاصه ای از محتویات هر میزبان به وسیله ردیابی ۴۰۰ صفحه ی اول قابل دسترس توسط جستجوی اول پهنا بدست آمد. نمونه خلاصه شده ۳/۳ میلیون صفحه را در برمی گیرد.

چارچوب کاری :

پایه واساس سیستم کشف هرزنامه، درخت تصمیم گیری حساس به هزینه است. برای یادگیری درخت از یک روش ترکیبی مبتنی بر لینک و نیز محتوا به منظور کشف انواع متفاوت هرزنامه وب استفاده کرده ایم. همه ی پیش بینی های گزارش شده با استفاده از ارزیابی متقاطع 10-fold محاسبه شده اند .

ویژگی ها:

برای ویژگی های مبتنی بر لینک از Becchetti و همکارن پیروی شده و برای ویژگی های مبتنی بر محتوا از Ntoulas و همکاران [۴۹, ۸۱].

اندازه های مرتبط با درجه:

تعدادی از اندازه های مرتبط با درجه داخلی و خارجی را روی میزبان ها و همسایگان آنها محاسبه شده، به علاوه اندازه های دیگر را نیز مورد توجه قرار گرفته، نظیر لبه متقابل و تعداد پیوندهایی که متقابل (دو جانبه هستند)، assortativity (نسبت درجه یک صفحه ویژه و درجه میانگین همسایگان). (۲۶ ویژگی)

رتبه صفحه :

اندازه های مختلف مرتبط با رتبه صفحه یک صفحه و رتبه صفحه همسایگان پیوند داخلی محاسبه شده است (۱۱ویژگی).

برآورد پشتیبان[۷۹]ها:

یک راه برای مبارزه با هرزنامه لینک شمردن پشتیبانی‌کننده‌ها است. x، d-supporter برای y است اگر کوتاهترین مسیر از x به y طول d داشته باشد. $N_d(x)$ را مجموعه d-supporter های x است. الگوریتم کلی به این صورت است که در هر تکرار الگوریتم، اگر صفحه y لینکی به صفحه x داشته باشد، بردار بیت صفحه x بصورت xORy بهنگام سازی می‌شود. پس از d تکرار، بردار بیت مرتبط با هر صفحه x اطلاعاتی درباره تعداد پشتیبانی‌کننده‌های x در فاصله d ارائه می‌کند. اگر یک صفحه تعداد بیشتری پشتیبانی‌کننده از دیگری داشته باشند، یک‌های بیشتری در پیکربندی نهایی بردار بیت آن ظاهر می‌شود.

یک اندازه جالب دیگر ، تعداد bottle neck است.

$b_d(x)$ یک صفحه x، که ما به صورت $b_d(x)= \min_{j\le d}\{|N_j(x)|/|N_{j-1}(x)\}$ تعریف شده واین اندازه کمینه نرخ رشد همسایگان x تا یک فاصله معین را نشان می دهد. صفحات هرزنامه خوشه هایی را تشکیل دهند که از مابقی گراف جدا هستند و آنها تعداد bottleneck کمتری نسبت به صفحات غیرهرزنامه دارند.

[7] Supporter [9]

ویژگی های مبتنی بر محتوا:

تعداد کلمات در صفحه، تعداد کلمات در عنوان، میانگین طول کلمات، کسری از تعداد کلمات در متن لنگر به کل کلمات در صفحه، کسری از متن قابل رویت، نرخ فشردگی، دقت و فراخوانی پیکره که k تا از پرتکرارترین کلمات در مجموعه داده یافته شده است، به استثنای stopword ها. دقت پیکره[80] بخشی از کلمات در یک صفحه نامیده می‌شود که در مجموعه‌ی کلمات محبوب وجود دارند. فراخوانی پیکره[81] بخشی از کلمات محبوب تعریف شده که در صفحه ظاهر شده است. برای هر دوی دقت پیکره و فراخوانی پیکره ٤ ویژگی استخراج شده است، برای k =100, 200, 500, 1000.

دقت پرسش و فراخوانی پرسش که مجموعه q را اصطلاحات محبوب پرتکرار در یک query log در نظر گرفته شده است (۸ ویژگی)، احتمال trigram مستقل، انتروپی trigram. به طورکلی ۲۴ ویژگی برای هر صفحه استخراج شد.

پس ویژگی های مبتنی بر محتوای صفحات برای رسیدن به ویژگی های مبتنی بر محتوای میزبان ترکیب می شود. فرض کنید h یک میزبان در برگیرنده m صفحه ی وب باشد ، به وسیله مجموعه $P=\{p_1,\ldots,p_m\}$ مشخص شود. $\hat{p}$ را صفحه ی خانگی میزبان h در نظر گرفته و p_P^* را صفحه با بزرگترین رتبه صفحه مابین صفحات در P در نظر گرفته می شود. c(P) ویژگی های محتوایی ۲۴ بعدی از صفحه P می باشد. برای میزبان h بردار ویژگی های مبتنی بر محتوا c(h) مطابق زیر تشکیل می شود.

$C(h)=<c(\hat{p}),c(p_P^*),E[c(p)],Var[c(p)]>$

در اینجا E[c(p)] میانگین همه بردارهای c(p) است و $p \in P$ و Var[c(p)] واریانس c(p) است. بنابراین برای هر میزبان داریم ۴ *۲۴=۹۶ ویژگی محتوایی، به طور کلی ۱۴۰+۹۶=۲۳۶ ویژگی مبتنی بر لینک و ویژگی محتوا تعریف می شود.
در فرایند ترکیب ویژگی های صفحه، میزبان های h برای صفحه خانگی $\hat{p}$ یا صفحه ی p_P^* بیشینه رتبه صفحه را نادیده گرفته شده است که در نمونه خلاصه ارائه نشده است [۴۴].
طبقه بندی کننده:
به عنوان طبقه بندی کننده پایه پیاده سازی C4.5 (درخت تصمیم) در وکا را استفاده شده است [۲۶]. با استفاده از هر دوی ویژگی های مبتنی بر لینک و محتوا، درخت نتیجه ۴۱ ویژگی واحد دارد که ۱۸ تای آنها ویژگی های مبتنی بر محتوا هستند. برای کمینه کردن خطای misclassify از درخت تصمیم حساس به هزینه استفاده شده و هزینه صفر را برای طبقه بندی به درستی در نظر گرفته شده و مجموعه هزینه misclassify یک میزبان هرزنامه را به عنوان نرمال R بار بزرگتر نسبت به misclassify یک میزبان نرمال به عنوان هرزنامه اعمال شده است. سپس طبقه بندی کننده پایه را با استفاده از bagging بهبود داده شده است.

bagging نتایج را به وسیله کاهش نرخ مثبت غلط بهبود می دهد. درخت تصمیم ایجاد شده به وسیله bagging اندازه ای مشابه با درخت تصمیم بدون bagging دارد و از ۴۹ ویژگی واحد استفاده می کند که ۲۱ تا از آنها ویژگی های محتوایی هستند.

[80] Corpus precision
[81] Corpus recall

طبقه بندی که پایه و اساس آزمایشات ما در آینده نیز می باشد از bagging با درخت تصمیم حساس به هزینه R=30 استفاده می کند.

خوشه بندی:
به طور مستقیم اگر اکثریت یک خوشه به عنوان هرزنامه پیش بینی شده است، پیش بینی برای همه میزبانها در خوشه به هرزنامه تغییر داده شده است. به طور مشابه اگر اکثریت یک خوشه به عنوان غیرهرزنامه پیش بینی شده است همه میزبانهای این خوشه را به عنوان غیرهرزنامه پیش بینی شده است.

گراف G با استفاده از الگوریتم‌های خوشه بندی گراف METIS خوشه بندی شده‌اند. ۱۱۴۰۰ میزبان گراف را به ۱۰۰۰ خوشه تقسیم بندی شدند. براساس نتایج تعداد خوشه‌ها زیاد مهم نیست و به نتایج مشابهی برای تقسیم گراف به ۵۰۰ و ۲۰۰۰ خوشه دست یافته شده است.

فرض کنید که خوشه بندی G از m خوشه ی c_1 تا c_m تشکیل شده است که پارتیشن جدای v را شکل می دهند. فرض $p(h) \in [0..1]$، پیش بینی یک الگوریتم طبقه بندی ویژه C باشد، برای هر میزبان h یک میزان p(h) مساوی با صفر غیرهرزنامه بودن و یک مقدار ۱ هرزنامه بودن را نشان دهد (به طور رسمی ، p(h) را درجه هرزنامه گی پیش بینی شده گراف h می خوانیم). برای هر خوشه ی Cj و j=1..m، میانگین هرزنامه گی به صورت زیر تعریف می شود.

$$\sum_{h \in Cj} p(h)\ \mathrm{p(Cj)}=\frac{1}{|Cj|} \qquad (۳-۸)$$

الگوریتم ما از دو آستانه استفاده می کند، یک آستانه کمتر t_1 و یک آستانه بالاتر t_n. برای هر خوشه Cj اگر $P(C_j) \le t_L$ همه میزبانها در C_j به عنوان هرزنامه برچسب می خوردند و P(h) با صفر تنظیم می شود برای همه $h \in C_j$. به طور مشابه اگر $P(C_j) \ge t_n$ همه میزبان ها در C_j به عنوان هرزنامه برچسب می خورند و p(h) مساوی ۱ تنظیم می شود.

