
بسم الله الرحمن الرحيم

دانشکده مهندسی کامپیوتر

رساله دکتری مهندسی هوش مصنوعی

تشخیص وب‌هرز با استفاده از ویژگی‌های کیفی

نگارنده: فائزه اصدقی

استاد راهنما:
دکتر علی سلیمانی ایوری

استاد مشاور:
دکتر مرتضی زاهدی

مهر ۱۳۹۹

سپاس بی کران پروردگار یکتا را که هستی مان بخشید و به طریق علم و دانش رهنمونمان شد و به همنشینی رهروان علم و دانش مفتخرمان نمود و خوشه چینی از علم و معرفت را روزیمان ساخت.

یارب دل ما را تو به رحمت جان ده — درد همه را به صابری درمان ده

این بنده چه داند که چه می باید جست — دانندهٔ تویی هر آنچه دانی آن ده

تعهدنامه

اینجانب **فائزه اصدقی** دانشجوی دوره دکتری رشته مهندسی کامپیوتر گرایش **هوش مصنوعی** دانشکده **مهندسی کامپیوتر** دانشگاه صنعتی شاهرود نویسنده پایان نامه **تشخیص وب‌هرز با استفاده از ویژگی‌های کیفی** تحت راهنمائی **دکتر علی سلیمانی** متعهد می‌شوم:

تحقیقات در این پایان نامه توسط اینجانب انجام شده‌است و از صحت و اصالت برخوردار است.

در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورد استفاده استناد شده‌است.

مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده‌است.

کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می‌باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود » و یا « **Shahrood University of Technology** » به چاپ خواهد رسید .

حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان نامه رعایت می‌گردد.

در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده‌است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده‌است.

تاریخ

امضای دانشجو

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج ، کتاب ، برنامه های رایانه‌ای ، نرم افزارها و تجهیزات ساخته شده‌است) متعلق به دانشگاه صنعتی شاهرود می‌باشد . این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود
- استفاده از اطلاعات و نتایج موجود در پایان نامه بدون ذکر مرجع مجاز نمی باشد.

چکیده

موتورهای جستجو بعنوان ابزاری برای دستیابی کارآمد به اطلاعات موجود در وب، نقش انکارناپذیری در دنیای امروز دارند. حضور در میان نتایج موتورهای جستجو و بالاخص حضور در میان چند نتیجه اول از مهمترین اهدافی است که اغلب مدیران وب برای صفحاتشان در نظر می‌گیرند. در این میان "وب‌هرز"ها با هدف فریب موتورهای جستجو و افزایش غیرواقعی رتبه وبسایت‌ها توسعه یافته‌اند. روش‌های مختلفی برای تشخیص "وب‌هرز"ها ارائه شده‌اند، اما تغییر مداوم و روزآمد شدن تکنیک‌های فریب الگوریتم‌های رتبه‌بندی، از مهم‌ترین چالش‌های پیش روی این روشها به شمار می‌روند. از طرف دیگر نامتوازن بودن داده‌ها و کم بودن تعداد صفحات هرز نسبت به صفحات غیرهرز نیز کار پایش و تشخیص خودکار "وب‌هرز"ها را دشوار می‌سازد.

در این رساله ضمن ارائه و معرفی ویژگی‌هایی با قابلیت جعل‌پذیری پایین‌تر، مدلی جهت تشخیص کارآمد "وب‌هرز"ها پیشنهاد شده‌است. در این مدل ابتدا به منظور کاهش ابعاد داده و افزایش سرعت تشخیص، الگوریتم انتخاب ویژگی به نام Smart-BT توسعه یافته و برای مجموعه داده‌های نامتوازن خاص منظوره شده‌است. سپس تعدادی ویژگی جدید جهت افزایش نرخ تشخیص وب‌هرز شامل تعیین میزان پیوستگی با استفاده از نرخ حضور ضمایر اشاره در متن قابل رویت صفحه، نحوه توزیع موضوعات در صفحه، میانگین نرخ پیوستگی نویسه‌های الفبایی در توکن‌های مسیر و نرخ استفاده از برخی برچسب‌های HTML مثل iftame، div، link و a معرفی شده‌اند. در نهایت با استفاده از مفهوم یابنده‌ها و سلول‌های حافظه در سیستم ایمنی مصنوعی، روشی برای تشخیص صفحات هرز ارائه شده‌است.

نتایج حاصل از مدل پیشنهادی در تشخیص صفحات وب‌هرز بر روی مجموعه داده WEBSPAM-UK، نشان‌دهنده دقت بالانس شده به میزان ۰/۸۷ با استفاده از ۴۸ ویژگی است که نسبت به بهترین نتیجه گزارش شده تا کنون ۱۶٪ افزایش دقت و ۶۵٪ کاهش تعداد ویژگی داشته است.

کلمات کلیدی: موتور جستجو، وب‌هرز، ویژگی مبتنی بر پیوند، ویژگی مبتنی بر محتوا، انتخاب ویژگی، انسجام متنی، مدلسازی موضوعی، سیستم ایمنی مصنوعی.

مقالات مستخرج از پایان نامه:

الف: مقالات ISI و علمی- پژوهشی

[۱] Asdaghi, F., Soleimani, A. (2018). An effective feature selection method for web spam detection. *Knowledge-Based Systems*, 166, 198-206. doi: 10.1016/j.knosys. 2018.12.026.

[2] Asdaghi, F., Soleimani, A., Zahedi, M. (2020). A Novel Set of Contextual Features for Web Spam Detection. *International Journal of Nonlinear Analysis and Applications*, 11(1), 321-339. Doi: 10.22075/ijnaa.2020.4297

ب: مقالات کنفرانسی

[۳] اصدقی ف. و سلیمانی ع.، (۱۳۹۴) "بررسی تأثیر روش‌های طبقه بندی بر میزان تشخیص صفحات وب اسپم" سومین کنفرانس بین المللی پژوهشهای کاربردی در مهندسی کامپیوتر و فن‌آوری اطلاعات، ص ۸۶، تهران.

[۴] سلیمانی ع. اصدقی ف.، (۱۳۹۶)، "بررسی تأثیر کاهش ویژگی بر افزایش نرخ دقت تشخیص صفحات وب‌هرز"، سومین کنفرانس پردازش سیگنال و سیستم‌های هوشمند ، ص ۱۳۴، شاهرود.

[۵] اصدقی ف. سلیمانی ع.، (۱۳۹۶)، "افزایش دقت تشخیص صفحات وب‌هرز با استفاده از سیستم ایمنی مصنوعی"، سومین کنفرانس محاسبات تکاملی و هوش جمعی، ص ۱۱۴، بم.

فهرست مطالب

چکیده ..... و
۱- شرح مسئله ۱
۱-۱- مقدمه ۲
۱-۲- موتور جستجو ۲
۱-۲-۱- مرتب‌ساز ۴
۱-۳- وب‌هرز و انواع آن ۶
۱-۳-۱- هرزنامه‌های محتوامحور ۸
۱-۳-۲- هرزنامه‌های پیوندمحور ۱۱
۱-۳-۳- هرزنامه‌های مبتنی بر پنهان‌سازی صفحه ۱۳
۱-۴- چالش‌های پیش رو ۱۴
۱-۴-۱- استخراج ویژگی ۱۴
۱-۴-۲- انتخاب ویژگی ۱۴
۱-۴-۳- طبقه‌بندی کننده مناسب ۱۵
۱-۵- نوآوری‌ها و دستاوردهای رساله ۱۵
۱-۶- ساختار رساله ۱۶
۲- مرور کارهای گذشته ۱۹
۲-۱- مقدمه ۲۰
۲-۲- روش‌های مبتنی بر گراف وب ۲۰
۲-۲-۱- الگوریتم PageRank ۲۱
۲-۲-۲- الگوریتم HITS ۲۱
۲-۲-۳- الگوریتمهای مبتنی بر انتشار برچسب ۲۳
۲-۲-۴- الگوریتمهای هرز پیوند و وزندهی مجدد ۲۷
۲-۲-۵- الگوریتمهای مبتنی بر پالایش برچسب ۲۹
۲-۳- روشهای متمرکز بر طبقه‌بندی کننده ۳۰
۲-۴- روشهای متمرکز بر ویژگیها ۳۳
۲-۵- روشهای ترکیبی ۴۱

۲-۶- جمع بندی ۴۲
۳- تئوری‌ها و روش پیشنهادی ۴۵
۳-۱- مقدمه ۴۶
۳-۲- استخراج ویژگی ۴۶
۳-۲-۱- ویژگی‌های مبتنی بر ساختار نشانی اینترنتی ۴۷
۳-۲-۲- ویژگی‌های مبتنی بر برچسب‌های کد HTML صفحات وب ۵۱
۳-۲-۳- ویژگی‌های مبتنی بر انسجام متن ۵۳
۳-۳- انتخاب ویژگی ۶۰
۳-۳-۱- روش‌های جستجو ۶۱
۳-۳-۲- ارزیابی ویژگی ۶۵
۳-۳-۳- روش پیشنهادی برای انتخاب ویژگی ۶۹
۳-۳-۴- پیچیدگی زمانی و مکانی ۷۲
۳-۴- طبقه بندی ۷۴
۳-۴-۱- سیستم ایمنی مصنوعی ۷۸
۳-۴-۲- روش پیشنهادی جهت طبقه بندی صفحات وب بر اساس سیستم ایمنی مصنوعی ۸۱
۳-۵- مدل پیشنهادی ۸۴
۳-۵-۱- پیچیدگی زمانی ۸۵
۴- نتایج ۸۷
۴-۱- مقدمه ۸۸
۴-۲- مجموعه داده ۸۸
۴-۳- فرض‌های مسئله ۹۰
۴-۴- معیارهای ارزیابی ۹۰
۴-۵- نتایج به دست آمده ۹۳
۴-۵-۱- مقایسه نحوه تشخیص صفحات وب‌هرز توسط طبقه‌بندهای رایج ۹۳
۴-۵-۲- بررسی تاثیر تعداد ویژگی‌ها بر عملکرد طبقه‌بند ۹۹
۴-۵-۳- بررسی نحوه تاثیر ویژگی‌های معرفی شده بر نرخ تشخیص وب‌هرز ۱۰۴
۵- جمع‌بندی و کارهای آتی ۱۱۱
۵-۱- مقدمه ۱۱۲

۵-۲- نتایج و دستاوردها ۱۱۳
۵-۳- کارهای آتی ۱۱۴
منابع ۱۱۵

فهرست شکل‌ها

شکل ۱-۱ فرآیند ارائه سؤال و دریافت پاسخ ۳
شکل ۱-۲ رده‌بندی سطح بالای هرزنامه‌ها ۶
شکل ۱-۳ شمایی از یک مزرعه پیوند رایج ۱۲
شکل ۲-۱ مفهوم قطب و قدرت ۲۲
شکل ۲-۲ انواع ویژگی‌ها ۳۳
شکل ۲-۳شمای کلی مدل اعتماد به محتوا..... ۳۶
شکل ۳-۱ توزیع چگالی احتمال برای ویژگی‌های مبتنی بر URL ۴۸
شکل ۳-۲ توزیع چگالی احتمال ویژگی‌های مبتنی بر کدهای HTML ۵۱
شکل ۳-۳ شمای گرافیکی الگوریتم LDA ۵۶
شکل ۳-۴- شبه کد الگوریتم پیشنهادی Smart-BT ۷۲
شکل ۳-۵- فلوچارت روش طبقه‌بندی پیشنهادی ۸۳
شکل ۳-۶ نمودار مدل پیشنهادی ۸۵
شکل ۴-۱ جدول درهم‌ریختگی ۹۰
شکل ۴-۲ روند تغییرات میزان IBA بر اثر تغییر نوع ویژگی‌های استفاده شده جهت طبقه‌بندی ۹۹
شکل ۴-۳ تاثیر تعداد موضوعات بر میزان پیوستگی مدل LDA ۱۰۸

فهرست جدول‌ها

جدول ۲-۱ لیست ویژگی‌های جدید معرفی شده برای شناسایی وب‌هرز ۴۰
جدول ۳-۱ ویژگی‌های مبتنی بر پیوستگی ۵۸
جدول ۳-۲ نگاشت AIS به مسئله تشخیص صفحات وب‌هرز ۸۱
جدول ۴-۱- نرخ تشخیص وب اسپم با ویژگی‌های مبتنی بر محتوا. ۹۴
جدول ۴-۲ نرخ تشخیص وب اسپم با ویژگی‌های مبتنی بر پیوند. ۹۵
جدول ۴-۳ نرخ تشخیص وب اسپم با ویژگی‌های مبتنی بر پیوند تبدیل یافته. ۹۶
جدول ۴-۴ نرخ تشخیص وب اسپم با ترکیب ویژگی‌های مبتنی بر محتوا و مبتنی بر پیوند. ۹۷
جدول ۴-۵ نرخ تشخیص وب اسپم با ترکیب ویژگی‌های مبتنی بر محتوا و پیوند تبدیل یافته. ۹۷
جدول ۴-۶ نرخ تشخیص وب اسپم با ترکیب هر سه دسته ویژگی. ۹۸
جدول ۴-۷ میزان IBA حاصل از روش‌های مختلف انتخاب ویژگی ۱۰۰
جدول ۴-۸ مقایسه نتایج استفاده از الگوریتم Smart-BT برای کاهش ویژگی و تاثیر آن بر نرخ تشخیص ۱۰۱
جدول ۴-۹- لیست ویژگی‌های انتخاب شده ۱۰۲
جدول ۴-۱۰ توزیع انواع ویژگی‌های انتخاب شده در حالت‌های مختلف ۱۰۳
جدول ۴-۱۱ مقایسه نتایج به‌دست آمده ۱۰۳
جدول ۴-۱۲ مقایسه نتایج حاصل از طبقه‌بندی داده با استفاده از ویژگی‌های مختلف ۱۰۵
جدول ۴-۱۳- نتایج به دست آمده توسط سیستم ایمنی مصنوعی در مقایسه با سایر روش‌های طبقه‌بندی ۱۰۶
جدول ۴-۱۴ مقایسه نتایج به دست آمده با دیگران ۱۰۷
جدول ۴-۱۵ لیست ویژگی‌های انتخاب شده‌از بین ویژگی‌های معرفی شده ۱۰۷

شرح مسئله

۱

وب‌هرز و انواع آن
روش‌های تشخیص و چالش‌های موجود
نوآوری و دستاوردها

۱-۱- مقدمه

امروزه با توجه به رشد اطلاعات در وب[1]، موتورهای جستجو[2] به عنوان درگاهی برای ورود به آن مورد توجه قرار گرفته‌اند. موتور جستجو نرم‌افزاری است که با دریافت سؤال از کاربر، اطلاعات مورد نظر وی را به عنوان پاسخ، در اختیار او قرار می‌دهد. تحقیقات صورت گرفته نشان می‌دهد که تقریباً ۶۰٪ کاربران، تنها از پنج پیوند اولیه‌ی صفحه نخست نتایج، بازدید می‌کنند [1]. درنتیجه حضور صفحه‌ای در نتایج بالای موتورهای جستجو به معنای بازدیدکننده بیشتر و در نتیجه درآمد بیشتر است. بنابراین افزایش رتبه سایت در فهرست نتایج موتورهای جستجو امری بسیار مهم در بحث تجارت الکترونیک است.

روش قانونی برای افزایش رتبه سایت‌ها در فهرست نتایج موتورهای جستجو افزایش کیفیت صفحات است، اما این روند زمانبر و پرهزینه است. راهکار دیگر استفاده از روش‌های غیرقانونی و غیراخلاقی به‌منظور فریب الگوریتم‌های رتبه‌دهی[3] و افزایش رتبه است. صفحاتی که به‌منظور افزایش رتبه خود از این‌گونه روش‌های ناعادلانه استفاده می‌کنند وب‌هرز[4] نامیده می‌شوند. ازآنجاکه در این رساله قصد داریم مدلی را برای شناسایی این‌گونه صفحات ارائه دهیم، ابتدا مطالبی در مورد ساختار و معماری موتورهای جستجو و عوامل موثر در رتبه وب سایت‌ها بیان کرده، سپس به شرح چالش‌های موضوع، دستاوردهای رساله و در نهایت ساختار آن می‌پردازیم.

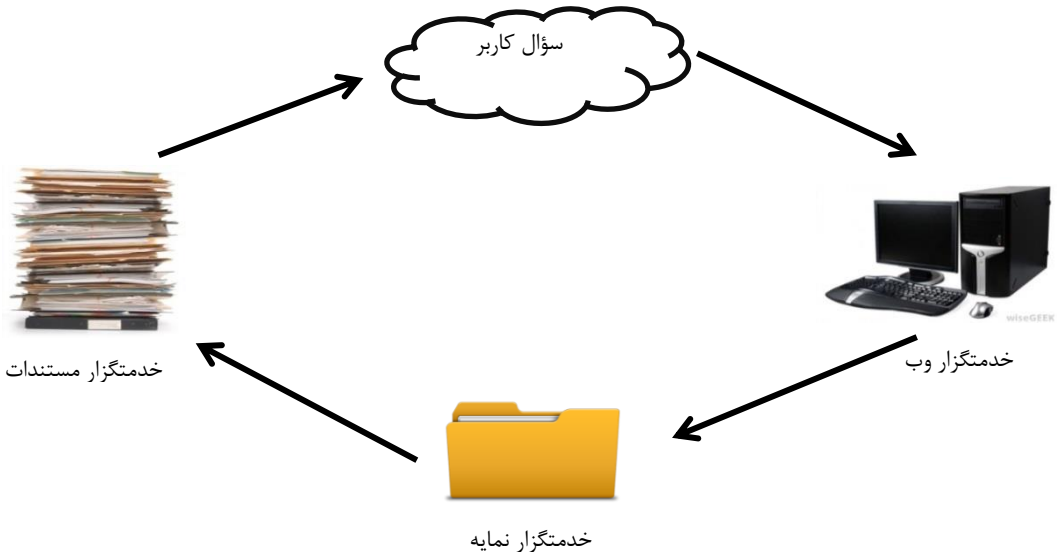
۱-۲- موتور جستجو

موتور جستجو نرم‌افزاری است که با دریافت سؤال[5] از کاربر، متون مرتبط با آن را در اختیار وی قرار می‌دهد. منظور از سؤال، کلیدواژه‌هایی است که کاربر به‌وسیله آن‌ها تلاش کرده‌است نیاز خود را

1. Web
2. Search Engine
3. Ranking
4.Web Spam
5 .Query

دقیق‌تر بیان کند. در واقع کار اصلی یک موتور جستجو، جمع‌آوری و حفظ سندهای قابل دسترسی در یک قالب بهینه، به‌دست‌آوردن مشابه‌ترین سند به سؤال کاربر و نمایش سندها به ترتیب میزان مرتبط بودن است.

موتور جستجو برای پیگیری نیاز اطلاعاتی کاربر، ابتدا پرسش را از کاربر دریافت می‌کند. این پرسش شامل کلیدواژه‌هایی است که وی به دنبال اطلاعاتی در آن زمینه است. همچنین امکان دارد واژه‌های ذخیره‌شده‌ای که به‌وسیله موتور جستجو برای ارائه دقیق‌تر به نیاز اطلاعاتی کاربر تعریف شده‌اند، به سؤال کاربر پیوست شوند. مراحل یادشده در شکل ۱-۱ **Error! Reference source not found.** نشان داده شده‌است.



شکل ۱-۱ فرآیند ارائه سؤال و دریافت پاسخ [2]

پس از این مرحله، پرسش کاربر به خدمتگزار نمایه [1] ارسال می‌شود. پس از انجام مقایسه لازم از سوی خدمتگزار، مستندات مربوط به پرسش کاربر، مشخص می‌شوند. تعدادی از مشخصات مهم این سندها برای آگاهی کاربران در اختیار خدمتگزار سندها قرار داده می‌شود. به طور کلی یک موتور جستجو از سه قسمت اساسی تشکیل شده‌است [3]:

1. Index server

- خزشگر[1]؛ اسناد موجود در وب را جمع‌آوری می‌کند.
- نمایه‌ساز[2]؛ سندهای پویش‌شده به‌وسیله خزشگر را به نمایه‌های قابل استفاده تبدیل می‌کند.
- مرتب‌ساز[3]؛ رتبه‌بندی مستندات به‌دست‌آمده را انجام می‌دهد و در واقع قلب موتور جستجو است

۱-۲-۱- مرتب‌ساز

مرحله نخست در استخراج اطلاعات کشف شباهت میان نیاز اطلاعاتی کاربر و اطلاعات موجود است. نمایش نتایج به صورت رتبه‌بندی‌شده یکی از مواردی است که برای کاربر اهمیت زیادی دارد. یکی از عمده دلایلی که نشان‌دهنده تفاوت رتبه‌بندی نتایج موتورهای جستجو با یکدیگر است، الگوریتم‌های رتبه‌دهی[4] آن‌ها است. این الگوریتم‌ها تعیین می‌کند که آیا یک صفحه وب، واقعی و دارای اطلاعات ارزشمند است و یا وب‌هرز به شمار می‌رود. هر موتور جستجو از روش‌های خاصی برای این کار استفاده می‌کند که معمولاً محرمانه هستند اما موارد خاصی وجود دارد که تمام الگوریتم‌های موتور جستجو در آن مشترک هستند که در ادامه به آن خواهیم پرداخت.

۱-۲-۱-۱- مرتبط بودن

مرتبط بودن صفحه با کلیدواژه‌ها از اولین مواردی است که یک الگوریتم موتور جستجو آن را بررسی می‌کند. تشخیص وجود کلمات کلیدی در هر صفحه وب و یا چگونگی استفاده از این کلمات کلیدی و ارتباط آنها با محتوای صفحه را الگوریتم مربوطه تعیین خواهد کرد. علاوه بر این موضوع، مکان قرارگیری کلمات کلیدی نیز یک فاکتور مهم برای الگوریتم موتورهای جستجو محسوب می‌شود . صفحات وبی که کلمات کلیدی را در عنوان‌ها و همچنین در سرتاسر یا چند خط اول متن، استفاده

1. Web Crawler
2. Indexer
3. Ranker
4. Ranking Algorithms
5. Relevancy

می‌کنند نسبت به صفحات وبی که این موارد را رعایت نکرده‌اند در نتیجه رتبه بندی بالاتری قرار خواهند گرفت. حضور متناوب کلمات کلیدی نیز برای الگوریتم‌ها حائز اهمیت می‌باشد. چنانچه کلمات کلیدی به طور مرتب ، طبیعی و متناوب ظاهر شوند، در صورتی که این حضور جهت بازی با کلمات و فریب دادن ربات‌ها نباشد ، آن وب سایت رتبه بندی بالاتری را از آن خود خواهد کرد .

۱-۲-۱-۲- فاکتورهای ویژه[1]

بخش دوم الگوریتم‌های موتور جستجو، فاکتورهای ویژه هستند که هر موتور جستجو را از دیگر موتور جستجوی متفاوت می‌سازد. هر موتور جستجو دارای الگوریتم‌های منحصر به فرد است و فاکتورهای ویژه این الگوریتم‌ها سبب ایجاد تفاوت در نتیجه ی جستجوی موتورهای جستجو همچون گوگل، یاهو و MSN می‌شود. یکی از رایج ترین فاکتورهای ویژه، تعداد صفحات ایندکس شده در یک موتور جستجوگر است. به بیان دیگر یک وبمستر ممکن است صفحات بیشتری از سایت خود را در موتورهای جستجو ایندکس کند و یا به صورت متناوب ایندکس کردن صفحات را انجام دهد، اما این میتواند نتایج مختلفی را برای هر موتور جستجو داشته باشد. برخی از موتورهای جستجو سایت‌ها را برای این موضوع به صورت اسپم شناخته و سایت را جریمه می‌کند و به اصطلاح دچار پنالتی می‌شود، در حالی که برای دیگر موتورهای جستجو این موضوع اهمیت ویژه‌ای ندارد.

۱-۲-۱-۳- فاکتورهای خارج از صفحه[2]

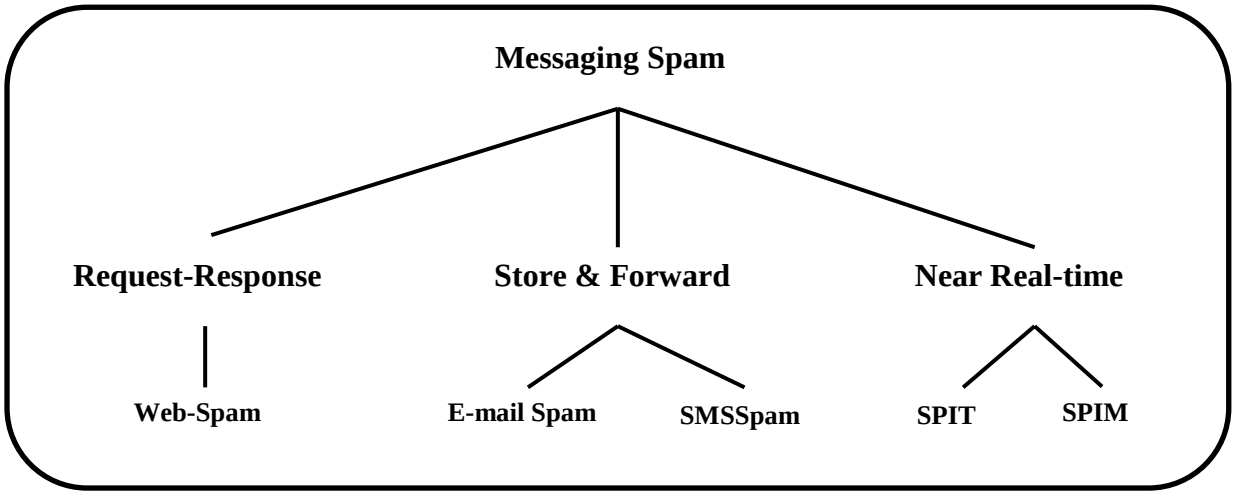
بخش دیگری از الگوریتم‌های موتورهای جستجو که هنوز برای هر موتور جستجو منحصر به فرد است، فاکتورهای خارج از صفحه می‌باشد. فاکتورهای خارج از صفحه مواردی همچون لینک و بک لینک‌های صفحه و اعتبار یک سایت در میان دیگر سایت‌ها می‌باشند. کلیک‌های متناوب و لینک‌ها می‌تواند شاخصی باشد که میزان ارتباط یک صفحه وب را با کاربران و بازدیدکنندگان واقعی و ارزش

[1]. Individual Factors
[2]. Off Page Factors

سایت از دید آنها را نمایان کند که این شاخص باعث می‌شود که یک الگوریتم رتبه بندی صفحات وب را مشخص کند. انجام صحیح فاکتورهای خارج از صفحه برای کارشناسان وب سخت تر است، اما می‌توانند تاثیر زیادی بر رتبه وب سایت بر اساس الگوریتم‌های موتور جستجو داشته باشند.

۱-۳- وب‌هرز و انواع آن

در سال‌های اخیر از کلمه هرز[1] برای اشاره به پیغام‌های انبوه و ناخواسته استفاده شده‌است. رایج-ترین شکل هرز، هرزنامه[2] است. هرزنامه، سوءاستفاده از سیستم انتقال پیام (ازقبیل پست الکترونیک و پیامک) است و به پیام‌های گروهی و ناخواسته اطلاق می‌شود که در ساختار درختی شکل ۱-۲ در دسته «ذخیره و ارسال» قرار می‌گیرد در حالی که وب‌هرزها در دسته «درخواست و پاسخ» قرار دارند.



شکل ۱-۲ رده‌بندی سطح بالای هرزنامه‌ها [3]

وب‌هرزها همزمان با ظهور موتورهای جستجوی تجاری پدیدار شدند. در سال ۱۹۹۵، Lycos نخستین موتور جستجوی تجاری وارد دنیای وب شد. از همان زمان مبارزه با وب‌هرزها تحت عنوان هرزآگهی[3] و با استفاده از روش‌های یادگیری ماشین تبدیل به یکی از مباحث مورد علاقه مجامع

1. Spam
2. Email spam
3. Spamdexing

دانشگاهی شد [4]. از سال ۲۰۰۵، کارگاه‌های آموزشی AIRWeb به محلی برای تبادل اطلاعات در این زمینه تبدیل شد [5].

مطابق تعریف ارائه‌شده به‌وسیله مرجع [6]، وب‌هرز به هر فعالیت غیرقانونی انجام‌شده‌از سوی افراد به‌منظور افزایش رتبه صفحات وب اطلاق می‌شود. مرجع [7] نیز وب‌هرز را به صورت رفتاری برای فریب موتورهای جستجو تعریف کرده‌است. آن‌ها باعث کاهش کیفیت نتایج جستجو و درنتیجه اتلاف وقت کاربر می‌شوند. افزایش تعداد این صفحات، افزایش تعداد صفحات جستجوشده به‌وسیلهٔ خزشگرها و مرتب‌شده به‌وسیلهٔ نمایه‌گزارها را در پی دارد که این امر موجب اتلاف منابع موتور جستجو و افزایش زمان پاسخگویی به کاربر می‌شود.

مورد دیگری که باعث اهمیت موضوع وب‌هرز می‌شود این است که گاهی به عنوان وسیله‌ای برای گسترش بدافزار[1]، دسترسی به محتوای ممنوعه و حملات فیشینگ[2] به کار گرفته می‌شوند. به عنوان مثال [8] حدود صد میلیون صفحه با استفاده از الگوریتم PageRank رتبه‌دهی و مشخص شد؛ ۱۱ صفحه از ۲۰ نتیجه ارائه‌شده به‌وسیله موتورهای جستجو، سایت‌های هرزه‌نگاری[3] بودند که به سبب تغییر محتوا و پیوند رتبه بالایی گرفته بودند. این امر موجب هدردادن مقادیر زیادی از منابع محاسباتی و ذخیره‌سازی به‌وسیله شرکت سازنده موتور جستجو می‌شود. طی تحقیقی در سال ۲۰۰۵ [9]، تخمین زده شد وب‌هرزها سالانه حدود ۵۰ میلیون دلار باعث اتلاف منابع مالی می‌شوند. این مقدار در تحقیقی مشابه [10] که در سال ۲۰۰۹ صورت گرفت به ۱۳۰ میلیون دلار افزایش یافت. ازجمله دلایل رشد سریع تعداد وب‌هرزها می‌توان به ساده شدن ابزار ایجاد وب (ازقبیل ویکی وب‌های رایگان، بسترهای بلاگ نویسی و غیره) و کاهش هزینه نگهداری وب‌سایت اشاره کرد. این امر سبب

1. Malware
2. Phishing
3. Pornography

شده‌است وب‌هرزها تکامل یافته، انواع جدیدی از آن‌ها ایجاد شود به گونه‌ای که الگوریتم‌های قدیمی برای شناسایی آن‌ها کارساز نباشند.

با رواج پدیده وب‌هرز و عدم وجود ابزاری برای تشخیص و کنترل آن‌ها، کیفیت جستجو پایین می‌آید. درنتیجه کاربران باید برای دست‌یافتن به نتایج دلخواهشان از صفحات بیشتری بازدید کنند؛ ازاین‌رو تشخیص وب‌هرزها برای موتورهای جستجو از اهمیت بسزایی برخوردار بوده، در بازار رقابتی با سایر موتورهای جستجو جزء اولویت‌های کاری آن‌ها محسوب می‌شوند. تحقیقات متعددی که در این زمینه صورت گرفته ([11] [12] [13]) نشان می‌دهد که میزان وب‌هرزها بین ۶ تا ۲۲ درصد است. دلیل گستردگی این بازه در این است که تاکنون تحقیقات جامعی روی کل صفحات وب صورت نگرفته و آمار موجود مربوط به حوزه‌های تخصصی است که به طور جداگانه مورد بررسی قرار گرفته‌اند. به عنوان مثال [11] نشان می‌دهد که ۲۲٪ میزبان‌ها هرز هستند این در حالی است که [12] این میزان را ۱۶/۵٪ تخمین زده است. از لحاظ تفکیک زبانی در [13] نشان داده شده‌است که ۱۳/۸٪ صفحات انگلیسی زبان، ۹٪ صفحات ژاپنی، ۲۲ ٪ صفحات آلمانی و ۲۵٪ صفحات فرانسوی زبان، هرز هستند. همچنین از لحاظ دامنه، ۷۰٪ صفحات با دامنه biz.* و ۲۰٪ صفحات com.* هرز هستند. محققان وب‌هرزها را به سه دسته اصلی محتوامحور[1]، پیوندمحور[2] و مبتنی بر پنهان‌سازی صفحه[3] تقسیم می‌کنند که در ادامه توضیح خواهیم داد.