انتشار

شبیه سازی قدم زدن تصادفی را از گرههایی که در طبقه بندی کننده پایه به عنوان "هرزنامه" برچسب گذاری شده است، شروع کرده، یک پیوند با احتمال α را دنبال نموده و با احتمال 1-α به گره هرزنامه برگردانده می شود. زمان برگشت به یک گره هرزنامه ، گره با احتمال متناسب با پیش بینی "spamcity" برداشته می شود، بعد از این فرآیند، از قسمت آموزشی داده برای یادگیری یک پارامتر آستانه استفاده می شود و از این آستانه برای طبقه بندی کردن قسمت تست به عنوان هرزنامه و غیرهرزنامه استفاده می شود.

یادگیری گرافیکی پشته ای

یک طرح یادگیری پایه C برای استخراج پیش بینی های اولیه برای اشیا در مجموعه داده استفاده می کند. سپس یک مجموعه ویژگی های اضافی را با ترکیب پیش بینی ها برای اشیا مرتبط در گراف برای هر شی خلق می کند. نهایتاً ویژگی های اضافی به ورودی C اضافه می کند و الگوریتم را برای بدست آوردن ویژگی های جدید اجرا می کند، پیش بینی ها برای داده ها بهتر هستند.

P(h)∈ [0..1] پیش بینی الگوریتم طبقه بندی کننده ویژه C که در بالا شرح داده شده می باشد . فرض کنید r(h) مجموعه صفحات مرتبط با h باشد، در اینصورت:

$$f(h)=\frac{\sum_{g \in r(h)} p(g)}{|r(h)|}$$

سپس f(h) را به عنوان یک ویژگی اضافی برای نمونه h در الگوریتم طبقه بندی C اضافه کرده و دوباره الگوریتم اجرا می شود. این فرایند می تواند به دفعات تکرار شود اما اغلب بهبود با تکرار نخست بدست آمده است [۴۴].

۳-۴-۵ تشخیص هرزنامه وب از طریق آنالیز مدلهای زبانی:

در این مطالعه رویکرد مدل زبان برای منابع اطلاعاتی استخراج شده از یک صفحه وب به منظور ارائه شاخص های با کیفیت بالا در تشخیص صفحات هرزنامه وب به کار می برد. از واگرایی کولبک-لیبلر به منظور توصیف ارتباط دو صفحه که به هم لینک شده اند استفاده شده است. با توجه به ماهیت متفاوت لینکهای خارجی و داخلی سه نوع از لینکها را که یک بهبود قابل توجه در طبقه بندی بدست میدادند متمایز کرده اند. در این مقاله چند ویژگی جدید براساس مدل های زبانی برای بهبود تشخیص هرزنامه وب ارائه داده شده است. از زمانی که مدل های آماری ارائه شدند و در اوایل ۱۹۶۰ در بازیابی اطلاعات مورد استفاده قرار گرفتند هیچ مزیتی واضح و روشنی برای مدل فضای برداری نداشت تا وقتی که پونته و کرافت کار خود را ارائه دادند که از مدل های مختلف احتمالاتی برای بازیابی اطلاعات استفاده کردند به عنوان مثال روش رتبه دهی احتمال پرس و جو [۹۱]. مدلهای آماری زبان برای ضبط ویژگیهای پنهان شده در متن ها توسعه یافته اند نظیر احتمال کلمات یا ترتیب کلمات در زبان. یک مدل آماری زبان (SLM) یک P(s) یک توزیع احتمالاتی روی رشته S است که تلاش میکند، منعکس کند که چگونه یک رشته S روی یک عبارت اتفاق می افتد .

یک مدل زبانی از هر منبع اطلاعاتی ساخته شده و سپس بررسی شده که این دو مدل زبانی چه تفاوتهایی نسبت به هم دارند.

طبقه بندی:

از دو مجموعه هرزنامه وب عمومی که در می ۲۰۰۶ و ۲۰۰۷ روی uk. دامین، خزش شده اند، استفاده شده است.

از یک الگوریتم طبقه بندی مبتنی بر درخت تصمیم حساس به هزینهبه همراه bagging بهره گرفته و برای آزمایش ها از ویژگی های لینک و محتوایی از پیش محاسبه شده استفاده شده که این ویژگی ها قبلاً در ۲ و ۱۴ ارائه شده اند [۹۲]. برای تمام پیش بینی ها از اعتبارسنجی ten-fold استفاده شده است.

مدلهای زبانی و ویژگی ها:

یکی از موفق ترین روشها مبتنی بر تجزیه و تحلیل توزیع اصطلاحات از مفهوم واگرایی کوبلک – لیبلر (KLD) برای محاسبه واگرایی مابین توزیع های احتمالاتی اصطلاحات دو سند خاص استفاده می کند [۹۳]. از KLD برای اندازه گیری واگرایی مابین دو واحد متن صفحات منبع و مقصد استفاده شده است. در طرح زیر دو نمونه از KLD به کار رفته به متن لنگر لینک و عنوان صفحه مورد اشاره توسط این لینک نمایش داده شده است.

$$KLD(T1||T2)=\sum_{t\in T1} P_{T1} \log \frac{P_{T1}(t)}{P_{T2(t)}}$$

$P_{T1}(t)$ احتمال اصطلاح t در نخستین واحد متن،$P_{T2}(t)$ احتمال اصطلاح t در دومین واحد متن می باشد.

KLD(Free Ringtones||Free Ringtones for Your Mobile Phone from remieringtones.com)=0.25

KLD(Best UK Reviews || Findabmw.co.uk-BMW Information Resoure)=3.89

واگرایی محاسبه شده KLD مابین متن لنگر یک پیوند و عنوان صفحه مورد اشاره به وسیله این لینک، نمونه استحراج شده از Web Spam UK2006.

مدلهای زبانی که استفاده شده بیشینه احتمال وقوع های unigram را تخمین می زند. نتایج نشان داده که با smoothing بهبود حاصل شده است هرچند تفاوت ناچیز است، علاوه بر این زمان محاسبه افزایش یافته است، به این دو دلیل ما تصمیم گرفته شده که از smoothing برای مدلهای زبانی در این کار استفاده نشود .

ویژگی ها:

مقادیر مختلف با محاسبه واگرایی KL مابین یک یا چند منبع اطلاعاتی از هر صفحه بدست آمده است. به ویژه به سه منبع اطلاعاتی از هر صفحه توجه شده i) متن لنگر ii) اطراف متن لنگر iiii)اصطلاحات URL. همچنین سه منبع اطلاعاتی را از صفحه مقصد بدست آورده اند: i) عنوان ii) محتوای صفحه iii) برچسب های متا.

در این مطالعه مقادیر مختلف واگرایی برای "متن لنگر– محتوا" (محتوای صفحه مقصد)، محیط متن لنگر-محتوا، اصطلاحات URL–محتوا(محتوای صفحه مقصد)، متن لنگر – عنوان(عنوان صفحه هدف)، اطراف متن لنگر-عنوان، اصطلاحات URL – عنوان، عنوان–محتوا(ارتباط بین عنوان و محتوای صفحه در همان سایت)، متا برچسب ها و منابع اطلاعاتی دیگر نظیر متن لنگر و محیط متن لنگر از صفحه منبع و محتوای صفحه و اصطلاحات URL از صفحه هدف مورد محاسبه قرار گرفته است.

ترکیب منابع اطلاعاتی

منابع مختلف اطلاعاتی از صفحه ی منبع ترکیب شده اند.متن لنگر (A)، محیط متن لنگر (S) و اصطلاحات URL (U) به عنوان منابع اطلاعاتی استفاده کرده و همچنین دو منبع جدید اطلاعاتی را پیشنهاد داده اند : ترکیب متن لنگر و اصطلاحات URL (AU) و ترکیب محیط اطراف متن لنگر و اصطلاحات URL (SU).علاوه بر این منابع اطلاعاتی از صفحه هدف نیز مورد توجه قرار گرفته است (محتوای صفحه (P)، عنوان صفحه (T) و متا برچسب ها (M)).

مابین لینک های داخلی و خارجی به منظور آنالیز واگرایی تمایز قائل شده و بنابراین برای هر صفحه وب ما ویژگی های سه گانه داریم: ۱۴ ویژگی برای لینک های خارجی، ۱۴ ویژگی برای لینک های داخلی و ۱۴ ویژگی برای لینک های داخلی و خارجی. میانگین توزیع هرزنامه در لینک های خارجی (KL≈3) نسبت به لینک های داخلی (KL≈ ۴/۵) بالاتر است.

از Web Spam uk2007 و Web Spam uk2006 استفاده شده است. از ویژگیهای از پیش محاسبه شده موجود برای مجموعه داده های عمومی استفاده کرده. به ویژه از ویژگی های مبتنی بر محتوا و ویژگی های مبتنی بر لینک تغییریافته استفاده شده است. و سرانجام ویژگی های مدلهای زبانی و محتوا و لینک را به منظور رسیدن به طبقه بندی کننده با دقت بیشتر ترکیب کرده اند. برای طبقه بندی از الگوریتم های meta cost (درخت تصمیم حساس به هزینه به همراه bagging) پیاده سازی شده در وکا استفاده کرده اند.