۱-۳-۱- هرزنامه‌های محتوامحور

هرزمحتوا[4] نخستین و گسترده‌ترین شکل وب‌هرز است. دلیل گستردگی این شکل از وب‌هرز در این حقیقت نهفته است که موتورهای جستجو برای رتبه‌دهی به صفحات وب از مدل‌های بازیابی اطلاعات مبتنی بر محتوای صفحه ازقبیل مدل فضای بردار [14]، BM25 [15] و مدل‌های آماری زبان

[1]. Content Based
[2]. Link Based
[3]. Cloacking
[4]. Content Spam

[16] استفاده می‌کنند. به همین دلیل هرزفرست‌ها[1] نقاط ضعف این مدل‌ها را تحلیل کرده و از آن‌ها سوءاستفاده می‌کنند. فرض کنید موتور جستجو طبق رابطه ۱-۱، از رتبه‌دهی TFIDF استفاده می-کند.

$$TFIDF(q.p) = \sum_{t \in q \wedge t \in p} TF(t).IDF(t) \qquad (۱-۱)$$

در این صورت اگر q کلمه مورد جستجو، p یک صفحه و t یک عبارت باشد، آنگاه هرزفرست‌ها می‌توانند با بالا بردن تعداد دفعات تکرار عباراتی که در صفحه مشاهده می‌شوند (TF)، رتبه خود را بالا ببرند. گفتنی است که IDF عبارتِ t، نسبت تعداد اسناد حاوی این عبارت به کل تعداد اسناد موجود است.

از دیدگاه‌های زیادی امکان طبقه‌بندی هرز محتوا وجود دارد که در اینجا آن‌ها را به پنج زیرگروه تقسیم می‌کنیم [6].

- **هرز بدنه**[2]

یکی از محبوب‌ترین و آسان‌ترین روش‌های هرز است. در این روش عبارات هرز در بدنه صفحه قرار می‌گیرند. به عنوان نمونه اگر یک هرزفرست قصد دستیابی به رتبه بالایی با جستجوی یک سری کلمات محدود و از پیش تعیین‌شده داشته باشد، کافی است آن کلمات و مشتقات آن را در صفحه تکرار کند. از سوی دیگر اگر هدف هرزفرست بالا رفتن رتبهٔ صفحه با هر جستجویی باشد کافی است تعداد زیادی کلمه تصادفی را در صفحه قرار دهد. برای پنهان ماندن این کار از سوی کاربر نیز کافی است رنگ نوشته و پس‌زمینه یکسان در نظر گرفته شود. در این صورت تنها ماشین قادر به تشخیص آن خواهد بود.

[1] Spamers
[2]. Body Spamming

- **هرز عنوان**[1]

برخی موتورهای جستجو وزن بیشتری برای عنوان اسناد قائل هستند. به همین دلیل هرزفرست‌ها عبارات هرز را در عنوان صفحه قرار می‌دهند.

- **هرز فرابرچسب**[2]

توضیحات فرابرچسب‌های HTML[3] این امکان را برای طراحان صفحات فراهم می‌کند که توضیحات کوتاهی راجع به صفحه قرار دهند. اگر کلمات نامربوط در این مکان قرار گرفته و الگوریتم-های موتورهای جستجو بر اساس این توضیحات صفحه را مورد بررسی قرار دهند، آنگاه این صفحات به سبب حضور کلمات نامربوط رتبه بالاتری دریافت می‌کنند. هرچند امروزه این برچسب چندان مورد اهمیت نیست.

- **هرز آدرس اینترنتی**[4]

برخی الگوریتم‌ها از عبارات موجود در URL[5] سایت به‌منظور محاسبه رتبه استفاده می‌کنند. به همین دلیل هرزفرست‌ها، آدرس‌های طولانی شامل کلمات عبارت جستجوشده برای صفحات ایجاد می‌کنند. به عنوان مثال اگر هرزفرستی بخواهد رتبه صفحه خود را در صورت جستجوی عبارت "cheap acer laptops" افزایش دهد، URLای به شکل cheap-laptops.com/cheap-laptops/acer-laptops.html ایجاد می‌کند.

- **هرز برچسب پیوند**

بخش برچسب یک پیوند، حاوی خلاصه‌ای از متن صفحه‌ای است که به آن اشاره می‌کند؛ بنابراین هرزفرست‌ها با قراردادن عبارات دلخواه خود (که معمولاً ارتباطی با محتوای صفحه ندارد) می‌توانند با

---

1. Title Spamming
2. Meta tag
3. Hyper Text Markup Language
4. URL Spamming
5. Uniform Resource Locator

فریب الگوریتم‌هایی که برای عبارات این بخش ارزش بیشتری قائل می‌شوند، رتبه خود را افزایش دهند.

۱-۳-۲- هرزنامه‌های پیوندمحور

دو دسته عمده از هرز پیوندها وجود دارد: هرز پیوندهای خروجی[۱] و هرز پیوندهای ورودی[۲]. هرز پیوندهای خروجی آسان‌ترین و ارزان‌ترین روش افزایش رتبه صفحات وب هستند. ازآنجاکه هرزفرست مالک صفحه وب خود بوده و امکان تغییر آن را دارد، به‌راحتی می‌تواند پیوند هر صفحهٔ دلخواه را در صفحه خود قرار دهد یا حتی محتوای سایت‌هایی ازقبیل DMOZ[۳] و Yahoo! Directory[۴] _ که حاوی فهرستی از آدرس صفحات اینترنتی هستند _ را در صفحه خود کپی کند. این روش الگوریتم-های رتبه‌دهی همچون HITS[۵] را نشانه گرفته و در پی افزایش رتبه قطب[۶] خود است.

هرزفرست‌ها در هرز پیوندهای ورودی الگوریتم رتبه‌دهی PageRank را نشانه گرفته و درنتیجه سعی در افزایش تعداد پیوندهای ورودی دارند. در اینجا بسته به اینکه هرزفرست مالک صفحه باشد یا تنها به آن دسترسی داشته باشد، روش‌ها متفاوت است. در صفحاتی که هرزفرست مالک آن‌هاست با توجه به کنترل آن بر همه صفحات، برای پیاده‌سازی روش‌های مختلف حق انتخاب دارد. این روش‌ها در چند دسته کلی زیر قرار می‌گیرند:

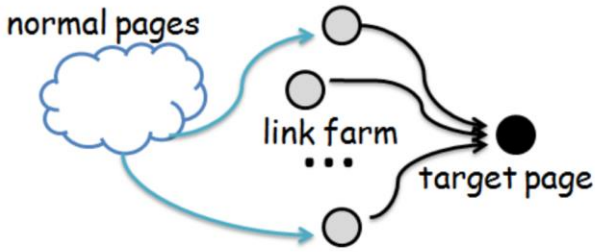
۱. ایجاد مزرعه پیوند[۷]

مزرعه پیوند مجموعه‌ای از صفحات یا سایت‌هایی است که به یکدیگر متصل هستند. یک هرزفرست برای افزایش رتبهٔ خود می‌تواند مزرعه پیوند مورد نظر خود را به هر شکلی که دوست دارد،

---

1. Output Link Spam
2. Input Link Spam
3. Dmoz.org
4. http://dir.yahoo.com/
5. hyperlink-induced topic search
6. hub
7. Link farm

پیاده‌سازی کند. یک شکل رایج از مزرعه پیوند شکل ۱-۳ است که به آن مزرعه ظرف عسل[1] هم می-گویند. در این حالت هرزفرست یک صفحه به‌ظاهر غیرهرز و معمولی می‌سازد، سپس پیوندهایی به صفحات هرز را در آن قرار می‌دهد. این کار باعث می‌شود درجه قدرت[2] سایت بالا رفته، رتبه بالاتری به‌وسیله الگوریتم PageRank به آن تعلق گیرد.



شکل ۱-۳ شمایی از یک مزرعه پیوند رایج [17]

شکل تهاجمی‌تر استفاده از ظرف عسل، ربایش[3] است. در این روش ابتدا یک سایت مشهور یا قابل اعتماد هک می‌شود. سپس در آن پیوندهایی به صفحات هرز قرار گرفته و آن سایت بخشی از مزرعه پیوند هرزفرست می‌شود.

۲. تبادل پیوند

صاحبان وب‌سایت‌ها به یکدیگر کمک کرده و بر روی صفحات خود، پیوندی از سایت دیگری قرار می‌دهند. این کار به آن‌ها کمک می‌کند سایت‌های مختلفی به آن‌ها رجوع کرده و درنتیجه رتبه آن‌ها بالاتر رود.

۳. خرید دامنه‌های منقضی‌شده

هرزفرست‌ها به‌منظور کاهش مدت‌زمان لازم برای ارتقای رتبه یک مزرعه پیوند، بسیار علاقه‌مند به خرید دامنه‌های منقضی‌شده و سوءاستفاده از مقبولیت قبلی آن‌ها هستند. این امر از آنجا ناشی می-

[1]. Honeypot farm
[2]. Authority
[3]. Hijacking

شود که موتورهای جستجو برای مدیریت منابع خود، فهرست‌های خود را دیربه‌دیر به‌روزرسانی و بازپیمایی می‌کنند. به همین دلیل ممکن است تا مدت‌ها پس از اتمام اعتبار یک سایت، آن را معتبر و قابل اعتماد بدانند که همین امر راه را برای سوءاستفاده هرزفرست‌ها باز می‌کند.

هرزفرست‌ها در صفحاتی که هرزفرست مالک آن‌ها نیست (صفحاتی ازقبیل بلاگ‌ها یا ویکی‌ها)، می‌توانند پیوندهایی را در صفحات وب خود قرار دهند. گفتنی است که این استراتژی معمولاً به صورت ترکیبی با روش‌های محتوامحور ازقبیل هرز برچسب پیوند استفاده می‌شود.

۱-۳-۳- هرزنامه‌های مبتنی بر پنهان‌سازی صفحه

در این روش وب‌هرزها محتوای متفاوتی نسبت به آنچه کاربر می‌بیند در موتورهای جستجو نمایش می‌دهند. انواع روش‌های آن عبارتند از:

- **پنهان‌سازی**[1]

در این روش درخواست مشاهده وب‌سایت بررسی می‌شود. اگر از سوی خزشگرهای موتورهای جستجو بود، محتوای متفاوتی نسبت به کاربر عادی در آن نمایش داده می‌شود.

- **تغییر مسیر**[2]

صفحه اصلی وب‌سایت از روش‌های مختلف وب‌هرز برای بالا بردن رتبه خود در موتورهای جستجو استفاده می‌کند. وقتی کاربر از کلیک‌کردن بر روی پیوند نتایج موتور جستجو، درخواست مشاهده صفحه را می‌کند، به صفحه دیگری منتقل می‌شود.

[1]. Cloaking
[2]. Redirection

۱-۴- چالش‌های پیش رو

در مسئله مطرح‌شده با چالش‌هایی روبه‌رو هستیم که برطرف‌کردن هریک از آن‌ها می‌تواند گامی مؤثر در انجام یک تشخیص درست باشد که در ادامه به مهمترین آن‌ها اشاره خواهیم کرد.

۱-۴-۱- استخراج ویژگی

همان‌طور که در بخش مرور مقالات اشاره شد، تاکنون ویژگی‌های زیادی از صفحات وب استخراج شده‌است؛ برخی از این ویژگی‌ها مبتنی بر محتوا و برخی دیگر مبتنی بر پیوند هستند. این ویژگی‌ها در عین تفاوت، خاصیت مشترکی دارند که آماری بودن آنهاست. ویژگی‌هایی ازقبیل نسبت تعداد کلمات کلیدی به کل متن، تعداد کلمات سرآیند و تعداد پیوندهای موجود در متن، همگی خاصیتی عددی و شمارشی دارند. این امر باعث می‌شود هرزفرست‌ها به‌راحتی بتوانند نحوه فریب الگوریتم‌های تشخیص وب‌هرز موتورهای جستجو را کشف و با استفاده از آن رتبه سایت خود را افزایش دهند. به همین دلیل استخراج و معرفی ویژگی‌های جدیدی که نتوان به آسانی از آن‌ها سوءاستفاده کرد و کسب مقادیر با ارزش در آنها، مستلزم رعایت نکات فنی و کیفیت صفحات وب باشد، یکی از چالش‌های عمده این مسئله می‌باشد.

۱-۴-۲- انتخاب ویژگی

مساله انتخاب ویژگی، یکی از مسائلی است که در مبحث یادگیری ماشین و همچنین شناسائی آماری الگو مطرح است. این مساله در بسیاری از کاربردها (مانند طبقه بندی) اهمیت به سزائی دارد، زیرا در این کاربردها تعداد زیادی ویژگی وجود دارد، که بسیاری از آنها یا بلا استفاده هستند و یا اینکه بار اطلاعاتی چندانی ندارند. حذف نکردن این ویژگی‌ها علاوه بر بالا بردن بار محاسباتی می‌تواند منجر به کاهش نرخ تشخیص شود. برای مساله انتخاب ویژگی، راه حل‌ها و الگوریتم‌های فراوانی ارائه شده‌است که بعضا بار محاسباتی زیادی دارند. اگر چه امروزه با ظهور کامپیوترهای سریع و منابع

ذخیره سازی بزرگ این مشکل، به چشم نمی‌آید ولی از طرف دیگر، مجموعه داده‌های بسیار بزرگ و کاربردهای بلادرنگ باعث شده‌است که همچنان پیدا کردن یک الگوریتم سریع برای این کار مهم باشد.

۱-۴-۳- طبقه‌بندی کننده مناسب

همان‌طور که در [8] نشان داده شد، انتخاب یک الگوریتم طبقه‌بندی‌کننده خوب در افزایش دقت تشخیص وب‌هرزها بسیار مؤثر است. دلیل این امر آن است که وب‌هرزها نسبت به کل صفحات موجود در وب درصد کمی را تشکیل می‌دهند. به همین دلیل فضای داده‌های آموزشی نامتقارن است و این امکان وجود دارد که سیستم درست آموزش نبیند. به عنوان مثال در مجموعه داده استانداردی که برای این مساله وجود دارد، اگر سیستم پیشنهادی همه صفحات را غیرهرز در نظر بگیرد، دقت ۹۴٪ به دست می‌آید. به همین دلیل در این نوع خاص از داده، نرخ بازیابی و دقت همزمان باید بالا باشد؛ زیرا همان‌طور که گفته شد دقت بالا به‌تنهایی به نتیجه مطلوب منجر نمی‌شود. همچنین اگر تنها نرخ بازیابی بالا باشد، باز هم نتیجه قابل قبول نیست؛ زیرا ممکن است بسیاری از صفحات مرتبط در اثر نرخ تشخیص غلط بالای آن، هرز در نظر گرفته شده و از فهرست نهایی نتایج حذف شوند. به همین سبب است که استفاده از طبقه‌بندی‌کننده مناسب و توسعه صحیح آن و همچنین انتخاب معیاری که به خوبی بتواند کارایی الگوریتم را در مواجهه با داده‌های نامتوازن نشان دهد، به‌منظور کسب بهترین نتیجه الزامی است.

۱-۵- نوآوری‌ها و دستاوردهای رساله

در این رساله برآنیم تا مدلی را جهت تشخیص صفحات وب‌هرز از صفحات مفید ارائه نماییم. برای این کار می‌بایست ابتدا یک صفحه وب به یک بردار ویژگی تبدیل شود. به همین سبب نیازمند استخراج ویژگی هایی هستیم که به خوبی بتوانند این صفحات را نمایش دهند. همان طور که قبلا

اشاره شد، برای استخراج ویژگی از صفحات وب دو دیدگاه عمده وجود دارد. دسته‌ای از پژوهشگران بر ویژگی‌های مبتنی بر محتوا تاکید دارند و برخی دیگر بر ویژگی‌های مبتنی بر پیوند. در این رساله در کنار استفاده از این ویژگی ها، ویژگی‌های جدیدی مبتنی بر آدرس و کد HTML صفحه و همچنین انسجام متنی و معنایی محتوای صفحه معرفی شده‌است.

همچنین از آنجا که سرعت تشخیص صفحات هرز عامل مهمی در کارآمد بودن به شمار می‌رود، لذا تعداد ویژگی‌ها از اهمیت بالایی برخوردار است. به همین دلیل می‌بایست برای دستیابی به نتیجه مناسب این ویژگی‌ها را بررسی کرده، آن‌هایی که دارای اطلاعات کم و یا وابسته هستند را حذف و ویژگی‌هایی را انتخاب کنیم که نقش مؤثرتری در افزایش دقت دارند. روش‌های زیادی برای این کار وجود دارد که در این رساله روش جدیدی به نام Smart-BT معرفی شده‌است.

مرحله نهایی این کار نیز شامل انتخاب و طراحی یک الگوریتم مناسب جهت طبقه‌بندی داده‌ها است. همان طور که قبلاً شرح داده شد، ازآنجاکه نسبت صفحات وب‌هرز به کل صفحات وب بسیار اندک است، الگوریتم طبقه بندی که مورد استفاده قرار می‌گیرد بایستی قابلیت یادگیری داده‌های کم را داشته باشد. یکی از روش‌های موثر در این حوزه، سیستم ایمنی مصنوعی[1] است. پیشنهاد ما استفاده از الگوریتم انتخاب منفی و یابنده‌های شعاعی آن به عنوان طبقه‌بند است. زیرا این الگوریتم به دلیل ماهیتش بیشتر در حوزه تشخیص ناهنجاری استفاده شده و گزینه مناسبی به شمار می‌رود.

۱-۶- ساختار رساله

این رساله در ۵ بخش تنظیم شده‌است. بخش اول مقدمه‌ای بر وب‌هرز و انواع آن است. چالش‌های مساله و چکیده‌ای از راهکار پیشنهادی در این بخش ارائه شده‌است. فصل دوم به بررسی دقیقتر تحقیقاتی که تاکنون در این زمینه صورت گرفته است می‌پردازد. فصل سوم به تشریح راهکار پیشنهادی و اصول و الگوریتم‌های به کار رفته در آن می‌پردازد. فصل چهارم نتایج به دست آمده را

[1] Artificial Immune System (AIS)

تشریح می‌کند و در نهایت در فصل پنجم به جمع بندی و پیشنهادهایی برای کارهای آتی اختصاص یافته است.

۲

مرور کارهای گذشته

روش‌های محتوا محور

روش‌های پیوند محور

انواع ویژگی‌ها

۲-۱- مقدمه

همان‌طور که در فصل گذشته اشاره شده، وب‌هرزها صفحاتی هستند که با استفاده از تکنیک‌های فریب الگوریتم رتبه‌بندی موتورهای جستجو، باعث تغییر رتبه صفحات وب در نتایج جستجو می‌شوند. از آنجا که این صفحات باعث کاهش کیفیت نتایج جستجو و در نتیجه اتلاف وقت کاربر می‌شوند، تحقیقات زیادی در جهت ارائه روشی برای شناسایی آنها صورت گرفته است. دو رویکرد عمده در این زمینه وجود دارد. رویکرد اول مبتنی بر گراف وب است. با توجه به این گراف و تخمین میزان اعتماد به یک صفحه برحسب اعتبار صفحاتی که به آن اشاره می‌کنند می‌توان صفحات هرز را تشخیص داد. از الگوریتم‌های موفق در این زمینه می‌توان به HITS و PageRank اشاره کرد.

رویکرد دوم مبتنی بر محتوا و اجزای یک صفحه وب است و به‌نوعی مسئله طبقه‌بندی و یادگیری با ناظر محسوب می‌شود. عمده کارهای انجام‌شده در زمینه تشخیص وب‌هرز بر همین اساس است [18] [19]. این مقالات نیز برحسب دیدگاهی که دارند به دو دسته کلی تقسیم می‌شوند. برخی از آن‌ها از ویژگی‌های رایج استفاده کرده، به دنبال ارائه طبقه‌بند مناسبی هستند که عملکرد بهتری در فضای حالت خاص این مسئله داشته باشد. برخی دیگر نیز به دنبال استخراج ویژگی‌های جدید به منظور افزایش نرخ تشخیص این صفحات هستند که در ادامه به تشریح تحقیقات صورت گرفته در هر یک از آنها خواهیم پرداخت.

۲-۲- روش‌های مبتنی بر گراف وب

گراف وب گرافی است که نودهای آن را صفحات وب و یال‌های آن را پیوندهای بین این صفحات تشکیل می‌دهد. با توجه به این گراف و تخمین میزان اعتماد به یک صفحه برحسب اعتبار صفحاتی که به آن اشاره می‌کنند می‌توان صفحات هرز را تشخیص داد. در این روش‌ها پیوندها و ارتباطات بین صفحات

وب از محتوای آنها اهمیت بیشتری دارد. جستجوی موضوعی مستنتج از پیوند (HITS[1]) و TrustRank اصلی‌ترین الگوریتم‌های این دسته از روش‌ها هستند [20][21] [22] [23] .

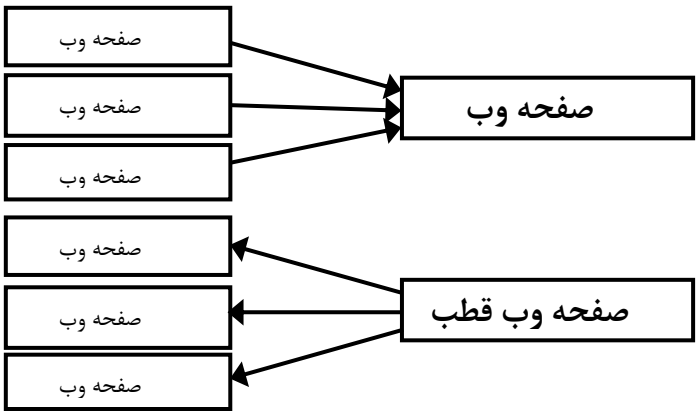
۲-۲-۱- الگوریتم PageRank

روش PageRank در سال ۱۹۹۸ از سوی پیج و همکاران معرفی شد [24]. این روش به عنوان یکی از بهترین روش‌های مقابله با وب‌هرز مورد توجه قرار گرفت. در این روش همه پیوندها وزن یکسانی در تعیین رتبه صفحه ندارند. پیوندهایی که مربوط به سایت‌هایی با رتبه بالاتر هستند، در مقایسه با پیوندهایی از سایت‌های با بازدیدکننده کمتر، ارزش بیشتری دارند. درنتیجه سایت‌هایی که به‌وسیله هرزفرست‌ها ایجاد می‌شوند نقش کمتری در تغییر رتبه وب‌سایت‌ها دارند. موتور جستجوی گوگل از این روش استفادمی‌کند.

۲-۲-۲- الگوریتم HITS

HITS روشی است که از سوی کلینبرگ ارائه شده‌است [25]. در این الگوریتم، سایت‌ها به دو گروه قطب‌ها[2] و قدرت‌ها[3] تقسیم می‌شوند. سایت‌های قطب شامل تعداد زیادی پیوندهای مربوط به سایت‌های قدرت هستند. شکل ۲-۱ این دو گروه را نشان می‌دهد. در این روش ابتدا تعدادی از صفحات که به نظر می‌رسد ارتباط بیشتری با پرس‌وجوی واردشده داشته باشند، استخراج می‌شوند (در مقاله اصلی این تعداد t = 200 است). برای غنی شدن این مجموعه، صفحات مورد نظر صفحه‌های یادشده نیز استخراج می‌شوند (حداکثر ۵۰ صفحه از هر صفحه طبق مقاله اصلی). سپس صفحات با توجه به پیوندهایی که با هم دارند به صورت یک گراف جهت‌دار در نظر گرفته می‌شوند.

[1] Hyperlink-Induced Topic Search
[2]. Hubs
[3]. Authorities



شکل ۲-۱ مفهوم قطب و قدرت [25]

با توجه به این گراف مشخص می‌شود هر صفحه به چه تعداد صفحاتی (درجه قطب) اشاره کرده و متقابلاً چه تعداد صفحاتی به آن اشاره می‌کنند (درجه قدرت). ترتیب نمایش نتایج با توجه به این مقادیر و بر اساس این اصول تعیین می‌شود:

- وب‌سایتی که به وب‌سایت‌های بیشتری اشاره می‌کند، قطب خوبی است.
- وب‌سایتی که وب‌سایت‌های بیشتری به آن اشاره می‌کند، قدرت خوبی است.
- وب‌سایتی که به قدرت‌های خوب بیشتری اشاره می‌کند، قطب بهتری است.
- وب‌سایتی که قطب‌های خوب بیشتری به آن اشاره می‌کنند، قدرت بهتری است.

این نوع الگوریتم‌ها حساس به عبارت جستجو شده توسط کاربر هستند و با یافتن این رشته در مجموعه‌ای از صفحات، نتایج مورد نظر را به کاربر نشان می‌دهد. نقطه ضعف اصلی این روش‌ها در این است که اگر یک صفحه جدید با اطلاعات کاملا مرتبط با جستجوی کاربر به صفحات وب اضافه شود ولی پیوندهای زیادی به سایتهای معتبر نداشته باشد، شانس زیادی برای دیده شدن در نتایج جستجو ندارد.

با ظهور این الگوریتم‌ها و موفقیت موتورهای جستجو در نمایش نتایج بهینه با استفاده از آن‌ها، هرزفرست‌ها تلاش کردند ساختار پیوند را به‌منظور افزایش رتبه خود تغییر دهند. به همین دلیل بسیاری

از الگوریتم‌های ارائه‌شده سعی در بررسی و تحلیل ساختار پیوند کرده‌اند. این الگوریتم‌ها را می‌توان به پنج گروه تقسیم کرد.

دسته نخست شکل رابطه صفحه را با صفحات دارای برچسب شناخته‌شده در نظر می‌گیرند. گروه دوم الگوریتم‌ها بر روی شناسایی گره‌ها و پیوندهای مشکوک تمرکز دارند. سومین گروه بر روی استخراج ویژگی‌های مبتنی بر پیوند برای هر گره و استفاده از روش‌های یادگیری ماشین به‌منظور شناسایی وب‌هرز متمرکز هستند. چهارمین گروه الگوریتم‌های پیوندمحور از ایده پالایش برچسب مبتنی بر توپولوژی گراف وب استفاده می‌کنند. آخرین گروه الگوریتم‌هایی هستند که از تکنیک‌های تنظیم گراف برای تشخیص وب‌هرز استفاده می‌کنند.

۲-۲-۳- الگوریتم‌های مبتنی بر انتشار برچسب

ایده اصلی پشت الگوریتم‌های این گروه به این صورت است که به برخی صفحات وب، برچسب «شناخته‌شده» تعلق می‌گیرد. آنگاه با توجه به قوانین انتشار، برچسب سایر صفحات نیز مشخص می‌شود. نخستین الگوریتمی که در این رده معرفی شد TrustRank بود [26]. این الگوریتم که مبتنی بر مفهوم اعتماد در شبکه‌های اجتماعی است، از محبوب‌ترین و موفق‌ترین روش‌های مقابله با وب‌هرز به شمار می‌رود. در این روش صفحات معمولاً به صفحات خوب و به‌ندرت به صفحات بد اشاره می‌کنند؛ بنابراین در ابتدا گروهی از صفحات معتبر انتخاب شده و به آن‌ها نمره اعتماد تعلق می‌گیرد. سپس این صفحات همانند الگوریتم PageRank دنبال می‌شوند.

الگوریتم TrustRank تفاوت چندانی با PageRank ندارد. در این الگوریتم انتخاب سایت‌های اولیه بسیار مهم است و معمولاً برای شروع از صفحاتی استفاده می‌شود که رتبه صفحه بالاتری دارند.

نویسندگان برای انتخاب یک مجموعه اولیه قابل اطمینان، پیشنهاد کردند از رتبه صفحه معکوس[1] بر گرافی که یال‌های آن معکوس شده‌اند، استفاده شود. سپس آن‌ها رتبه معکوس را برای همه صفحات وب محاسبه کرده و از آن میان k صفحه‌ای که بیشترین رتبه را داشتند، انتخاب کردند و از کاربر انسانی خواستند بر اساس میزان اطمینانی که به این صفحات دارد به آن‌ها نمره دهد. پس از آن بر اساس نمرات کسب‌شده، الگوریتم PageRank و TrustRank بر روی صفحات اجرا شد که الگوریتم TrustRank نتایج بهتری را به دست آورد. همچنین گینگی و همکاران برای افزایش دقت شناسایی صفحات وب‌هرز از ترکیب PageRank و TrustRank استفاده کردند که در این روش صفحاتی که PageRank بالا و TrustRank پایینی دارند به عنوان صفحات هرز پیوندمحور در نظر گرفته می‌شود [27].

در مرجع [12] الگوریتمی به نام PageRank کوتاه‌شده معرفی شد. نویسندگان فرض کردند صفحات هرزی که مزرعه پیوند هستند، ممکن است در بازه زمانی کوتاهی حامیان زیادی داشته باشند، اما در درازمدت حمایت کننده‌ای ندارند یا تعداد آن‌ها بسیار کم است. بر مبنای این فرض آن‌ها روش PageRank کوتاه‌شده را ارائه کردند. در این روش نخستین سطح پیوندها نادیده گرفته شده و گره[2]های مراحل بعد محاسبه می‌شوند.

کار بعدی که در زمینه انتشار اعتماد انجام شد، Anti-Trust Rank نام دارد [23] و بر پایه این فرض بنا شده که اگر صفحه‌ای به صفحه بدی اشاره کند، آن صفحه نیز به احتمال زیاد صفحه خوبی نیست. این الگوریتم در واقع معکوس الگوریتم TrustRank است که نمرات بد را توزیع و صفحات بد را به جای صفحات خوب انتخاب می‌کند. در قیاس با TrustRank، این الگوریتم از دقت بالاتری برخوردار است.

1. Inverted PageRank
2. Node

یکی دیگر از الگوریتم‌های ضد وب‌هرز BadRank است. در این الگوریتم، برای شروع یک مجموعه از صفحات بد در نظر گرفته می‌شود و به آن‌ها یک نمره بد تعلق می‌گیرد. سپس این صفحات بد دنبال شده و به صفحاتی که از این طریق دیده می‌شوند، نمرات بد تعلق می‌گیرد. این روند مرتب تکرار می‌شود. درنهایت صفحات هرز آن‌هایی هستند که نمرات بالایی دارند [28].

در [7] پژوهشگران تصمیم گرفتند از هر دو روش انتشار اعتماد و بی‌اعتمادی استفاده کنند. بدین-منظور ابتدا به بررسی نحوه انتشار درستی در الگوریتم TrustRank پرداختند. در این روش هریک از فرزندان رتبه‌ای معادل رتبه اعتماد والدین خود دریافت می‌کند. آن‌ها برای رفع این مشکل دو استراتژی ارائه دادند:

۱. تقسیم‌بندی ثابت: در این روش هر فرزند اعتماد کاهش‌یافته یکسانی نسبت به والد خود دریافت می‌کند.

$$TR(Child) = C * TR(Parent) \quad (۱-۲)$$

۲. تقسیم‌بندی لگاریتمی: در این روش هر فرزند اعتماد برابری نسبت به والدین خود دریافت می‌کند که به‌وسیله لگاریتم تعداد فرزندان نرمال‌سازی شده‌است.

$$TR(Ch) = c.\frac{TR(p)}{\log(1 + |Out(p)|)} \quad (۲-۲)$$

در این فرمول TR(p) میزان اعتماد به صفحه p، Out(p) تعداد لینک‌های خروجی p و c یک عدد ثابت است. همچنین برای محاسبه میزان نهایی اعتماد به یک صفحه نیز فرمول ۳-۲ ارائه شد:

$$Total\ Score(p) = \eta.TR(p) - \beta.AntiTR(p) \quad (۳-۲)$$

که در آن $\eta, \beta \epsilon (0,1)$ ضریب تأثیر و AntiTR(p) میزان بی‌اعتمادی نسبت به p است. آزمایش‌ها نشان داد این روش بسیار بهتر از TrustRank عمل می‌کند.