بهترین F-measure در آزمایشات زمانی بدست آمد که ویژگی های مدلهای زبانی و محتوایی ترکیب شدند (C U L U M) با بهبود ۲ درصد نسبت به پایه [۹۴].

۶-۴-۳ تاثیر زبان صفحه بر ویژگی های تشخیص هرزنامه وب:

تاثیر زبان صفحه بر ویژگی های شناسایی هرزنامه بررسی شده است. به بررسی نحوه توزیع مجموعه ویژگی های تشخیص انتخابی و تغییر آنها بر طبق زبان صفحه پرداخته شده و علاوه بر آن ما به مطالعه تاثیر زبان صفحه بر روی نرخ تشخیص طبقه بندی کننده فرضی با استفاده از مجموعه انتخابی ویژگی ها پرداخته شده است. نتایج تحلیلی نشان می دهند که انتخاب ویژگی های مناسب برای یک طبقه بندی کننده که تفکیک کننده صفحات هرزنامه است، بسیار زیاد وابسته به زبان صفحه وب است.

در ابتدا به بررسی نحوه توزیع ویژگی های تشخیص انتخابی پرداخته که مطابق با زبان صفحه با هم تفاوت دارند. دوم اینکه آزمایشاتی را اجرا شده که به بررسی تاثیر زبان بر تشخیص هرزنامه وب و نرخ اشتباه با استفاده از ویژگی های تشخیصی ثابت برای یک طبقه بندی کننده فرضی می پردازنند. از زبان های انگلیسی و عربی به عنوان مطالعات موردی استفاده شده است.

ویژگی های مورد استفاده: ۱- میزان متن لنگر در صفحه وب ۲-تعداد کلمات در صفحه وب ۳-میانگین طول کلمات در صفحه وب ٤-تعداد کلمات در عنوان صفحه وب، چرا که هرزنامه نویسان ممکن است از واژگان غیر مرتبط در عنوان ها استفاده کنند تا نمره عنوان را در روند رده بندی افزایش دهند. ٥-نرخ بهم فشردگی صفحه وب ٦-تعداد واژگان منحصر به فرد در صفحه وب ۷-تعداد کاراکترها در عنصر متا، چراکه هرزنامه نویسان از واژگان کلیدی برای افزایش رده بندی صفحه استفاده می کنند ۸-تعداد کلمات در عنصر متا چرا که هرزنامه نویسان از واژگان کلیدی برای افزایش رده بندی صفحه استفاده می کنند ۹-طویل ترین واژه در صفحه وب، چراکه هرزنامه نویسان ها از واژگان طولانی برای افزایش رده بندی صفحه استفاده می کنند ۱۰-کوتاه ترین واژه در صفحه وب ، چراکه هرزنامه نویسان از واژگان طولانی برای افزایش رده بندی صفحه استفاده می کنند ۱۱-تعداد تصاویر در صفحه وب.

مجموعه داده ای مورد استفاده:

دو مجموعه داده ای وب اسپم مورد استفاده قرار گرفتند: UK-2011 [۹٥]، وب اسپم گسترده عربی ۲۰۱۱ [۹۶].

متدلوژی:

از درخت تصمیم MATLAB استفاده شده که مبتنی بر بهترین روش های شناخته شده برای تشخیص هرزنامه است. علاوه بر آن از ارزیابی متقاطع و هرس کردن بهره برده است (یک تکنیک برای کاهش سایز درخت تصمیم با حذف قسمتهای کمی از درخت که تاثیر چندانی روی طبقه بندی ندارد). برای ارزیابی متقاطع ویژگی ها از نمونه گیری تصادفی ۱۵۰۰ صفحه وب از میان ۳۶۸۸ صفحه در مجموعه ۱ و ۹۹۸۸ صفحه در مجموعه ۲ استفاده شد.

نتایج:

ابتدا ماهیت و پراکندگی هر ویژگی را در مجموعه داده ای انگلیسی و عربی بررسی کرده و سپس از طبقه بندی کننده درخت تصمیم برای هر دو زبان انگلیسی و عربی استفاده شده و خطای طبقه بندی با استفاده از این طبقه بندی کننده نشان داده شده است. از ارزیابی متقاطع leave one out استفاده می شود. طبقه بندی کننده ساخته می شود و مشاهدات یک به یک حذف شده و سپس بررسی می شود که آیا مشاهده حذف شده به درستی طبقه بندی می شود یا خیر(با جایگزینی).

باید درختی انتخاب شود که دارای حداقل خطای طبقه بندی است. یک راه برای انتخاب درخت با یک خطای استاندارد به همراه حداقل خطای طبقه بندی وجود دارد (تابع TREETEST MATLAB FUNCTION).

در وهله اول هر ۱۱ ویژگی برای درخت تصمیم انتخاب می شود تا عملکرد کلی آنها بررسی شود. نرخ شناسایی برای صفحات وب عربی بالاتر از انگلیسی است. این امر به این دلیل است که صفحات وب عربی نسبت به نمونه های انگلیسی برای گریز از تشخیص هرزنامه توسط موتور جستجوگر مهارت کافی را ندارند.

ویژگی ۸ ارائه دهنده بهترین عملکرد تشخیص در مجموعه داده ای انگلیسی است، ویژگی ۱ بهترین عملکرد را در مجموعه داده ای عربی دارد.

همچنین خطای طبقه بندی برای ترکیبات متفاوت هر ۱۱ ویژگی در مجموعه داده ای انگلیسی و عربی با استفاده از ترکیبات دو تایی ویژگی ها مورد محاسبه قرار گرفت. لازم به ذکر است که ویژگی ۷ در مجموعه داده ای در ترکیب با ویژگی های دیگر خوب عمل نمی کند. ویژگی های ٦ و۸ در ترکیب با دیگر ویژگی ها عملکرد خوبی دارند. در مجموعه داده ای عربی، استفاده از ویژگی ۱۰ بدترین عملکرد پراکندگی را به بار می آورد.

هم چنین خطای طبقه بندی برای ترکیبات ۳ تایی ویژگی ها مورد بررسی قرار گرفت. ویژگی های عمومی کمی وجود دارد که در هر دو مجموعه داده ای خوب عمل کنند همچون ۱۱ و ۱، اما با این وجود می توان ذکر نمود که انتخاب طبقه بندی کننده ویژگی ها بسته به زبان صفحه است.برای مجموعه داده ای انگلیسی، ویژگی های ۲ و ۳ به نسبت دیگرویژگی ها دارای نرخ مثبت و منفی غلط کمتری در ترکیب با دو ویژگی دیگر هستند (برای حداقل ۷۵ درصد از توزیع شان). علاوه برآن ، ویژگی های ۱ و ۱۱ در مجموعه داده ای عربی کمترین تعداد مثبت غلط و منفی غلط را در ترکیب با دو ویژگی دیگر دارند.از سوی دیگر ویژگی ۷ (تعداد کاراکترها در عناصر متا) در مجموعه داده ای انگلیسی و ویژگی ۱۰ (کوتاهترین کلمه در صفحه وب) در مجموعه داده ای عربی دارای بالاترین خطای رده بندی در ترکیب با هر جفت ویژگی هستند.

سپس ترکیبات چهار تایی از ویژگی ها برای خطای طبقه بندی مورد بررسی قرار گرفت. برای مجموعه داده ای انگلیسی ، درست مثل مجموعه ۳ تایی، ویژگی های ۲ و ۳ کمترین خطای طبقه بندی را در ترکیب با هر دو ویژگی دیگر بدست آورده اند. ویژگی های ۱ و ۱۱ در مجموعه داده ای عربی کمترین خطای طبقه بندی و کمترین میزان پراکندگی را در هنگام ترکیب با دو ویژگی دیگر دارند. در مجموعه داده ای انگلیسی و ویژگی ۱۰ در مجموعه داده ای عربی بالاترین خطای طبقه بندی را در ترکیب با دیگر ترکیبات۳ گانه ویژگی ها دارند.

نتایج نشان می دهد که ویژگی های اندکی وجود دارند که در هر دو مجموعه داده ای نتایج مشابهی را می دهد، کارایی ویژگی های متفاوت دیگر مطابق با زبان صفحه مورد آزمایش متفاوت است.

نتیجه این مطالعه بیان میدارد که هرزنامه نویسان یک زبان فرضی از مجموعه مشابهی از تکنیک های هرزنامه وب استفاده می کنند. با فرض بر اینکه محتوای صفحات وب عربی عموما در دسترس نیستند ، می توانیم تنها حدس بزنیم که تکنیک های کاربردی هرزنامه وب در مجموعه داده ای عربی ساده تر از نمونه های کاربردی در مجموعه داده ای انگلیسی هستند [۹۴].