الگوریتم استفاده‌شده در [11] از ویژگی‌های تجزیه رتبه صفحه برای تخمین مقدار رتبه صفحه نادرستی استفاده می‌کنند که از گره‌های مشکوک به دست می‌آید. بدین منظور الگوریتم SpamRank پیشنهاد شده‌است از شبیه‌سازهای مونت کارلو برای یافتن پشتیبان‌های[1] یک صفحه استفاده کنند. ایده آن‌ها این بود که مقادیر رتبه صفحه پیوندهای ورودی در صفحات معمولی باید توزیع قانون قدرت[2] را دنبال کند. آن‌ها توزیع رتبه صفحه همه پیوندهای ورودی را بررسی کردند. اگر یک الگوی طبیعی به‌وسیله توزیع دنبال نشود، آنگاه برای این صفحه جریمه در نظر گرفته می‌شود.

در [29] مفهوم "هرز دسته‌ای"[3] آمده است. هرز دسته‌ای مقدار رتبه صفحه‌ای را که از صفحات هرز می‌آید، اندازه می‌گیرد. این هرز مشابه الگوریتم TrustRank برای تخمین مقدار رتبه صفحه‌ای که از صفحات خوب می‌آید، به هسته‌شناسی صفحات خوب نیاز دارد. الگوریتم از دو مرحله تشکیل شده‌است. ابتدا بردار PageRank $\vec{\pi}$ وTrustRank $\overrightarrow{\pi`}$ را محاسبه و با استفاده از فرمول (۴-۲) مقدار هرز دسته‌ای را برای هر صفحه تخمین می‌زند. در مرحله بعد، مقدار آستانه‌ای که وابسته به هرز دسته‌ای است، ایجاد می‌شود. نکته مهم این است که این الگوریتم می‌تواند شناخت مؤثری راجع به صفحات مخرب فراهم کند.

$$\overrightarrow{m} = \frac{\vec{\pi} - \vec{\pi`}}{\vec{\pi}} \qquad (۴-۲)$$

در ادامه گوپتا و همکاران در کنار استفاده از روش شناسایی هرز دسته‌ای، از تحلیل محتوا نیز استفاده کردند که منجر به کاهش ۶۷-۴۰ درصدی نرخ false positive شد [30].

1. Support
2. Power Rule Distribution
3. Spam Mass

از دیگر کارهای انجام‌شده در این زمینه، تحلیل پیوند بر اساس اعتبار است که در [31] شرح داده شده‌است. برای هر صفحه چندین حوزه[1] اعتبارسنجی تعریف و چندین روش تخمین پیشنهاد شده‌است و نشان می‌دهد چطور از هریک برای تشخیص وب‌هرز استفاده می‌شود. برای این کار ابتدا مفهوم مسیر بد[2] تعریف می‌شود که به معنای تعداد پرش‌های تصادفی است که از صفحه جاری شروع شده و به صفحه هرز خاتمه می‌یابد. بدین‌ترتیب مقدار هر حوزه به دست می‌آید.

۲-۲-۴- الگوریتم‌های هرز پیوند و وزن‌دهی مجدد[3]

الگوریتم‌های معرفی‌شده در این بخش، تمایل به پیداکردن پیوندهای نامطمئن و پایین آوردن رتبه آن‌ها دارند. مقاله اصلی که در [32] ارائه شد، مشکلات الگوریتم HITS [25] ازقبیل سلطه روابط تقویت متقابل[4] و موضوع رانش گراف همسایه[5] را برطرف می‌کند. برای این کار یک تحلیلگر محتوا به تحلیلگر پیوند اضافه شده‌است. در ادامه به هر یال یک وزن معتبر $\frac{1}{k}$ تعلق می‌گیرد که k نشان‌دهنده تعداد صفحاتی است که از یک سایت به یک صفحه از سایت دیگر پیوند می‌خورد. همچنین اگر یک صفحه از سایت نخست به L صفحه از سایت دیگر اشاره کند، یک وزن قطب $\frac{1}{l}$ به لبه تعلق می‌گیرد. برای رفع مشکل رانش موضوع نیز از توسعه پرس‌وجو با استفاده از K پرتکرارترین کلمات هر صفحه استفاده شد. همچنین برای ایجاد مجموعه صفحات کاندید هرز شدن، میزان مرتبط بودن صفحات به عنوان یک فاکتور در محاسبات HITS مورد استفاده قرار گرفت.

1. Scope
2. BadPath
3. Reweighting
4. Domination of mutually reinforcing relationships
5. Topic drift

ایرون و همکاران در [8] الگوریتم HostRank را معرفی کردند که در مقایسه با الگوریتم PageRank از مقاومت بالاتری در خصوص هرزپیوند برخوردار است. به‌علاوه، لمپل و همکاران [33] مفهوم «تأثیر TKC[1]» را برای نخستین بار منتشر کردند. در این روش صفحاتی که با یکدیگر در ارتباط هستند در رتبه بالاتری قرار می‌گیرند. برای فریب‌دادن این الگوریتم، هرزفرست‌ها از مزارع پیوند[2] استفاده می‌کنند. به همین دلیل آن‌ها الگوریتم SALSA[3] را ارائه کردند. در این الگوریتم دو پیمایش به صورت تصادفی برای تخمین امتیاز قدرت و قطب صفحات انجام می‌شود که در مقایسه با الگوریتم HITS نسبت به تأثیر TKC پایدارتر است. این روش در [34] توسعه داده شد و در آن از ساختار خوشه‌بندی بر روی صفحات و الگوی ارتباطی برای ازبین‌بردن پیوندهای مخرب استفاده شد. ایده اصلی این است که به جای محاسبه تعداد گره‌های جدا از هم، تعداد خوشه‌هایی که به یک صفحه اشاره دارند، شمارش شوند.

در [35] برای حل مشکل داده‌های کم[4] در الگوریتم HITS از وزن‌دهی مجدد به پیوندهای ورودی و خروجی صفحات در مجموعه اولیه استفاده شده‌است. به این صورت که اگر درجه ورودی یک صفحه جزء ۳ درجه کوچک و درجه خروجی جزء ۳ درجه بزرگ باشد، وزن ۴ به تمام صفحات پیوند داده‌شده در صفحات ریشه تعلق داده شده و دوباره الگوریتم HITS اجرا می‌شود.

در [7] مفهوم پیوند کامل[5] عنوان شده که به معنای الحاق برچسب پیوند مرتبط با پیوند مربوطه است که برای شناسایی صفحات با الگوی مشکوک استفاده می‌شود. در این روش فرض شده‌است صفحاتی که بالاترین هم‌پوشانی[6] را در پیوندهای خود دارند، با احتمال زیادی به‌وسیلهٔ ماشین ایجاد شده‌اند. در این الگوریتم از ماتریس A که شامل صفحات و پیوندهای آن‌ها است، استفاده شده‌است. $A_{ij}=1$ نشان‌دهنده

1. Tightly Knit Community
2. Link Farms
3. Stochastic Approach for Link-Structure Analysis
4. Small-in-large-out
5. Complete Hyperlink
6. Overlap

این است که صفحه P_i شامل j پیوند کامل است. درنهایت از الگوریتم مشابه HITS بر روی ماتریس استفاده شده‌است.

۲-۲-۵- الگوریتم‌های مبتنی بر پالایش برچسب

ایده اصلی پالایش برچسب مربوط به داده‌کاوی است و از آن برای رفع مشکلات طبقه‌بندی استفاده شده‌است. در این بخش از این ایده برای تشخیص وب‌هرز استفاده شده‌است. تعدادی از استراتژی‌های پالایش گراف صفحات وب در [12] ارائه شده‌است. الگوریتم این مقاله در دو مرحله عمل می‌کند؛ ابتدا برچسب‌ها با استفاده از الگوریتم تشخیص وب‌هرز که در [17] آمده اختصاص داده شده و سپس برچسب-ها به سه طریق تصحیح می‌شوند.

استراتژی نخست، استراتژی خوشه‌بندی است. به این صورت که اگر از قبل پیش‌بینی شده باشد که بیشتر صفحات یک خوشه وب‌هرز باشند، تمام صفحات آن خوشه به عنوان وب‌هرز در نظر گرفته می‌شود. استراتژی دوم برای تصحیح برچسب‌ها، مبتنی بر گسترش با گام‌های تصادفی است که در [36] توضیح داده شده‌است. در این استراتژی نرمال‌سازی الگوریتم اصلی برای پیش‌بینی خوشه‌ها انجام می‌شود. استراتژی سوم از یادگیری گرافیکی پشته‌ای استفاده می‌کند [23]. ایده اصلی در این استراتژی، گسترش ویژگی‌های اصلی یک شیء و اضافه کردن ویژگی‌های جدید به آن بر اساس متوسط پیش‌بینی‌های انجام-شده برای صفحات مرتبط است. درنهایت با اضافه شدن ویژگی‌های جدید، الگوریتم داده‌کاوی دوباره اجرا می‌شود.

در [37] پیشنهاد شده‌است یک فضای ویژگی جدید و مستقل بر اساس پیش‌بینی‌های مرحله نخست شامل برچسبی از طبقه‌بندی اولیه، درصد پیوندهای ورودی وب‌هرز، درصد پیوندهای خروجی که به وب-

هرز اشاره دارد و غیره ایجاد شود. درنهایت هفت ویژگی جدید در نظر گرفته و گزارش دادند که باعث افزایش کارایی نسبت به طبقه‌بندی اولیه شده‌است.

۲-۳- روش‌های متمرکز بر طبقه‌بندی کننده

در این دسته از روش‌ها که مبتنی بر محتوا و اجزای یک صفحه وب هستند، به جای تمرکز بر معرفی ویژگی‌های جدید، تلاش می‌شود با استفاده از ویژگی‌های موجود و ارائه الگوریتم‌های طبقه‌بندی جدید، باعث بهبود عملکرد سیستم‌های تشخیص صفحات وب‌هرز شوند. در این تحقیقات معمولا از الگوریتم‌های طبقه‌بندی از قبیل درخت تصمیم، ماشین بردار پشتیبان (SVM) و غیره استفاده می‌شود [38][39] [40]. این الگوریتم‌ها در برخی موارد به منظور افزایش کارایی با استفاده از روش‌های تجمعی[1]، تقویت می‌شوند [41][42].

به تازگی در [43] یک چارچوب سیستماتیک مبتنی بر الگوریتم AIDVCH و در [44] یک چارچوب شناسایی صفحات وب‌هرز با استفاده از منطق فازی ارائه شده‌است. شبکه عصبی مصنوعی، شبکه باور عمیق[2] و SVM با حاشیه دوگانه[3] نیز از دیگر الگوریتم‌هایی هستند که به منظور تشخیص صفحات وب‌هرز مورد استفاده قرار گرفته‌اند [45] [46, 47]. همچنین در [48] و [49] یک چارچوب به‌منظور یادگیری افزایشی برای تشخیص وب‌هرز ارائه شده‌است. در[45] نیز یک مدل طبقه‌بند SVM با حاشیه دوگانه و در [50] طبقه‌بندی مبتنی بر اصل حداقل طول توصیف[4] به‌منظور وب‌هرز معرفی کرده‌اند.

در این دسته از روش‌ها صفحات وب با استفاده از ویژگی‌هایی که از محتوای آن‌ها استخراج شده‌است، تبدیل به بردار ویژگی می‌شوند. سپس با استفاده از الگوریتم‌های یادگیری ماشینی سعی در آموزش یک

[1] Ensamble
[2] Deep Belief Network
[3] Dual-Margin
[4] Minimum Description Length Principle

سیستم به منظور تشخیص صفحات هرز و غیرهرز شده‌است. این روش‌ها معمولا از ویژگی‌هایی از قبیل تعداد کلمات در صفحه، تعداد کلمات در عنوان، میانگین طول کلمه، نرخ فشرده سازی، انتروپی صفحه، تعداد کلمات در صفحه اصلی، تعداد کلمات در عنوان صفحه خانه[1]، میانگین طول کلمه در صفحه خانه، نرخ فشرده سازی در صفحه خانه، انتروپی صفحه خانه، میانگین و انحراف معیار تعداد کلمات در کل صفحات وب‌سایت، میانگین و انحراف معیار طول کلمات در کل وب‌سایت، میانگین و انحراف معیار نرخ فشرده سازی صفحات وب‌سایت و ویژگی هایی از این دست جهت تبدیل صفحات وب به بردار ویژگی استفاده می‌کنند.

همان‌طور که اشاره شد مقالاتی که در این بخش بررسی می‌شوند تمرکز عمده شان بر روی روش طبقه‌بندی و نه ویژگی‌های مورد استفاده است. به عنوان مثال کریم‌پور و دیگر نویسندگان ابتدا با استفاده از روش کاهش ویژگی PCA[2] ویژگی‌ها را کاهش داده، سپس با استفاده از یک روش طبقه‌بندی نیمه-نظارتی به نام EM-Naïve Bayesian اقدام به شناسایی وب‌هرز کردند [51]. رانگ ساوانگ و همکاران [52] نیز از الگوریتم کلونی مورچه‌ها برای طبقه‌بندی وب‌هرز استفاده کردند. نتایج نشان داد این روش در مقایسه با SVM و درخت تصمیم دقت بیشتر و خطای کمتری دارد. همچنین سیلوا و دیگر نویسندگان [53] در مقاله خود تأثیر روش‌های مختلف طبقه‌بندی ازقبیل درخت تصمیم، SVM، KNN، LogitBoost، Bagging و AdaBoost را بر میزان دقت سیستم تحلیل و بررسی کردند.

ازآنجاکه مجموعه داده صفحات هرز و غیرهرز یک مجموعه داده نامتوازن است و تعداد صفحات هرز کمتر از غیرهرز است در [54] تاک و دیگران، به‌منظور غلبه بر این مشکل از روش گراف شبکه‌های عصبی

[1] Home Page
2. Principal Component Analysis

(GNN[1]) حساس به هزینه استفاده کرده‌اند. این روش با روش‌های GNN لایه‌لایه[2]، GNN و MLP[3] مقایسه شده و نتایج بهتری را به بار آورده است.

با توجه به اینکه وب‌هرزها از تکنیک‌های مختلفی استفاده می‌کنند، در مرجع [55] داده‌های آموزشی را با استفاده از منطق فازی طبقه‌بندی کرده و اقدام به تولید خوشه می‌کنند. سپس بر این اساس داده‌های آزمون را طبقه‌بندی می‌کنند. سانی و ماتور [56] ابتدا ویژگی‌های استخراج‌شده برای صفحات هرز را به‌وسیلهٔ روش SVD[4] کاهش داده، سپس با استفاده از Naïve Bayes عمل طبقه‌بندی را انجام داده‌اند. کیهان‌پور و مشیری [57] نیز ابتدا با استفاده از تحلیل ضریب همبستگی[5] ویژگی‌های صفحات وب‌هرز را کاهش داده، سپس با استفاده از روش برنامه‌نویسی ژنتیک سطح‌بندی شده چندجمعیتی[6]، عملیات طبقه‌بندی صورت پذیرفته است.

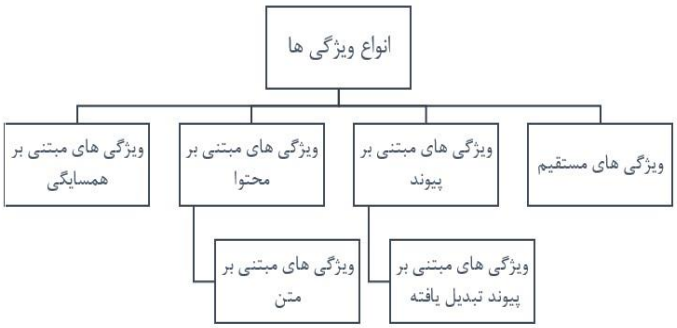
در کار جدیدی که در [8] ارائه شد نیز تأثیر ویژگی‌ها و مدل‌های مختلف یادگیری ماشین بر کیفیت شناسایی وب‌هرز بررسی شد. نویسندگان از جدیدترین روش‌های طبقه‌بندی ازقبیل LogitBoost و RandomForest و ویژگی‌های متن‌محوری استفاده کردند که نحوهٔ محاسبه آسانی داشتند. آن‌ها نشان دادند ویژگی‌های عمومی و محاسباتی سخت ازقبیل رتبه صفحه، تنها بهبود اندکی در افزایش کیفیت طبقه‌بندی دارند؛ بنابراین نویسندگان ادعا کردند انتخاب مدل طبقه‌بندی بسیار مهم‌تر از ویژگی‌های به‌دست‌آمده است.

1. Graph Neural Networks
2. Layered
3. Multi-Layer Perceptron
4. Singular Value Decomposition
5. Correlation Coefficient
6. Layered Multi-Population Genetic Programming

تحقیقات نشان می‌دهد که رویکردهای یادگیری ماشین نسبت به روش‌های رتبه بندی به نتایج صحیح تری منجر می‌شود. اما چالشی که در این زمینه وجود دارد این است که طبقه بندها نیاز دارند تا با استفاده از داده‌های جدید به روز شوند. همچنین بعضی از روش‌ها مثل شبکه عصبی مصنوعی (ANN) به زمان محاسباتی قابل توجهی برای شناسایی وب‌هرزها نیاز دارند که این امر برای موتورهای جستجو که سرعت فاکتور بسیار مهمی برای آنها است قابل اغماض نیست.

۲-۴- روش‌های متمرکز بر ویژگی‌ها

پیش از اینکه به بررسی تحقیقات صورت گرفته در این زمینه بپردازیم ابتدا لازم است با انواع ویژگی‌های مستخرج از صفحات وب آشنا شویم (شکل ۲-۲).



شکل ۲-۲ انواع ویژگی‌ها [58]

ویژگی‌های منبعث از صفحات وب بر حسب منبعی که برای استخراجشان استفاده شده به چهار دسته اصلی مستقیم، مبتنی بر پیوند، مبتنی بر محتوا و مبتنی بر همسایگی تقسیم می‌شوند. ویژگی‌های مستقیم ویژگی‌هایی هستند که بدون باز کردن صفحه، قابل رویت هستند، از قبیل طول آدرس اینترنتی، دامنه مورد استفاده و مواردی از این دست. تعداد این ویژگی‌ها اندک است و معمولا در زمره ویژگی‌های

مبتنی بر پیوند قرار می‌گیرند. ویژگی‌های مبتنی بر همسایگی نیز ویژگی‌هایی است که از گراف وب صفحه استخراج می‌شود و در قسمت قبل توضیح داده شد.

ویژگی‌های مبتنی بر محتوا از اطلاعات فایل متنی صفحات استفاده می‌کنند که قسمت عمده این فایل را محتوای صفحه تشکیل می‌دهد. این ویژگی‌ها شامل مقادیری از قبیل تعداد کلمات در صفحه، میانگین تعداد کلمات، تعداد کلمات استفاده شده در عنوان و غیره می‌باشند. ویژگی‌های مبتنی بر پیوند بر تعداد و نحوه چینش پیوندهای موجود در صفحات وب تمرکز دارند [۴]. ازجمله این ویژگی‌ها می‌توان تعداد پیوندهای خروجی صفحه اصلی، تعداد پیوندهای ورودی صفحه اصلی و نسبت تعداد پیوندهای خروجی داخلی به‌کل پیوندهای خروجی را نام برد [۵]. این ویژگی‌ها هم در صفحه اصلی میزبان[1] و هم در صفحه‌ای از میزبان که بالاترین رتبه[2] را دارد محاسبه می‌شوند.

ویژگی‌های مبتنی بر پیوند تبدیل یافته زیرمجموعه‌ای از ویژگی‌های پیوند محور است که شامل تبدیلات عددی ساده و ترکیب ویژگی‌های مبتنی بر پیوند می‌شود [۴]. ازجمله این ویژگی‌ها می‌توان لگاریتم پیوندهای ورودی به صفحه اصلی، لگاریتم پیوندهای خروجی از صفحه اصلی و لگاریتم میانگین پیوندهای ورودی از صفحات خروجی به صفحه اصلی را نام برد. تبدیلات بیشتر شامل نسبت بین Indegree/PageRank یا TrustRank/PageRank و لگاریتم ویژگی‌های مختلف است.

علاوه بر این ویژگی‌ها که مجموعه کامل آنها در مجموعه داده WEBSPAM-UK2007 جمع آوری و توضیح داده شده‌است، محققان سعی در بررسی دقیق رفتار وب‌هرزها و معرفی ویژگی‌های جدیدی که باعث افزایش نرخ تشخیص آنها شود داشته اند. به‌عنوان مثال در [59] تعدادی ویژگی با توجه به رفتار کاربران معرفی شده‌است. همچنین در [60] از مدل‌سازی موضوعی و در [61] از بخش‌های زبانی به‌منظور

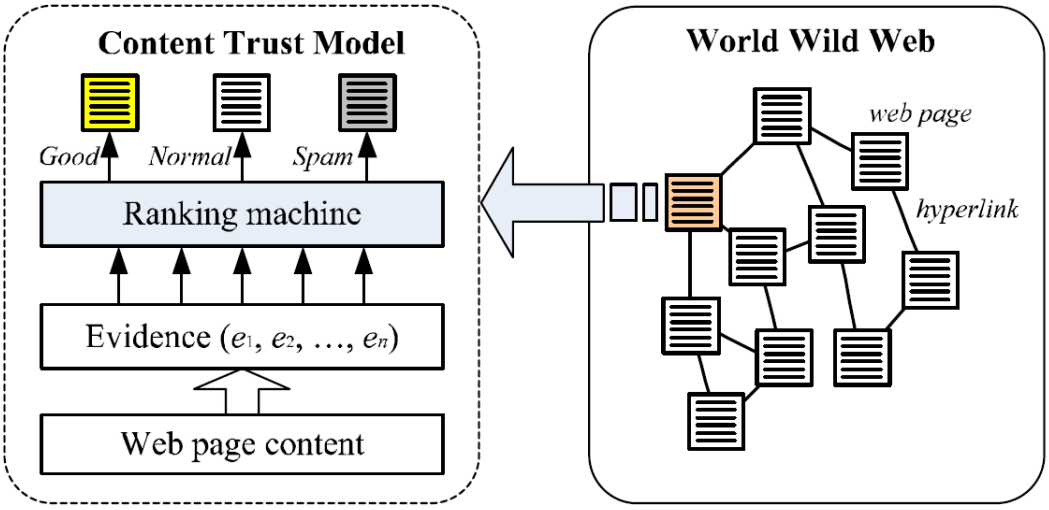
[1] Home Page
[2] PageRank

استخراج ویژگی استفاده شده‌است. برخی نیز از ویژگی‌های معرفی‌شده توسط دیگر منابع استفاده کرده، عمل انتخاب [62] یا ترکیب ویژگی [57] را به‌منظور کاهش ابعاد فضای حالت و بهبود نرخ تشخیص انجام داده‌اند. در [63] تعدادی ویژگی با توجه به میزان انتروپی داده‌های پرت[1] معرفی شده‌است. در [64, 41, 65] ویژگی‌هایی برمبنای مدل‌زبانی[2] و در [66, 67] نیز ویژگی‌هایی مبتنی بر کیفیت پیوندهای[3] صفحه ارائه شده‌است. همچنین در [68, 69] از مدل‌سازی موضوعی[4] و در [61] [70] از بخش‌های زبانی[5] به‌منظور استخراج ویژگی استفاده شده‌است.

صالح و همکاران [71] یک مجموعه ویژگی جدیدی ازقبیل کلمات کلیدی مورد پسند عموم، کاراکترهای گراف N-Gram و سطح جمله کلمات پرتکرار را برای تشخیص وب‌هرزهای عربی معرفی کردند. سپس از ترکیب این ویژگی‌ها و ویژگی‌های عمومی که معمولاً برای شناسایی وب‌هرز استخراج می‌شود، برای طبقه‌بندی صفحات استفاده کردند. بهترین F-measure به‌دست‌آمده به‌واسطهٔ رده‌بند جنگل تصادفی[6] ۵۴/۹۹ بوده است.

در [72] یک مدل کلی از میزان اعتماد به محتوا به‌منظور شناسایی وب‌هرز ارائه شده‌است که شمایی از آن را در شکل ۲-۳ مشاهده می‌کنید.

[1] Outlier
[2] Language Model
[3] Qualified Link
[4] Topic Modelling
[5] Lexical Items
[6] Random Forrest



شکل ۲-۳شمای کلی مدل اعتماد به محتوا [72]

این کار با استفاده از الگوریتم‌های رتبه‌دهی موجود در حوزه یادگیری ماشین صورت پذیرفته است. به این صورت که با درنظرگرفتن ویژگی‌های کیفی افزون بر ویژگی‌های مبتنی بر متن رایج، ۱۵ ویژگی متنی و ۶ ویژگی کیفی معرفی و مورد بررسی قرار گرفته‌اند. سپس یک سیستم واقعی تشخیص وب‌هرز توسعه داده شده و با داده‌های واقعی تست شده‌است. در این مقاله دو دسته ویژگی معرفی شده‌است. ویژگی‌های مبتنی بر متن و ویژگی‌های مبتنی بر کیفیت اطلاعات که در مدل ارائه شده و ترکیب آن‌ها مورد استفاده قرار گرفته است. ویژگی‌های مبتنی بر متن عبارتند از:

- تعداد کلمات موجود در صفحه
- تعداد کلمات موجود در عنوان صفحه
- متوسط طول کلمات
- نسبت تعداد کلمات موجود در برچسب پیوند به کل کلمات صفحه
- نسبت محتوای دیداری: نسبت مجموع طول همه کلمات صفحه به سایز کل صفحه
- نرخ فشرده‌سازی: نسبت اندازه صفحه فشرده‌شده به حالت عادی صفحه
- نسبت مجموع کلمات کلیدی استفاده‌شده در متن به کل کلمات موجود در یک صفحه

- نسبت تعداد کلمات کلیدی استفاده‌شده در متن به تعداد کلمات کلیدی در نظر گرفته‌شده
- احتمال n-gram مستقل
- احتمال n-gram شرطی
- ویژگی‌های استخراج‌شده‌از جزء میزبان یک URL
- شناسایی آدرس‌های IP ارجاع‌شده به‌وسیلهٔ هاست‌های معتبر
- داده پرت موجود در توزیع گراف صفحه
- نرخ به‌روزرسانی صفحات در یک سایت
- تکرار بیش‌ازحد محتوا

دسته دوم ویژگی‌های مبتنی بر کیفیت اطلاعات هستند. مسئله مهم در ارزیابی کیفیت صفحات وب، انتخاب معیارهای کیفی مناسب است. بدین‌منظور شش فاکتور معرفی شد:

- به‌روزرسانی: صفحه چه مدت یک بار به‌روزرسانی می‌شود. بدین‌منظور می‌توان زمان آخرین اعمال تغییرات در صفحه را مورد بررسی قرار داد.
- قابلیت استفاده: چه تعداد پیوند خراب در صفحه وجود دارد. برای این کار نسبت تعداد پیوندهای خراب صفحه را به تعداد کل پیوندها در نظر می‌گیریم.
- نرخ اطلاعات مفید: عبارت است از نسبت اطلاعات مفید به سایز کل صفحه. برای محاسبه آن مجموع طول نمونه‌ها[1] بعد از پیش‌پردازش، تقسیم بر سایز صفحه می‌شود.
- مالکیت: مالک صفحه چه میزان معتبر است. این کار بر اساس امتیازدهی مجله Yahoo! Internet Life صورت گرفته است.

1. Token

- شهرت: این صفحه به چه میزان از صفحات دیگر ارجاعات دریافت کرده‌است که با اندازه‌گیری تعداد پیوندهای ارجاع‌شده به صفحه به‌وسیلهٔ سایت Altavista[1] صورت می‌گیرد.
- انسجام: میزان تمرکز صفحه بر روی یک موضوع مشخص.

در کنار ویژگی‌های توضیح داده‌شده، از برخی ویژگی‌های متنی که معمولاً در این نوع کارها به کار برده می‌شود نیز استفاده و درنهایت ۳۸ ویژگی پیاده‌سازی شده‌است. برای آموزش مدل نیز از روش‌های SVM، Naïve Bayes و درخت تصمیم استفاده شد که SVM بهترین نتیجه را در پی داشته است. عملیات رتبه‌بندی به‌وسیلهٔ نرم‌افزار weka صورت گرفته است که درنهایت دقت ۹۰/۱۳٪ به دست آمده است.

در [41] یک سیستم شناسایی وب‌هرز مبتنی بر طبقه‌بندی ارائه می‌شود که از دو گروه ویژگی استفاده می‌کند. گروه نخست ویژگی‌های مبتنی بر پیوند هستند که اعتبار پیوندها را چک می‌کنند و گروه دوم مبتنی بر محتوا هستند که به کمک یک مدل زبانی استخراج شده‌اند. درنهایت با استفاده از یک طبقه‌بند خودکار هر دو نوع ویژگی در جهت رسیدن به دقت بالاتر ترکیب می‌شوند. سیستم ارائه‌شده، در دو مرحله اطلاعات پیوندهای موجود در یک صفحه را بازیابی و تحلیل کرده، چندین ویژگی که پیچیدگی محاسباتی کمتر و اطلاعات بیشتری داشته‌اند از آن استخراج می‌کند. برای بررسی تأثیر این ویژگی‌ها از پیاده‌سازی الگوریتم‌های طبقه‌بندی Metacost، Naïve Bayes و SVM به‌وسیلهٔ نرم‌افزار weka استفاده شده که بالاترین دقت به‌وسیلهٔ SVM و به میزان ۹۲٪ به دست آمده است.

از دیگر کارهایی که در این زمینه صورت گرفته است می‌توان به مرجع [73] و [74] اشاره کرد. در این مقالات نویسندگان به ویژگی‌هایی ازقبیل اعتبار واژگان، تنوع کلمات و محتوا، تنوع نحوی و آنتروپی، احساسی بودن، استفاده از آواهای فعال و غیرفعال و دیگر ویژگی‌های NLP توجه کرده‌اند. سرانجام در

[1] www.altavista.com

[75] نویسندگان تعدادی ویژگی مبتنی بر وقوع کلمات کلیدی صفحه را که ارزش تبلیغاتی یا هرز بودن بالایی دارند، ارائه کردند. آن‌ها نشان دادند ویژگی‌های زیر قدرت جداکنندگی بالایی دارند:

- قصد تبلیغ برخط (OCI) که مقداری است که به‌واسطهٔ Microsoft Ad_Center به URLها اختصاص می‌یابد.
- نتیجه طبقه‌بندی Yahoo! Mindset در مورد تجاری یا غیرتجاری بودن.
- کلمات کلیدی رایج در Google Adwords.
- تعداد تبلیغات Google Adsense موجود در صفحه.

بر اساس گزارش‌ها افزودن این ویژگی‌ها دقت کار را بیش از ۴٪ بالا برده و از ۶۹٪ به ۷۳٪ رسانده است. نتایج به‌دست‌آمده نشان داده این روش میزان بازیابی بهتری دارد، در حالی که الگوریتم‌های TrustRank [23] و Trust-Distrust [76] دقت بیشتر و میزان یادآوری کمتری دارند. جدول ۱-۲ لیست ویژگی‌های ارائه شده توسط محققان را که در این بخش بررسی شد را نشان می‌دهد.