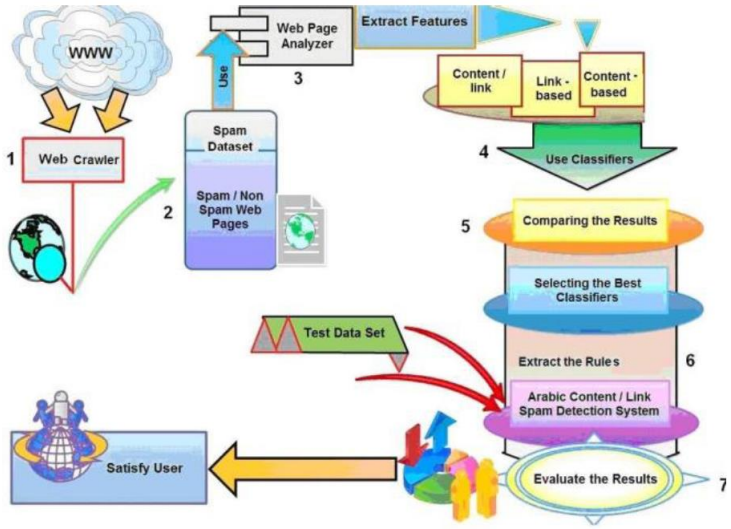
۷-۴-۳-رویکرد ترکیب ویژگی های مبتنی بر محتوا و لینک برای صفحات عربی:

محققین متدلوژی خود را مطابق با گام های زیر ارائه داده اند که می تواند برای ساختن سیستم تشخیص هرزنامه وب عربی محتوا / لینک مورد استفاده قرار گیرد.

۱-توسعه یک کاوشگر وب تعبیه شده که این کاوشگر صفحات وب را دانلود می کند و همه ی فوق پیوندها و محتوای صفحات وب را تجزیه و تحلیل می کند.

۲-ساخت یک مجموعه داده هرزنامه عربی که مجموعه داده آموزشی ۱۸۰۰۰ صفحه را دربرمی گیرد و مجموعه داده تست ۵۰۰۰ صفحه را دربرمیگیرد.

۴- توسعه یک آنالیزگر وب برای استخراج تعداد بیشتری ویژگی
۵- استفاده از سه الگوریتم طبقه بندی درخت تصمیم، logistic regression، K-NN که به وسیله وکا پشتیبانی می شود.
۶- مقایسه نتایج الگوریتم های طبقه بندی برای شناسایی بهترین الگوریتم
۷- استخراج قوانین بهترین الگوریتم طبقه بندی به منظور توسعه قسمت نهایی سیستم تشخیص هرزنامه عربی
۸- ارزیابی سیستم با استفاده از مجموعه داده تست



شکل ۳-۴: طرح کلی متدلوژی [۱۰۰]

در این تحقیق، داده های آموزشی به تعدادی گروه مبتنی بر درصد دقت تشخیص هرزنامه تقسیم شده اند. بهترین سه گروه با درصدهای هرزنامه متفاوت ۲ و ۳۰ و ۴۰ بودند.

هرزنامه نویسان وب عربی از کلمات کلیدی انباشته[۸]برای افزایش رتبه صفحات خود استفاده می کنند. کلمات کلیدی انباشته تکنیکی است مبتنی بر کپی تعدادی از کلمات در عناصر HTML. همچنین هرزنامه نویسان عربی از ارتباط کلمات انگلیسی بی معنی و حروف عربی مطابق با آن کلمات در کلمات کلیدی انباشته استفاده می کنند. در نتیجه آنالیزگر وب توسعه داده شده هر کلمه انگلیسی را به کلمه عربی معنی دار یا بی معنی مطابق با آن تبدیل میکند و سپس از پایگاه داده ای که فهرست کلمات عربی را در بر می گیرد استفاده می کند تا تعیین کند این کلمه معنی دار است یا بی معنی[۹۷].

هرزنامه نویسان در صفحات وب عربی از دو نوع تکنیک هرزنامه نویسی مبتنی بر لینک استفاده می کنند; مزرعه لینک و دامنه های منقضی شده [۹۸].

آنالیزگر ویژگی های مبتنی بر محتوا بیان شده در [۹۸,۹۹] را محاسبه می کند و هم چنین ویژگی های مبتنی بر لینک نظیر تعداد لینک های خارجی و داخلی صفحه مورد بررسی و طول URL و تعداد کلی لینک های شکسته، تعداد کلی لینک های تغییر مسیریافته، تعداد کلی لینکها با متن خالی و تعداد کلی لینکهای خالی مورد توجه قرار گرفته است.

نتایج طبقه بندی صفحات هرزنامه

به کاربردن الگوریتم رگرسیون، K-NN و درخت تصمیم نشان می دهد که درخت تصمیم بهترین الگوریتم برای تشخیص انواع هرزنامه عربی می باشد. با توجه به اینکه درخت تصمیم بهترین نتایج را روی گروه ۲ درصدی دارد قوانین درخت تصمیم برای هر نوع هرزنامه استخراج شده و سپس از جاوا برای ساخت سیستم تشخیص هرزنامه استفاده نموده اند [۱۰۰].

[8] keyword stuffing [2]

۵-۳- جمع بندی:

در این فصل مجموعه داده های مورد استفاده محققین معرفی گردید. سپس مطالعات پیشین در سه گروه مبتنی بر محتوا، مبتنی بر لینک و مبتنی بر لینک و محتوا مورد بررسی قرار گرفت.

فصل پنجم
نتیجه گیری و کارهای آتی

۱-۵- نتیجه گیری:

ایده اصلی این پایان نامه اعمال روش های کاهش ویژگی و یافتن ویژگی های موثر در الگوریتم های داده کاوی می باشد و همچنین یافتن الگوریتم های بهینه داده کاوی که با اعمال ویژگی های کمتر نیز جوابهای بهینه و کارا حاصل شود.

مجموعه داده های موجود جهت داده کاوی UK2007، DC2010، MSN است که در این تحقیق از مجموعه داده UK2007 استفاده گردید. مجموعه داده انتخابی دارای ۱۴۰ ویژگی، شامل ویژگی های محتوا و لینک است.

در ابتدا به کمک فیلترها مجموعه داده پیش پردازش گردید. بعد از پیش پردازش مجموعه داده، ۳۴ طبقه بندی کننده موجود در وکا بر روی مجموعه داده با ۱۴۰ ویژگی جهت انتخاب الگوریتم های بهینه آزمایش گردید.

بهترین نتایج از نظر معیار F_measure از اعمال الگوریتم های Random Forest، Decorate، Rotation Forest، Threshold Selector، IBK، FT بدست آمد.

سپس الگوریتم های کاهش ویژگی ChisquaredAttributeEval، cfssubseteval، Consistencysubseteval، GainRatioAttributeEval، InfoGain، Symetricaluncertattributeeval و principalcomponent با روش های جستجوی متفاوت بر روی ۱۴۰ ویژگی جهت کاهش ویژگی ها به کار گرفته شد.

۶ الگوریتم که بر روی ۱۴۰ ویژگی بهترین جوابها را نتیجه می داد، بر روی ویژگی های بدست آمده از الگوریتم های کاهش ویژگی آزمایش شد.

۶ الگوریتم Random Forest، Decorate، Rotation Forest، Threshold Selector، IBK، FT مورد آزمایش قرار گرفت.

ارزیابی این الگوریتم ها بر روی ویژگی های کاهش یافته نشان داد که بهترین نتیجه از نظر معیار F_measure و هم چنین تعدادِ کمتر ویژگی ها مربوط به الگوریتم کاهش ویژگی cfssubseteval و روش جستجوی Linear Forward selection می باشد که تعداد ویژگی ها به ۱۹ کاهش یافته و از نظر معیار سنجش با توجه به مقادیر بدست آمده ۰/۳۳۹ و ۰/۹۴۱ برای کلاس هرزنامه و میانگین وزنی و مقایسه آن با نتایج الگوریتم Random Forest بر روی ۱۴۰ ویژگی (۰/۳۱۵،۰/۹۴۱) بهبود حاصل شده است.

بهترین نتیجه درصد درستی تشخیص کلاس هرزنامه و میانگین وزنی دو کلاس از نظر معیار انتخابی مربوط به الگوریتم CFS و روش جستجوی Rank search می باشد که با ۱۰۱ ویژگی و الگوریتم Rotation Forest حاصل شده است.

۲-۵- کارهای آتی:

۱- استفاده از الگوریتم های بهینه بر روی دیگر پیکره های موجود
۲- استفاده از روش های ترکیبی و بررسی نتایج حاصله
۳- ایجاد بانک های اطلاعاتی جامع تر تا ویژگی های بیشتری به جهت مطالعه دقیق تر فراهم گردد.

منابع:

[1] Han, J., Kamber, M., 2001, ***"Data Mining: Concepts and Techniques"***, Morgan Kaufman, San Francisco.

[2] Abernethy, J., Chapelle,O., Castillo.,C., Nov.2010, ***"Graph regularization methods for web spam detection"***. Mach. Learn., Vol. 81.

[3] http://searchengineland.com/ businessweek-dives-deep-into-googles-search-quality-27317, 2011.

[4] Eiron, N., McCurley, K. S., Tomlin., J. A., 2004, ***"Ranking the web frontier"***, In Proceedings of the 13th International Conference on World Wide Web, WWW'04, New York.

[5] Page, L., Brin, S., Motwani, R., Winograd., T., 1998, ***"The pagerank citation ranking: Bringing order to the web"***.