جدول ۲-۱ لیست ویژگی‌های جدید معرفی شده برای شناسایی وب‌هرز

نویسندگان	سال	ویژگی
Wang et al. [77]	۲۰۰۷	تعداد لنگر متن[1] صفحه
		نرخ محتوای قابل دیدن
		نرخ استفاده از کلمات ترند جهانی در صفحه
		تکرار بیش از حد محتوا
Jakub et al. [73]	۲۰۰۸	نسبت تعداد ضمایر در صفحه به کل کلمات صفحه
		نرخ مجموع افعال یکتا و اسامی به تعداد کل کلمات صفحه
Martinez et al. [78]	۲۰۰۹	رابطه بین عنوان و محتوای آدرس اینترنتی با توضیحات برچسب در صفحه‌ای که به آن اشاره می‌کند.
Wang et al. [72]	۲۰۱۰	تاریخ آخرین به روززسانی
		نسبت پیوندهای ازکارافتاده به همه پیوندها
		نسبت مجموع طول کلمات به کل طول متن
Pavlov et al. [79]	۲۰۱۱	تعداد افعال زمان گذشته در متن
		میانگین تعداد نشانه‌های ویرایشی استفاده شده در متن
		نرخ کلمات با طول کمتر از ۳ و بیشتر از ۷ به کل کلمات
Prieto et al. [80]	۲۰۱۲	تعداد متا برچسب ها
		تعداد غلط‌های گرامری و دیکته‌ای
Karunakaran et al. [65]	۲۰۱۴	محاسبه میزان شباهت متن با استفاده از مدل زبانی Jansen-Shannon
Luckner et al. [61]	۲۰۱۴	نرخ واژه‌های بی معنی در متن
		مقدار شاخص مه شکن[2]
Hunagund et al. [66]	۲۰۱۵	تعداد کلمات توقف[3] استفاده شده در عنوان و متن
Kumar et al. [67]	۲۰۱۶	نرخ استفاده از کلمات معنی دار در متن
		وجود الگوهای تکرای یا کدهای مشابه
Singh et al. [44]	۲۰۱۷	چگالی استفاده از کلمات کلیدی
		نرخ مجموع فاکتورهای ارتباط[4] به کل کلمات صفحه

[1] Anchor Text
[2] Gunning Fog Index
[3] Stop Words
[4] Pertinence

۲-۵- روش‌های ترکیبی

در روش‌های ترکیبی هم به ارائه ویژگی جدید و هم طبقه کننده هماهنگ با آن پرداخته می‌شود. به عنوان مثال، الکعبی و همکاران در [81] به طراحی و پیاده‌سازی یک سامانه برخط شناسایی وب‌هرزهای زبان عربی پرداختند که هم از تحلیل محتوا و تحلیل پیوند و هم از بازخورد کاربر از طریق یک مرورگر استفاده می‌کند و همچنین دارای سیستمی جهت یادگیری و بهبود عملکرد با توجه به نظرات کاربر است.

ژانگ نیز از ویژگی‌های تحلیل محتوا و پیوند برای مقابله با وب‌هرز استفاده کرده است [82]. آن‌ها پروسه تکرارشونده‌ای طراحی کردند که کیفیت محتوا و پیوند را در دیگر صفحات وب توزیع می‌کند. تیان و همکاران نیز روشی مبتنی بر یادگیری ماشین ارائه کردند که در آن با استفاده از نظرات کاربران و الگوریتم‌های نیمه‌نظارتی اقدام به شناسایی وب‌هرز می‌شود [83]. چاکرابارتی و همکاران مدل DOM[1] را برای هر صفحه وب تشکیل داده و فهمیدند زیردرخت‌هایی که به نسبت سایر قسمت‌ها ارتباط بیشتری با موضوع جستجو دارند، رفتار ویژه‌ای نسبت به روند تقویت متقابل[2] نشان می‌دهند [84].

در [85] روشی پیشنهاد شده که از داده‌های مورد جستجوی کاربر برای تشخیص وب‌هرز استفاده شود. ایده به کار گرفته‌شده تشکیل یک گراف جستجو $g = (v, \varepsilon, \tau, \sigma)$ است که در آن v تعداد صفحات، ε ارتباطات بین لبه‌ها، τ زمان توقف و σ احتمال تصادفی پرش است. سپس برای هر صفحه یک مقدار با استفاده از الگوریتمی مشابه الگوریتم رتبه صفحه نسبت داده می‌شود. خصوصیت مهم راه حل پیشنهادی استفاده از پروسه مارکوف زمان پیوسته به عنوان زیربنای مدل است؛ زیرا از آنجایی که هرزفرست‌ها می-خواهند رتبه بالایی در صفحه نتایج موتور جستجو داشته باشند، بخش مهمی از ترافیک وب‌سایت‌های هرز

1. Document Object Model
2. Mutual Reinforcement

بر اساس بازدیدهای انجام‌شده‌از یک سایت جستجو است. همچنین کاربران به‌راحتی یک سایت هرز را شناسایی و سپس با سرعت آن را ترک می‌کنند. بنابراین داده‌های جستجوی کاربر شامل اطلاعات زمانی است.

۲-۶- جمع بندی

در این فصل به بررسی انواع روشهای تشخیص وب‌هرز پرداختیم. گفته شد که این روش‌ها به دو دسته کلی مبتنی بر گراف وب و مبتنی بر الگوریتم‌های یادگیری تقسیم می‌شوند. در روشهای مبتنی بر گراف وب، نکته کلیدی ایجاد نقشه ارتباطات بین صفحات بر اساس پیوندهای موجود و وزن دهی آنها است و به محتوای صفحات توجه زیادی نمی‌شود. به همین دلیل صفحات قدیمی اولویت بیشتری نسبت به صفحات جدید دارند چون ارتباطات کمتری دارند. از سوی دیگر این روش‌ها حساس به پرس و جوی کاربر هستند و به همین دلیل زمانبر هستند. اما در روشهای مبتنی بر یادگیری ماشین محتوا بسیار مهم است. این روش‌ها به دنبال ویژگی‌های تفکیک کننده و الگوریتم‌های با عملکرد بالا به منظور تشخیص صفحات وب‌هرز هستند.

تحقیقات نشان می‌دهد که رویکردهای مبتنی بر یادگیری ماشین به نتایج صحیح‌تری نسبت به روش‌های مبتنی بر گراف منجر می‌شود. اما نکته ای که در مورد این روش‌ها وجود دارد این است که آنها نیازمند آموزش دوباره هنگام روبرو شدن با داده‌های جدید هستند که بعضا زمانبر است. به همین دلیل دستیابی به دقت بالا در زمان اندک چالشی است که این روشها با آن روبرو هستند. همچنین مشکلات دیگری از قبیل بالابودن ابعاد فضای بردار ویژگی، کم بودن نمونه‌های آموزشی و در نتیجه نامتوازن بودن داده‌ها و بار محاسباتی بالای برخی روشها از دیگر چالش‌های این حوزه به شمار می‌رود.

با توجه به موارد اشاره شده، در این رساله قصد داریم به مسئله ویژگی و نقش آن در میزان دقت طبقه‌بندها بپردازیم و با ارائه ویژگی‌های جدید و نیز یک روش انتخاب ویژگی جدید، سعی در بهبود

عملکرد آن‌ها نماییم. همچنین روش جدیدی مبتنی بر ایمنی مصنوعی به منظور افزایش نرخ دقت تشخیص صفحات وب‌هرز ارائه شده‌است که به راحتی با ورود داده‌های جدید به‌روز می‌شود که شرح آن در فصل آتی آمده است.

۳

تئوری‌ها و روش پیشنهادی

روش‌های طبقه‌بندی

روش‌های انتخاب ویژگی

پردازش متن

سیستم ایمنی مصنوعی

۳-۱- مقدمه

همان طور که قبلا اشاره شد، در این رساله در پی آن هستیم که مدلی جهت شناسایی صفحات وب‌هرز ارائه کنیم. این مدل از سه بخش اصلی تشکیل شده‌است که شامل استخراج ویژگی، کاهش ویژگی و طبقه‌بند مناسب می‌باشد. در این فصل به بررسی تئوری‌ها و اصول پایه استفاده شده در این مدل و همچنین راهکار پیشنهادی در هر بخش خواهیم پرداخت.

۳-۲- استخراج ویژگی

همان‌طور که در فصل دوم اشاره شد، ویژگی هایی که از صفحات وب استخراج می‌شوند بر حسب منبعی که از آن به دست آمده اند، طبقه‌بندی می‌شوند. در این رساله ما تعدادی ویژگی مبتنی بر متن معرفی کرده‌ایم که می‌تواند به منظور بهبود نرخ تشخیص صفحات وب‌هرز مورد استفاده قرار گیرد. بعضی از این ویژگی‌ها مبتنی بر ساختار نشانی اینترنتی (URL) و برخی دیگر با توجه به برچسب‌های زبان نشانه‌گذاری فوق متن (HTML[1]) استفاده شده در کد منبع صفحات وب هستند. برای اینکه یک برآورد کلی از مفید بودن ویژگی‌های معرفی شده داشته باشیم، از تابعی به نام چگالی احتمال استفاده کردیم.

این تابع هر بردار ویژگی را یک بردار n بعدی با ویژگی‌های عددی در نظر میگیرد که به صورت تصادفی ایجاد شده‌اند و توزیع آنها بستگی به ذات و خاصیت نمونه‌ای دارد که نمایش دهنده آن هستند. یک متغیر تصادفی $X: \Omega \to E$ یک تابع قابل اندازه‌گیری روی یک مجموعه از خروجی‌های ممکن Ω در یک فضای قابل اندازه‌گیری E است. احتمال اینکه X مقداری در مجموعه قابل اندازه‌گیری $S \subseteq E$ باشد، بر اساس فرمول (۳-۱) به دست می‌آید.

$$\Pr(X \in S) = P(\{\omega \in \Omega | X(\omega) \in S\}) \qquad (۱-۳)$$

[1] Hyper Text Markup Language

در طبقه‌بندی، یک ویژگی خوب ویژگی است که احتمال رخداد آن در بازه‌های مشخص و مجزا برای هر دسته زیاد است. تابع چگالی احتمال به همین منظور استفاده می‌شود و فرمول آن به شرح ذیل می‌باشد.

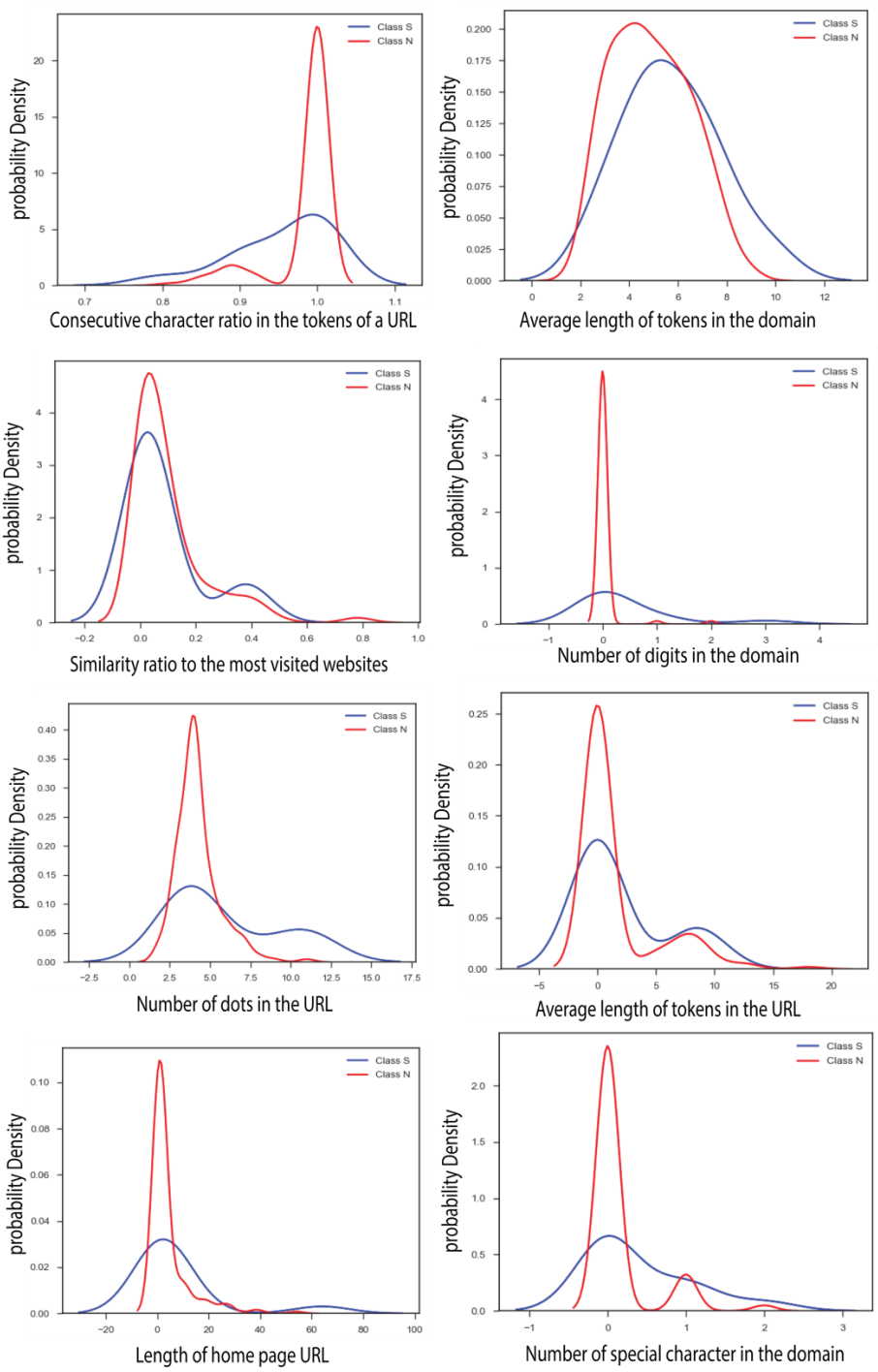
$$P(a \leq X \leq b) = \int_a^b f(x)dx \qquad (۲-۳)$$

با دانستن چنین اطلاعاتی میتوان یک بازه با بیشترین احتمال رخداد برای هر ویژگی یافت. اگر این بازه برای داده‌های متعلق بر هر دسته، مجزا و مشخص باشد، آن ویژگی می‌تواند به عنوان یک ویژگی مناسب برای طبقه‌بندی در نظر گرفته شود. به همین دلیل ما از این تابع به منظور یافتن ویژگی‌های مناسب جهت طبقه‌بندی استفاده کردیم.

۳-۲-۱- ویژگی‌های مبتنی بر ساختار نشانی اینترنتی

همان طور که در [86] توضیح داده شده، ویژگی‌های لغوی که در نشانی اینترنتی وجود دارد، منبع بسیار خوبی جهت شناسایی وب‌هرز است. تعداد زیردامنه‌ها، طول URL و عبارتهای موجود در آن اطلاعات خوبی را در اختیار طبقه‌بند قرار می‌دهد تا بین صفحات به هنجار[1] و وب‌هرز تفاوت قائل شود. تابع چگالی احتمال این ویژگی‌ها در شکل ۳-۱ نمایش داده شده که توضیح آن به شرح ذیل است.

[1] Normal



شکل ۳-۱ توزیع چگالی احتمال برای ویژگی‌های مبتنی بر URL [87]

۳-۲-۱-۱- تعداد زیردامنه‌ها

هرزفرست‌ها معمولا برای ایجاد چندین وب سایت بر روی یک دامنه، چند زیردامنه می‌سازند. این کار به منظور کاهش هزینه خرید چندین دامنه و هزینه آبونمان میزبانی[۱] است. بنابراین صفحات هرز اغلب روی یک زیردامنه میزبان می‌شوند و لذا تعداد نقاط (".") موجود در آدرس اینترنتی، شاخص خوبی برای شناسایی این صفحات است. آزمایشات (فصل چهارم) نشان می‌دهد که بیش از ۷۰٪ آدرس‌های اینترنتی حاوی بیش از دو و کمتر از پنج نقطه هستند. به همین دلیل اگر آدرسی حاوی بیش از ۵ نقطه باشد، به احتمال زیاد وب‌هرز است.

۳-۲-۱-۲- تعداد ارقام و نشانه‌های خاص در دامنه و آدرس اینترنتی

هرچه نام دامنه بیشتر حاوی ارقام یا نویسه‌های خاص[۲] باشد، کمتر به اسامی کاربرپسند[۳] نزدیک است و این احتمال بیشتر وجود دارد که توسط نرم‌افزارهای خاص و به طور خودکار و در جهت ایجاد مزرعه پیوند[۴] ایجاد شده باشد. آزمایشات نشان می‌دهد که صفحات به‌هنجار، رقم یا نشانه خاص در نشانه‌های[۵] دامنه کمتر دیده می‌شود.