[6] Jennings, R., 2005, ***"The global economic impact of spam"***, Ferris Research.

[7] Silverstein, C., Marais, H., Henzinger, M., Moricz, M., Sept. 1999, ***"Analysis of a very large web search engine query log"***, SIGIR Forum, 33.

[8] Bencz´ur, A. A., Csalog´any, K., Sarl´os, T., Uher, M., May 2005, ***"Spamrank: Fully automatic link spam detection work in progress"***, In Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'05,.

[9] Castillo, C., Donato, D., Gionis, A., Murdock, V., Silvestri, F.,2007, ***"Know your neighbors: web spam detection using the web topology"***. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR' 07, Amsterdam, The Netherlands.

[10] Gulli, A., Signorini ,A., 2005, ***"The indexable web is more than 11.5 billion pages"***, In *Proceedings of the 14th World Wide Web Conference (WWW), Special interest tracksand posters*, pages 902–903.

[11] The Official Google Blog, 2008.

[12] Cho, J. , Garcia-Molina, H.,2000, ***" The evolution of the web and implications for an incremental crawler"***, In *The VLDB Journal*, pages 200–209.

[13] Bar-Yossef, Z., Broder, A. Z., Kumar, R., Tomkins, A.,*2004*, ***" Sic transit Gloria telae: Towards an understanding of the web's decay"***, In *Proceedings of the 13th World Wide Web Conference (WWW)*, pages 328–337. ACM Press.

[14] Berners-Lee, T., Hendler, J., Lassila, O., 2001, ***"The semantic web. Scientific American"***.

[15] Davison, B. D., 2000, ***" Recognizing nepotistic links on the web"***, In *AAAI-2000 Workshop on Artificial Intelligence for Web Search*, pages 23–28, Austin, TX.

[16] Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A., Wiener, J. , 2000, ***" Graph structure in the web"***, In *Proceedings of the 9th World Wide Web Conference (WWW)*, pages 309–320. North-Holland Publishing Co. .

[17] Silverstein, C., Marais, H., Henzinger, M., Moricz, M., 1999, ***"Analysis of a very large web search engine query log"*** *SIGIR Forum*, 33(1):6–12, 1999.

[18] Arasu, A., Cho, J., Garcia-Molina, H., Paepcke, A., Raghavan, S., August 2001, ***" Searching the web. ACM Transactions on Internet Technology (TOIT)"***, 1(1):2–43,.

[19] Brin , S., Page, L., 1998, ***" The anatomy of a large-scale hypertextual Web search engine"***, *Computer Networks and ISDN Systems*, 30(1-7):107–117.

[20] Risvik , K. M., Michelsen, R., 2002,***" Search engines and web dynamics"***, *Computer Networks*, 39(3):289–302.

[21] Gyongyi, Z., Garcia-Molina, H., 2004, ***" Web Spam Taxonomy"***, Technical Report, Stanford University.

[22] Baeza-Yates, R., Ribeiro-Neto, B., 1999, ***" Modern Information Retrieval"***, Addison-Wesley, Boston.

[23] S. E., Robertson , Jones, K. S., 1988, ***"Relevance weighting of search terms"***, In *Document retrieval systems*, pages 143–160. Taylor Graham Publishing, London, UK.

[24] Csalogány, K. ,2009, ***" Methods for Web Spam Filtering"***, Technical Report, Eötvös Loránd University.

[25] Salton, G., Buckley, C., 1988, ***"Term-weighting approaches in automatic text retrieval"***, *Information Processing & Management*, 24(5):513-523.

[26] Page, L., Brin, S., Motwani, R., Winograd, T., 1998, ***"The PageRank citation ranking: Bringing order to the web"***, Technical Report 1999-66, Stanford University.

[27] Motwani , R., Raghavan, P., 1995, ***" Randomized Algorithms"***, Cambridge University Press.

[28] Brin, S., Page, L., Apr.1998, ***"The anatomy of a large-scale hypertextual Web search engine"***, In *Proceedings of the 7th International World Wide Web Conference*, pages 107-117, Brisbane, Australia.

[29] Chakrabarti, S., Dom, B. E., Kumar, S. R., Raghavan, P., Rajagopalan, S., Tomkins, A., Gibson, D., Kleinberg, J.***," Mining the Web's link Structure"***. *Computer*, 32(8):60–67.

[30] Bianchini, M., Gori, M., Scarselli, F., 2005***," Inside PageRank"***, *ACM Transactions on Internet Technology*, 5(1):92–128.

[31] Liu, B., 2007, ***"Web Data minig. Exploring Hyperlinks, Contents, and Usage Data***", *pages 230-233.* Springer-Verlag Berlin Heidelberg, New York.

[32] Gyöngyi, Z., Garcia-Molina, H., 2005, ***" Web spam taxonomy"*** In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, Chiba, Japan.

[33] Wu, B.,2007,***" Finding and Fighting Search Engine Spam"***, Phd thesis, Lehigh University.
[34] Hastie, T., Tibshirani, R., Friedman, J. H., 2001, ***" The elements of statistical learning: data mining, inference, and prediction: with 200 full-color illustrations"***. New York: Springer-Verlag.

[35] Liu, B., 2007, ***"Web Data minig. Exploring Hyperlinks, Contents, and Usage Data***", *pages 63-64.* Springer-Verlag Berlin Heidelberg, New York.

[36] Shannon, E., 1984, ***" A Mathematical Theory of Communication"***, In *Bell System Technical Journal*, 27: pp. 379–423.

[37] Liu, B., 2007, ***"Web Data minig. Exploring Hyperlinks, Contents, and Usage Data***", *pages 97-103.* Springer-Verlag Berlin Heidelberg, New York.

[38] Breiman, L., 1996, ***"Bagging Predictors"***, *Machine Learning*, 24(2), 123–140.

[39] Salton , G., McGill, M.,1983, ***"An Introduction to Modern Information Retrieval"***, New York, NY: McGraw-Hill.

[40] Freund, Y., Schapire, R. E., 1996, ***"Experiments with a New Boosting Algorithm"***, In *Proc. of the 13th Intl. Conf. on Machine Learning* (*ICML'96*), pp. 148–156.

[41] Quinlan, J. R., 1996, ***"Bagging, Boosting, and C4.5"***, In *Proc. of National Conf. on Artificial Intelligence* (*AAAI-96*), pp. 725-730.

[42] Liu, B., 2007, ***"Web Data minig. Exploring Hyperlinks, Contents, and Usage Data***", *pages 72-75.* Springer-Verlag Berlin Heidelberg, New York.

[43] Wu, B., Goel, V., Davison, B. D., May 2006 ***"Propagating trust and distrust to demote web spam"***, In Proceedings of the Workshop on Models of Trust for the Web, Edinburgh,Scotland.

[44] Castillo, C., Donato, D., Gionis, A., Murdock, V., Silvestri, F.,2007, ***"Know your neighbors: web spam detection using the web topology"***. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR' 07, Amsterdam, The Netherlands.

[45] Becchetti, L., Castillo, C., Donato, D., Boldi, P., Leonardi, S., Santini, M., Vigna, S., 2006, ***"A Reference Collection for Web Spam"***.

[46] Mahmoudi, M., Yari, A., Khadivi, S., 2010, ***"Web Spam Detection Based on Discriminative Content and Link Features"***, 5th International Symposium on Telecommunications

[47] Fetterly, D., Manasse, M., Najork, M., 2004, ***"Spam, damn spam, and statistics: using statistical analysis to locate spam web pages"***, In Proceedings of the 7th International Workshop on the Web and Databases: collocated with ACM SIGMOD/PODS 2004, WebDB'04, Paris, France.

[48] Erd´elyi, M., Garz´o, A., Bencz´ur, A. A., 2011, ***" Web spam classification: a few features worth more"***, In Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality, WebQuality'11, Hyderabad, India.

[49] Ntoulas, A., Najork, M., Manasse, M., Fetterly, D., 2006***,"Detecting spam web pages through content analysis"***, In Proceedings of the 15th International Conference on World Wide Web, WWW'06, Edinburgh, Scotland.

[50] Fetterly, D., Manasse, M., Najork, M., 2007, ***"Detecting phrase-level duplication on the world wide web"***, In Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'05, Salvador, Brazil.

[51] D Fetterly, D., Manasse, M., Najork, M., Oct. 2003, ***"On the evolution of clusters of near-duplicate web pages"***, J. Web Eng., 2.

[52] Broder, A. Z., 1993, ***"Some applications of rabin's fingerprinting method"***, In Sequences II: Methods in Communications, Security, and Computer Science. Springer-Verlag.

[53] Rabin, M., 1981***,"Fingerprinting by Random Polynomials"***, Technical report, Center for Research in Computing Technology, Harvard University.

[54] Erd´elyi, M., Garz´o, A., Bencz´ur, A. A., 2011, ***" Web spam classification: a few features worth more"***, In Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality, WebQuality'11, Hyderabad, India.

[55] Urvoy, T., Lavergne, T., Filoche, P., Aug. 2006, ***"Tracking Web Spam with Hidden Style Similarity"***, In Proceedings of the Second International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'06, Seattle, Washington.

[56] Mishne, G., Carmel, D., Lempel, R., May 2005, ***"Blocking blog spam with language model disagreement"***, In Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'05, Chiba, Japan.

[57] Hiemstra, D., 2009, ***"Language models"***, In Encyclopedia of Database Systems.

[58] Piskorski, J., Sydow, M., Weiss, D., 2008, ***"Exploring linguistic features for web spam detection: a preliminary study"***, In Proceedings of the 4th International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'08, Beijing, China.

[59] Sydow, M., Piskorski, J., Weiss, D., Castillo, C. ,2007, ***"Application of machine learning in combating web spam"***, *Polish Ministry of Science gran.*

[60] Bencz´ur, A., B´ır´o, I., Csalog´any, K., Sarl´os T., 2007, ***"Web spam detection via commercial intent analysis"***, In Proceedings of the 3rd International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'07.

[61] Chellapilla, K., Chickering, D., 2006, ***"Improving cloaking detection using search query popularity and monetizability"***, *AIRWeb'06*, pp. 20-26.

[62] Wahsheh, H. A., Al-Kabi, M. N., 2011, ***"Detecting Arabic Web Spam"***, *The 5th* International *Conference on* 21 Wahsheh et al. *Information Technology, ICIT 2011*, Paper ID (631), pp. 1-8.

[63] Wang, W., Zeng, G., Sun, M., Gu, H., Zhang, Q.,2007, ***"EviRank: An Evidence Based Content Trust Model for Web Spam Detection"***. *APWeb/WAIM*, pp. 299-307.

[64] Wang,W., Zeng, G., Tang, D., 2010, ***"Using evidence based content trust model for spam detection"***, *Expert Systems with Applications,* 37 (8), pp. 1-8.

[65] Wang,W., Zeng, G., 2007, ***"Content Trust Model for Detecting Web Spam"***. *IFIP International Federation for Information Processing*. pp. 139-152.

[66] Ntoulas, A., Najork, M., Manasse, M., Fetterly, D., 2006,***"Detecting spam web pages through content analysis"***, In Proceedings of the 15th International Conference on World Wide Web, WWW'06, Edinburgh, Scotland.

[67] Wahsheh, H., Doush, I. A., Al-Kabi, M., Alsmadi, I., Al-Shawakfa, E., 2012, ***"Using Machine Learning Algorithms to Detect Content-based Arabic Web Spam"*** Journal of Information Assurance and Security.ISSN 1554-1010 Volume 7 ,pp. 14-23.

[68] Spirin, N., Han, J., 2011, ***"Survey on Web Spam Detection: Principles and Algorithms"***, ACM SIGKDD Explorations Newsletter, Volume 13, pp. 50-64.

[69] Gyöngyi, Z., Garcia-Molina, H., Pedersen, J., 2004,***"Combating web spam with TrustRank"***, In *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB)*, pages 576–587, Toronto, Canada.

[70] Bencz´ur, A. A., Csalog´any, K., Sarl´os, T., Uher, M., 2005, ***"Spamrank: Fully automatic link spam detection work in progress"***, In Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'05.

[71] Guha, R., Kumar, R., Raghavan, P., Tomkins, A., 2004, ***"Propagation of trust and distrust"***, In Proceedings of the 13th International Conference on World Wide Web, WWW'04, New York, NY.

[72] Gyongyi, Z., Garcia-Molina, H., 2006, ***"Link spam detection based on mass estimation"*** , In Proceedings of the 32nd International Conference on Very Large Databases, VLDB'06.

[73] Caverlee, J., Liu, L., 2007, ***"Countering web spam with credibility-based link analysis"***. In Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing, PODC'07, Portland, OR.

[74] Baeza-Yates, R., Boldi, P., Castillo, C., 2006, ***"Generalizing pagerank: damping functions for link-based ranking algorithms"***, In Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'06, Seattle,Washington.

[75] Becchetti, L., Castillo, C., Donato, D., Leonardi, S., Baeza-Yates, R., , 2006, ***"Using rank propagation and probabilistic counting for link-based spam detection"***, In Proceedings of the Workshop on Web Mining and Web Usage Analysis, WebKDD'06, Philadelphia, USA.

[76] Bharat,K., Henzinger, M. R., 1998, ***"Improved algorithms for topic distillation in a hyperlinked environmen***t", In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'98, Melbourne, Australia.

[77] Nomura,S., Oyama, S., Hayamizu, T., Ishida, T., Nov.2004, ***"Analysis and improvement of hits algorithm for detecting web communities"***, Syst. Comput. Japan, 35.

[78] Lempel, R., Moran, S., 2001, ***"SALSA: the stochastic approach for link-structure analysis"***, ACM Trans. Inf. Syst., 19.

[79] Roberts, G., Rosenthal, J., 2003, ***"Downweighting tightly knit communities in World Wide Web rankings"*** Advances and Applications in Statistics (ADAS).

[80] Davison, B., ***"Recognizing nepotistic links on the web. In Workshop on Artificial Intelligence for Web Search"***, AAAI'00.

[81] Becchetti, L., Castillo, C., Donato, D., Leonardi, S., Baeza-Yates, R., 2006, ***"Link-based characterization and detection of web spam"***, In Proceedings of the Second In-ternational Workshop on Adversarial Information Retrieval on the Web, AIRWeb'06, Seattle, USA.

[82] Zhou, D., Bousquet, O., Lal, T. N., Weston, J., Sch¨olkopf, B., Olkopf, B. S., 2003, ***"Learning with Local and Global Consistency"***, In Proceedings of the Advances in Neural Information Processing Systems 16, volume Vol. 16.

[83] Kou, Z., Cohen, W. W., April 2007, ***"Stacked graphical models for efficient inference in markov random fields"***, In Proceedings of the Seventh SIAM International Conference on Data Mining, SDM'07, Minneapolis, Minnesota.

[84] Robertson, S. E., Walker, S., 1994, ***"Some simple e_ective approximations to the 2-poisson model for probabilistic weighted retrieval"***, In In Proceedings of SIGIR'94, pages 232{241. Springer-Verlag.

[85]. Google Search Engine Ranking Factors. http://www.seomoz.org/article/search-ranking-factors. Accessed 29 June 2009.

[86]. Bifet, A., Castillo, C., Chirita, P. A., Weber, I., 2005, ***"An analysis of factors used in search engine ranking. In"***, Adversarial Information Retrieval on the Web.

[87]. Evans, M.P., 2007, ***"Analysing Google rankings through search engine optimization data"***, Internet Res. **17**(1), 21–37

[88]. Karlberger,C., Bayler,G., Kruegel,C., Kirda, E., 2007, ***"Exploiting redundancy in natural language to penetrate bayesian spam filters",*** In:First USENIX Workshop on Offensive Technologies (WOOT07).

[89] Egele, M., Kolbitsch, C., Platzer, C., 2009, ***"Removing web spam links from search engine results"***, Springer-Verlag France.

[90] Cohen, W. W., Kou, Z., 2006, ***"Stacked graphical learning:approximating learning in markov random fields using very short inhomogeneous markov chains"***, Technical report.

[91] Ponte, J. M., Croft, W. B., 1998, ***"A language modeling approach to information retrieval"***, In *SIGIR '98:Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, New York, NY,USA.

[92] Witten, I. H., Frank,E., 2005, ***"Data Mining: PracticalMachine Learning Tools and Techniques***",Morgan Kaufmann, 2 edition.

[93] Cover, T. M., Thomas, J. A., 1991, ***"Elements of information theory"*** Wiley-Interscience, New York, NY, USA.

[94] Martinez-Romo, J., Araujo, L., 2009, ***"Web Spam Identification Through Language Model Analysis***" ACM.

[95] Uk-2011 web spam dataset. Accessed: May 2012. https://sites.google.com/site/heiderawahsheh/home/web-spam-2011-datasets/uk-2011-web-spam-dataset.

[96] Extended arabic web spam 2011 dataset. Accessed: May 2012. https://sites.google.com/site/heiderawahsheh/home/web-spam-2011-datasets/arabic-web-spam-2011-dataset.

[97] Gadge, J., Sane, S., Kekre, H., 2011,***"Layered Approach to Improve Web Information Retrieval***", Proceedings on 2nd National Conference on Information and Communication Technology NCICT. v7, pp. 28-32.

[98] Wahsheh, H., Al-Kabi, M., Alsmadi, I., 2012, ***"Evaluating Arabic spam Classifiers Using Link Analysis"***, In Proceeding of the 3rd International Conference on Information and Communication Systems (ICICS'12), ACM, Irbid, Jordan. (2012d) pp.1-5.

[99] Wahsheh, H. A., Al-Kabi, M. N., 2011, ***"Detecting Arabic Web spam"***, The 5th International Conference on Information Technology (ICIT 2011), Amman-Jordan, pp. 1-8.

[100] Wahsheh, H. A., Al-Kabi, M. N., Alsmadi, I. M., 2013, ***"A link and Content Hybrid Approach for Arabic Web Spam Detection"***, MECS.

[101] Witten, I.H., FranK, E., (2005). ***"Data Mining: Practical Machine Learning Tools and Techniques, Second Edition"***, Morgan Kaufmann Publishers Inc, ISBN:0120884070

[102] Nathan, P., (2005), "Enhancing Random Forest Implementation in Weka", Machine Learning Conference Paper for ECE591Q.

[103] Sharma, T.C., Manoj, J., (2013), "WEKA Approach for Comparative Study of Classification Algorithm", International Journal of Advanced Research in Computer and Communication Engineering ,Vol. 2, Issue 4.

[104] Mooney, R.J., Melville, P., (2003), "***Constructing Diverse Classifier Ensembles using Artificial Training Examples*****",** Proceedings of the IJCAI-2003, pp.505-510,Acapulco, Mexico.

[105] Kuncheva, L.I, Rodriguez, J.J., (2007), "***An Experimental Study on Rotation Forest Ensembles*****", technical report,** School of Electronics and Computer Science, University of Wales, Bangor, UK.

[106] *Liu, B.,(2007), " Web Data minig.* Exploring Hyperlinks, Contents, and Usage Data", Springer-Verlag Berlin Heidelberg, New York, *pages 112-113.*

پیوست ۱:

فهرست ویژگی های انتخابی با الگوریتم chisquaredAttributeEval و روش جستجوی Ranker

AVG_64	AVG_68	HMG_41	AVG_53
HST_20	STD_76	AVG_67	STD_96
STD_95	HMG_33	AVG_66	HST_19
AVG_65	AVG_50	STD_77	outdegree_hp
outdegree_mp	HMG_44	AVG_59	STD_53
HMG_43	AVG_69	STD_82	AVG_58
STD_86	STD_75	HST_16	STD_80
AVG_55	STD_90	AVG_60	STD_78
STD_81	HST_18	prsigma_hp	HST_21
HST_10	avgin_of_out_mp	AVG_70	STD_88
HST_22	STD_92	HMG_34	STD_84
STD_87	STD_94	STD_89	avgin_of_out_hp
HMG_42	HST_17	STD_91	AVG_61
HST_24	HMG_48	AVG_56	STD_93
indegree_hp	HST_9	HST_7	HMG_45
HMG_46	AVG_72	AVG_63	HST_12
HMG_32	AVG_62	AVG_71	prsigma_mp
STD_74	HST_8	HMG_31	HST_13
siteneighbors_2_mp	AVG_57	HMG_26	HST_2
neighbors_3_mp	neighbors_4_mp	HST_14	HST_5
neighbors_4_mp	indegree_mp	HMG_36	HMG_35
siteneighbors_2_hp	HMG_38	HST_6	siteneighbors_1_hp
truncatedpagerank_2_hp	siteneighbors_1_mp	siteneighbors_4_mp	HMG_37
truncatedpagerank_1_hp	truncatedpagerank_1_mp	pagerank_hp	truncatedpagerank_2_mp
truncatedpagerank_4_hp	truncatedpagerank_3_hp	pagerank_mp	truncatedpagerank_4_mp
HST_15	neighbors_2_hp	neighbors_2_mp	truncatedpagerank_3_mp
siteneighbors_4_hp	assoratitivity_mp	HST_1	siteneighbors_3_hp
STD_83	STD_79	HST_11	eq_hp_mp
	Siteneighbors_3_mp	STD_85	HMG_40

پیوست ۲:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی cfssubseteval و روش جستجوی best first

Indegree_hp	neighbors_3_mp	pagerank_hp	prsigma_hp
prsigma_mp	siteneighbors_3_mp	HSR_2	HST_5
HST_16	HST_11	HST_10	HST_7
HMG_40	HMG_37	HST_24	HST_20
AVG_50	HMG_48	HMG_44	HMG_42
AVG_64	AVG_58	AVG_56	AVG_53
STD_95	AVG_68	AVG_66	AVG_65
			STD_96

پیوست ۳:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی cfssubseteval و روش جستجوی genetic search

eq_hp_mp	assortativity_mp	avgin_of_out_hp	indegree_hp
neighbors_2_hp	neighbors_3_mp	pagerank_hp	prsigma_hp
prsigma_mp	siteneighbors_2_hp	siteneighbors_3_hp	siteneighbors_4_mp
HST_1	truncatedpagerank_4_mp	truncatedpagerank_4_hp	truncatedpagerank_2_hp
HST_10	HST_8	HST_5	HST_2
HST_16	HST_13	HST_12	HST_11
HST_24	HST_20	HST_18	HST_17
HMG_36	HMG_34	HMG_32	HMG_31
AVG_62	AVG_61	HMG_42	HMG_41
AVG_66	AVG_65	AVG_64	AVG_63
STD_80	STD_78	STD_77	AVG_70
STD_87	STD_86	STD_82	STD_81
STD_96	STD_95	STD_90	STD_88
			HMG_37

پیوست ۴:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی cfssubseteval و روش جستجوی greedystepwise

Indegree_hp	neighbors_3_mp	pagerank_hp	prsigma_hp
HST_5	HST_2	siteneighbors_3_mp	prsigma_mp
HST_20	HST_16	HST_11	HST_10
AVG_58	HMG_42	HMG_37	HST_24
AVG_68	AVG_66	AVG_65	AVG_64
HMG_48	HMG_44	STD_96	STD_95
HST_7	AVG_56	AVG_53	AVG_50
			HMG_40

پیوست ۵:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی cfssubseteval و روش جستجوی Linear Forward selection

Neighbors_4_mp	pagerank_hp	prsigma_hp	HST_11
HST_16	HST_18	HST_20	HMG_44
HMG_48	AVG_50	AVG_53	AVG_55
AVG_66	AVG_65	AVG_64	AVG_58
	STD_96	STD_95	AVG_68

پیوست ۶:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی cfssubseteval و روش جستجوی Rank search

avgin_of_out_hp	avgin_of_out_mp	indegree_hp	indegree_mp
neighbors_2_hp	neighbors_2_mp	neighbors_3_mp	neighbors_4_mp
outdegree_hp	outdegree_mp	pagerank_hp	pagerank_mp
siteneighbors_1_mp	siteneighbors_1_hp	prsigma_mp	prsigma_hp
siteneighbors_4_mp	siteneighbors_3_mp	siteneighbors_3_hp	siteneighbors_2_hp
truncatedpagerank_2_mp	truncatedpagerank_2_hp	truncatedpagerank_1_mp	truncatedpagerank_1_hp
truncatedpagerank_4_mp	truncatedpagerank_4_hp	truncatedpagerank_3_mp	truncatedpagerank_3_hp
HST_6	HST_5	HST_2	HST_1
HST_10	HST_9	HST_8	HST_7
HST_14	HST_13	HST_12	HST_11
HST_19	HST_18	HST_17	HST_16

HMG_31	HST_24	HST_22	HST_20
HMG_35	HMG_34	HMG_33	HMG_32
HMG_42	HMG_41	HMG_40	HMG_37
AVG_50	HMG_48	HMG_44	HMG_43
AVG_57	AVG_56	AVG_55	AVG_53
AVG_61	AVG_60	AVG_59	AVG_58
AVG_66	AVG_65	AVG_64	AVG_63
AVG_70	AVG_69	AVG_68	AVG_67
STD_75	STD_74	STD_73	AVG_53
STD_79	STD_78	STD_77	STD_76
STD_83	STD_82	STD_81	STD_80
STD_87	STD_86	STD_85	STD_84
STD_91	STD_90	STD_89	STD_88
STD_95	STD_94	STD_93	STD_92
			STD_96

پیوست۷:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی cfssubseteval و روش جستجوی Scatter search

prsigma_hp	pagerank_hp	neighbors_3_mp	Indegree_hp
HST_7	HST_5	siteneighbors_3_mp	prsigma_mp
HST_21	HST_20	HST_18	HST_16
HMG_40	HMG_37	HMG_33	HST_24
AVG_53	AVG_50	HMG_48	HMG_44
AVG_65	AVG_64	AVG_58	AVG_56
STD_96	STD_95	AVG_68	AVG_66

پیوست۸:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی cfssubseteval و روش جستجو subsetsizeforward selection

Neighbors_4_mp	prsigma_hp	HST_11	HST_16
HST_18	HST_20	HMG_44	HMG_48
AVG_50	AVG_53	AVG_55	AVG_58
AVG_64	AVG_65	AVG_66	AVG_68
	Pagerank_hp	STD_96	STD_95

پیوست۹:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی Consistencysubseteval و روش جستجویbestfirst

Assortativity_mp	indegree_hp	outdegree_mp	prsigma_hp
prsigma_mp	siteneighbors_2_mp	HST_2	HST_7
HST_15	HST_18	HST_19	HST_24
HMG_43	HMG_41	HMG_38	HMG_33
AVG_64	AVG_57	AVG_53	AVG_50
STD_73	AVG_72	AVG_68	AVG_65
		STD_82	STD_75

پیوست۱۰:

فهرست ویژگی های انتخابی با روش انتخاب ویژگی Consistencysubseteval و روش جستجویgenetic search

eq_hp_mp	assortativity_mp	avgout_of_in_hp	indegree_hp
neighbors_2_hp	neighbors_3_hp	outdegree_mp	pagerank_hp
reciprocity_mp	prsigma_mp	prsigma_hp	pagerank_mp
truncatedpagerank_4_mp	truncatedpagerank_1_mp	siteneighbors_2_mp	siteneighbors_1_hp
HST_5	HST_3	HST_2	HST_1
HST_12	HST_11	HST_8	HST_7
HST_20	HST_19	HST_15	HST_14
HMG_33	HMG_27	HST_24	HST_22
HMG_40	HMG_39	HMG_38	HMG_37
AVG_50	HMG_45	HMG_43	HMG_41
AVG_61	AVG_55	AVG_53	AVG_52
AVG_66	AVG_65	AVG_64	AVG_63
STD_73	AVG_72	AVG_71	AVG_67
STD_82	STD_81	STD_80	STD_75
STD_91	STD_86	STD_85	STD_84
siteneighbors_4_hp	reciprocity_hp	STD_96	STD_94
			AVG_59

پیوست ۱۱:

فهرست ویژگیهای انتخابی با روش انتخاب ویژگی GainRatioAttributeEval و روش جستجوی Ranker

HST_24	HMG_48	truncatedpagerank_3_hp	truncatedpagerank_2_hp
truncatedpagerank_2_mp	truncatedpagerank_4_mp	truncated_3_mp	truncatedpagerank_4_hp
truncatedpagerank_1_mp	truncatedpagerank_1_hp	pagerank_mp	pagerank_hp
AVG_66	HST_20	siteneighbors_1_mp	siteneighbors_4_mp
STD_96	HST_5	neighbors_3_mp	neighbors_4_mp
HMG_37	HST_6	HMG_44	STD_95
AVG_68	HST_11	STD_77	AVG_65
STD_86	HMG_40	STD_76	HST_16
STD_80	STD_73	AVG_67	AVG_64
STD_83	AVG_58	STD_79	STD_78
STD_85	STD_82	STD_90	HST_21
STD_88	STD_81	HST_22	AVG_55
STD_84	avgin_of_out_mp	STD_75	HST_13
STD_89	outdegree_hp	AVG_56	indegree_hp
STD_92	HST_12	STD_92	prsigma_hp
STD_87	STD_91	STD_94	HST_12
outdegree_mp	HST_18	AVG_59	siteneighbors_3_mp
AVG_53	STD_93	avgin_of_out_hp	AVG_50
HST_17	AVG_60	HMG_42	AVG_69
HST_10	siteneighbors_1_hp	siteneighbors_2_hp	HST_19
HMG_41	neighbors_2_hp	HMG_34	HST_7
HST_8	HMG-32	neighbors_2_mp	AVG_70
HMG_35	HST_14	HMG_43	HMG_33
indegree_mp	siteneighbors_3_hp	AVG_61	HMG_31
AVG_57	AVG_63	STD_74	prsigma_mp
AVG_62	HST_2	HST_9	HST_1
siteneighbors_4_hp	HMG_46	HMG_45	HMG_26
HMG_36	HST_15	AVG_71	AVG_72
eq_hp_mp	assortativity_mp	HMG_38	siteneighbors_2_mp

پیوست۱۲:

فهرست ویژگیهای انتخابی با روش انتخاب ویژگی InfoGainAttribute و روش جستجوی Ranker

AVG_64	AVG_68	STD_96	STD_95
AVG_67	HST_20	HMG_41	STD_76
AVG_53	HST_19	outdegree_mp	AVG_65
HMG_43	STD_77	STD_73	outdegree_hp
STD_75	AVG_59	AVG_69	AVG_66
STD_86	AVG_50	HMG_33	STD_82
AVG_60	STD_81	AVG_55	HMG_44
AVG_58	HST_18	STD_80	HST_11
HST_16	STD_83	STD_78	STD_79
avgin_of_out_mp	AVG_70	STD_85	prsigma_hp
avgin_of_out_hp	STD_84	HST_21	HST_10
HST_17	HMG_40	STD_88	STD_90
STD_92	AVG_61	HMG_34	HMG_42
HST_9	STD-89	STD_94	HMG_45
STD_93	STD_91	HST_22	STD_87
AVG_63	AVG_56	prsigma_mp	HST_7
AVG_62	STD_74	HMG_46	AVG_71
AVG_72	HST_12	HMG_31	indegree_hp
HST_2	AVG_57	HST_8	HMG_32
HMG_38	HMG_336	HST_13	HMG_26
siteneighbors_3_mp	indegree_mp	HMG_35	HST_14
siteneighbors__2_hp	HST_5	siteneighbors_2_mp	siteneighbors_1_hp
HST_15	neighbors_4_mp	neighbors_3_mp	HST_6
HMG_48	assortativity_mp	neighbors_2_hp	neighbors_2_mp
siteneighbors_3_hp	HST_1	HMG_37	HST_24
truncatedpagerank_3_mp	siteneighbors_4_hp	siteneighbors_1_mp	siteneighbors_4_mp
truncatedpagerank_4_mp	truncatedpagerank_4_hp	pagerank_hp	pagerank_mp
truncatedpagerank_3_hp	truncatedpagerank_1_mp	truncatedpagerank_2_hp	truncatedpagerank_2_mp
		eq_hp_mp	truncatedpagerank_1_hp

پیوست۱۳:

فهرست ویژگیهای انتخابی با روش انتخاب ویژگی Principalcomponent و روش جستجویRanker

eq_hp_mp	assorativity_hp	assorativity_mp	avgin_of_out_hp
indegree_hp	avgout_out_of_in_mp	avgout_of_in_hp	avgin_of_out_mp
neighbors_3_hp	neighbors_2_mp	neighbors_2_hp	indegree_mp
outdegree_hp	neighbors_4_mp	neighbors_4_hp	neighbors_3_mp
prsigma_hp	pagerank_mp	pagerank_hp	outdegree_mp
siteneighbors_1_hp	reciprocity_mp	reciprocity_hp	prsigma_mp
siteneighbors_3_hp	siteneighbors_2_mp	siteneighbors_2_hp	siteneighbors_1_mp
truncatedpagerank_1_hp	siteneighbors_4_mp	siteneighbors_4_hp	siteneighbors_3_mp
truncatedpagerank_3_hp	truncatedpagerank_2_mp	truncatedpagerank_2_hp	truncatedpagerank_1_mp
trustrank_hp	truncatedpagerank_4_mp	truncatedpagerank_4_hp	truncatedpagerank_3_mp
HST_3	HST_2	HST_1	trustrank_mp
HST_7	HST_6	HST_5	HST_4
			HST_8

پیوست۱۴:

فهرست ویژگیهای انتخابی با روش انتخاب ویژگی SymetricalUncertAttributeeval و روش جستجوی Ranker

HST_20	AVG_66	STD_96	STD_95
STD_77	AVG_65	HMG_44	AVG_68
AVG_67	HST_11	AVG_64	STD_76
HMG_40	STD_86	STD_73	HST_16
STD_79	STD_82	STD_78	STD_80
STD_75	AVG_58	HMG_48	HST_24
STD_81	STD_83	AVG_55	outdegree_hp
STD_90	outdegree_mp	HST_21	STD_85
prsigma_hp	STD_88	AVG_59	avgin_of_out_mp
AVG_50	AVG_53	HST_22	STD_84
HST_19	STD_92	STD_89	HST_18
STD_87	AVG_56	AVG_69	STD_94
HST_5	neighbors_3_mp	neighbors_4_mp	STD_91
HMG_41	AVG_60	avgin_of_out_hp	indegree_hp

HST_17	HMG_42	STD_93	HST_12
HMG_34	HST_10	HST_6	HST_13
HST_7	HMG_33	HMG_43	AVG_70
HMG_37	neighbors_3_mp	siteneighbors_1_mp	siteneighbors_4_mp
HMG_31	HST_8	HMG_32	AVG_61
STD_74	HST_14	siteneighbors_1_hp	prsigma_mp
HMG_ 35	HST-9	AVG_63	siteneighbors_2_hp
AVG_62	HMG_45	AVG_57	indegree_mp
HMG_46	HST_2	neighbors_2_hp	neighbors_2_mp
truncatedpagerank_2_mp	truncatedpagerank_4_hp	truncatedpagerank_3_mp	HMG_26
truncatedpagerank_1_mp	truncatedpagerank_1_hp	truncatedpagerank_2_hp	truncatedpagerank_3_hp
AVG_72	pagerank_hp	pagerank_mp	truncatedpagerank_4_mp
HMG_36	HST_1	siteneighbors_3_hp	AVG_71
siteneighbors_4_hp	HST_15	siteneighbors_2_mp	HMG_38
		eq_hp_mp	assorativity_mp

Abstract:

Nowadays, spams is one of the main problems for search engines, because they causes undesirable quality of search results. In recent years, there have been many developments in detection of fake pages, but on the contrary the new spam techniques have emerged. It is necessary to outshine these attacks by improving anti-spam techniques.

There is a common problem in this respect that many of the documents have gained high rating by the search engine while they did not deserve. According to the web expanding and the emergence of new spam techniques, the purpose of this thesis is to investigate new method based on data mining techniques to better identify the spam pages of nonspam pages.

Data mining algorithms and softwares are the tools used in this study. We have provided optimal models by UK2007 standard data sets and Weka software and we tried to introduce a good performance models with reduced features to detect spam pages of nonspam.