۳-۲-۱-۳- طول دامنه و آدرس اینترنتی

برخلاف صفحات به‌هنجار که سعی دارند از عبارات کوتاه برای دامنه استفاده کنند تا سریعتر و راحت‌تر در خاطر کاربران بمانند، صفحات هرز علاقه دارند که آدرس‌های اینترنتی طولانی استفاده کنند. چرا که این امر به آنها فضای بیشتری برای گنجاندن کلمات کلیدی در URL می‌دهد. به عنوان مثال در آدرس اینترنتی http://www.buycheapextralongshowercurtains.com ، کلمات buy و cheap صرفا جهت بالا بردن رتبه صفحه در آدرس قید شده‌اند و نوعی فریبکاری است. آزمایشات نیز نشان می‌دهد که بیشتر صفحات به‌هنجار طول آدرس اینترنتی کمتر از ۲۰ دارند.

---

[1] Hosting Charge
[2] Special Character
[3] User Friendly
[4] Link Farm
[5] Token

۳-۲-۱-۴- میانگین طول نشانه‌ها در دامنه و آدرس اینترنتی

یک نشانی اینترنتی به اجزائی مانند دامنه، مسیر و مولفه‌های پرس و جو[1] تقسیم می‌شود. نشانه‌های یک آدرس اینترنتی از قطعه قطعه کردن آن توسط نویسه‌هایی مثل نقطه و ممیز (. و /) به دست می‌آید. در یک صفحه به‌هنجار، نشانه‌ها اغلب کلماتی بامعنی هستند و طول آنها در یک بازه مشخص قرار دارد اما در صفحات هرز کلمات به منظور انباشتن کلمات کلیدی و افزایش رتبه کلمات به یکدیگر متصل می‌شوند و در نتیجه نشانه‌ها طولانی و بی معنی هستند. آزمایشات نشان می‌دهد که میانگین طول نشانه‌ها در دامنه و آدرس اینترنتی کمتر از صفحات وب‌هرز است.

۳-۲-۱-۵- نرخ نویسه‌های متوالی در دامنه و آدرس اینترنتی

هرزفرست‌ها معمولا دامنه‌ها را به طور حجمی و توسط نرم‌افزارهای خودکار ثبت می‌کنند. وب سایتهایی که این دامنه‌ها میزبانشان هستند برای کاربران مناسب نیستند چرا که آنها برای خزشگرهای موتورهای جستجو طراحی شده‌اند تا رتبه بالاتری به دست بیاورند. در نتیجه آنها فقط برای مدت زمان کوتاهی دردسترس هستند. این نوع از دامنه‌ها معمولا حاوی ترکیبی از حروف و اعداد هستند که عبارات بی معنی را ایجاد می‌کنند. بنابراین یک نام دامنه حاوی یک نشانه با ترکیبی از اعداد و حروف الفبا به احتمال زیاد وب‌هرز است. به منظور تشخیص چنین دامنه‌هایی یک شاخص به نام "نرخ نویسه‌های متوالی"[2] معرفی کرده‌ایم. این شاخص طول بزرگترین زیررشته حاوی حروف الفبا را بر طول نشانه‌ای که حاوی آن است تقسیم می‌کند. هرچه این عدد بزرگتر و به یک نزدیکتر باشد، احتمال تولید خودکار آن کمتر است.

۳-۲-۱-۶- نرخ میزان شباهت شناسه‌های دامنه به اسامی دامنه‌های پربازدید

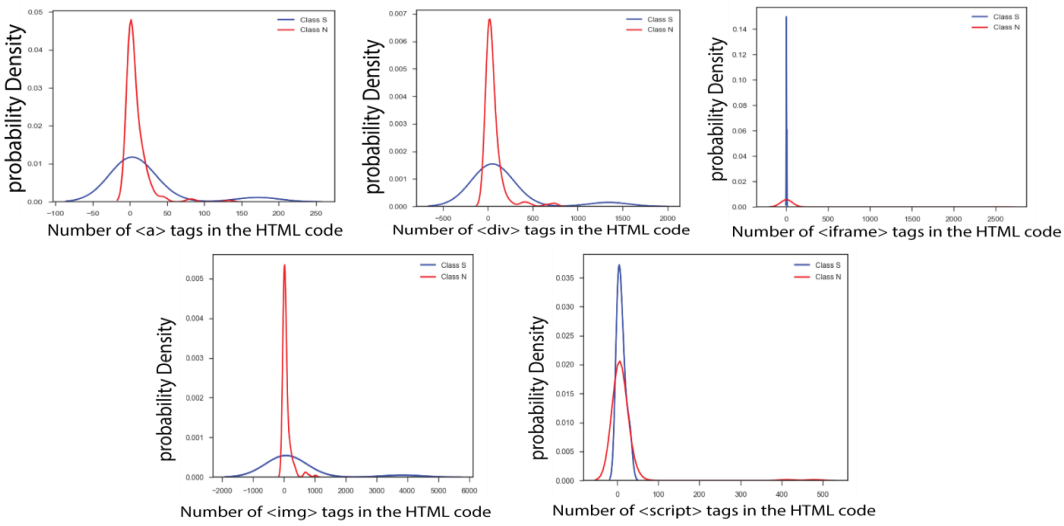
یکی از حقه‌هایی که هرزفرست‌ها به کار می‌برند، استفاده از اسامی دامنه‌های پرطرفدار به عنوان زیردامنه یا زیررشته‌های نشانه‌های یک آدرس اینترنتی است. دلیل این امر آن است که خیلی از

[1] Query Parameters
[2] Consecutive Character Ratio

کاربران عادی ممکن است تصور کنند که آدرس اینترنتی مانند 'microsoft.com.phishy.net' متعلق به سایت microsoft.com است. بنابراین شباهت نشانه‌های یک آدرس اینترنتی به نشانه‌های دامنه‌های معروف می‌تواند شاخص خوبی به منظور تشخیص صفحات وب‌هرز باشد. معیار اینکه یک سایت را سایت معروفی بدانیم، رتبه آن در سایت Alexa است که معتبرترین سازمان جهان در تعیین رتبه وب‌سایتها براساس ترافیک آنها است. در این رساله ما میزان شباهت شناسه‌های آدرس اینترنتی را با اسم ۵۰۰ وبسایت برتر دنیا مقایسه کردیم. نتایج نشان می‌دهد که نشانه‌های URL صفحات به‌هنجار به ندرت حاوی نام وب‌سایتهای معروف است.

۳-۲-۲- ویژگی‌های مبتنی بر برچسب‌های کد HTML صفحات وب

کد HTML یک صفحه وب انتخاب مناسبی برای استخراج ویژگی است. استفاده کمتر یا بیشتر از حد معمول برخی عناصر طراحی صفحات در وب‌هرزها باعث افزایش یا کاهش میزان استفاده از برخی از برچسب‌های کد HTML می‌شود. به همین دلیل این برچسب‌ها می‌تواند شاخص خوبی برای تشخیص صفحات هرز باشد. برای تشخیص میزان تاثیر هر یک از این برچسب‌ها در تشخیص وب‌هرز، از تابعی به نام چگالی احتمال استفاده کردیم که در شکل ۳-۲ نمایش داده شده‌است.



شکل ۳-۲ توزیع چگالی احتمال ویژگی‌های مبتنی بر کدهای HTML [87]

- **A/ Link**

در زبان HTML، برچسب‌های <A> و <link> برای ایجاد پیوند به یک منبع خارجی یا سند مورد استفاده قرار می‌گیرد. برچسب <Link> به منظور تعریف یک پیوند بین صفحه جاری و یک منبع خارجی است که می‌تواند یک صفحه یا فایل باشد این در حالی است که برچسب <A> برای ایجاد یا تعریف پیوند به داخل صفحه استفاده می‌شود. تفاوت دیگر این دو برچسب در مکانی است که مورد قرار می‌گیرند. برچسب <link> در بخش <head> و برچسب <A> در بخش <body> کد صفحه استفاده می‌شود. از آنجا که در صفحات هرز تعداد پیوندها به صفحات دیگر بیشتر از صفحات به‌هنجار است، تعداد این برچسب‌ها شاخص خوبی برای تشخیص وب‌هرز به شمار می‌رود.

- **Div**

برچسب به منظور تعریف یک قسمت یا بخش در یک سند HTML استفاده می‌شود. این برچسب مثل یک ظرف عناصر صفحه را بسته‌بندی می‌کند و سند HTML را به بخش‌های مختلف تقسیم می‌کند. توسعه‌دهندگان وب از برچسب برای گروه بندی عناصر صفحه استفاده می‌کنند تا به آسانی بتوانند سبک‌های CSS[1] را در آن واحد به تعداد زیادی از عناصر اعمال کنند. آزمایشات نشان می‌دهد که این برچسب به ندرت در صفحات به‌هنجار استفاده می‌شود.

- **Iframe**

این برچسب به منظور ایجاد یک قاب[2] داخلی برای جاسازی کردن تعدادی سند در صفحه HTML جاری است. برچسب <Iframe> یک منطقه مستطیلی شکل در صفحه ایجاد می‌کند که مرورگر می‌تواند آن را به صورت یک صفحه جداگانه نمایش دهد. آزمایشات نشان داده‌اند که این برچسب در صفحات وب‌هرز به ندرت استفاده می‌شوند.

[1] Cascading Style Sheets
[2] Frame

- **Img**

این برچسب برای تعریف تصویر در یک صفحه HTML به کار می‌رود. در واقع این برچسب یک پیوند بین صفحه و فایل حاوی تصویر ایجاد می‌کند. آزمایشات نشان می‌دهد که صفحات هرز تمایل دارند که از تصاویر بیشتری استفاده کنند.

- **Script**

این برچسب برای تعریف یا استفاده از اسکریپت نویسی در صفحه وب استفاده می‌شود و حاوی کد برنامه‌نویسی است (Java Script). برنامه‌های کاربردی رایج JavaScript شامل مدیریت تصویر، اعتبارسنجی فرم و تغییرات پویای محتوا است. صفحات هرز از کدهای JavaScript برای هدایت بازدیدکنندگان به صفحات دیگر یا نمایش آگهی عبارات جلب توجه کننده استفاده می‌کنند. به همین دلیل استفاده از این برچسب در این صفحات بیشتر از صفحات به‌هنجار است.

۳-۲-۳- ویژگی‌های مبتنی بر انسجام متن

یکی از ویژگی‌های اصلی وب‌هرزها، کیفیت پایین محتوای آن از لحاظ نگارشی و معنایی است. چرا که معمولا این صفحات از مطالب موجود در وبسایت‌های دیگر استفاده می‌کنند و بدون توجه به ارتباطشان با یکدیگر، آنها را پشت سرهم قرار می‌دهند و یا اینکه به خاطر استفاده از تکنیک انباشتگی کلمات کلیدی، حاوی تعدادی کلمه هستند که بدون رعایت ساختارهای نگارشی در کنار هم قرار گرفته‌اند. به همین دلیل انسجام متن می‌تواند شاخص خوبی برای ایجاد تمایز بین صفحات به‌هنجار و وب‌هرز باشد. بدین منظور در این رساله ما میزان انسجام ساختاری (پیوستگی) و انسجام معنایی (همبستگی) متن را اندازه می‌گیریم و از آن به عنوان ویژگی برای تشخیص وب‌هرز استفاده می‌کنیم.

زبان شناس مشهور، هالیدی، معتقد است آنچه متن را یکپارچه و از نوشته‌های پراکنده متمایز می سازد، انسجام است. انسجام متن حاصل روابط جمله‌های آن با یکدیگر است. انسجام مفهومی است که

به پیوستگی و اتصال جمله‌های نوشته یا گفته اشاره دارد. اتصال جمله‌ها یا روابط عوامل آن با یکدیگر هنگامی پدید می‌آید که تعبیر عاملی در یک بخش از کلام به عاملی دیگر در همان کلام وابسته باشد. اینجاست که رابطهٔ انسجامی شکل می‌گیرد و این دو عامل سبب شکل‌گیری متن می‌شوند. بر همین اساس ایشان ابزارهای آفریننده انسجام متن را در زبان انگلیسی به سه دسته تقسیم کردند: ابزارهای دستوری شامل ارجاع، جایگزینی و حذف؛ ابزارهای پیوندی که عبارت است از حرف ربط؛ ابزارهای واژگانی شامل تکرار و هم آیی [88]. هر چه تعداد این ابزارها در یک متن بیشتر باشد، متن از انسجام و خوانایی بیشتری برخوردار است. بنابراین تعداد کلمات مرتبط با هر یک از این ابزارها در متن می‌تواند به عنوان یک ویژگی خوب برای محاسبه میزان انسجام مورد استفاده قرار گیرد. همچنین می‌توان میزان تاثیرگذاری هر یک از این ابزارها را بر انسجام و خوانایی متن سنجید و از آنها به صورت وزن دار استفاده کرد.

با این حال یک متن میتواند همبسته باشد بدون اینکه پیوسته باشد. به همین دلیل پیوستگی متن تضمینی بر همبستگی آن نیست. پیوستگی به وسیله ارتباطات دستوری و واژگانی بین جملات تعیین می‌شود در حالی که همبستگی ناشی از ارتباطات معنایی جملات است. به عبارت دیگر، همبستگی یک ویژگی معنایی سخن است که از طریق تفسیر هر جمله نسبت به جملات دیگر برقرار می‌شود. در این تعریف کلمه "تفسیر" به تعامل بین متن و خواننده اشاره دلالت می‌کند. به همین دلیل یکی از روش‌های ارزیابی همبستگی یک متن، بررسی و تحلیل ساختار موضوعی آن است [89].

در این رساله ما فرض می‌کنیم که وجود همبستگی در یک متن باعث ایجاد محدودیت در تعداد موضوعات مورد بحث در متن می‌شود. به این معنا که موضوعات پاراگرفهای متوالی به یکدیگر مرتبط و متعلق به یک حوزه هستند. به همین منظور، تعداد موضوعات یک متن یک ویژگی است که می‌تواند برای شناسایی صفحات وب‌هرز استفاده شود. تخصیص دیریکله پنهان[1] (LDA) یک الگوریتم محبوب

[1] Latent Dirichlet Allocation

برای تعیین تعداد مباحث موجود در متن است. این الگوریتم موضوعات پنهان در مجموعه نوشته‌ها را با مشخص کردن احتمال ظهور هر موضوع در متن ورودی مشخص می‌کند.

۳-۲-۳-۱- تخصیص دیریکله پنهان (LDA)

الگوریتم‌های مدل‌سازی موضوعی روش‌های آماری هستند که کلمات درون یک متن را تحلیل کرده و از این طریق موضوعات داخل متون را استخراج می‌کنند. همچنین ارتباط این موضوعات با یکدیگر و نیز تغییر آن‌ها در طول زمان را مشخص می‌کنند. این الگوریتم‌ها نیازی به هیچ فرض اولیه‌ای در مورد موضوعات متون و یا برچسب‌گذاری متون ندارند؛ بلکه ورودی آن‌ها متن اصلی است. برای پیاده‌سازی مدل‌سازی موضوعی دو روش اصلی LDA و LSA[1] وجود دارد.

روش LDA ساده‌ترین روشی است که در این زمینه وجود دارد. این روش بر دو فرض استوار است. اول اینکه اسناد دارای موضوعات متفاوتی هستند که می‌توان آن‌ها را با توجه به کلمات موجود در متن شناسایی کرد. دوم آنکه هر سند از کلماتی تشکیل شده‌است که هریک متعلق به یک موضوع بوده و نسبت تعداد کلمات مربوط به یک موضوع مشخص به کل کلمات سند، با یکدیگر متفاوت است. با توجه به این نسبت‌ها است که می‌توانیم آن متن را در یک موضوع خاص طبقه‌بندی کنیم.

مجموعهٔ ثابتی از کلمات را به عنوان واژه‌نامه در نظر می‌گیریم. روش LDA فرض می‌کند هر موضوع توزیعی روی این مجموعه کلمات است؛ یعنی کلمات مربوط به یک موضوع، در آن موضوع دارای احتمال بالایی هستند. فرض می‌کنیم این موضوعات از قبل مشخص شده‌اند. حال برای هر سند از مجموعه اسنادی که در اختیار داریم کلمات در دو مرحله تولید می‌شوند:

- به طور تصادفی توزیع احتمالی روی موضوعات انتخاب شود.
- برای هر کلمه از سند:

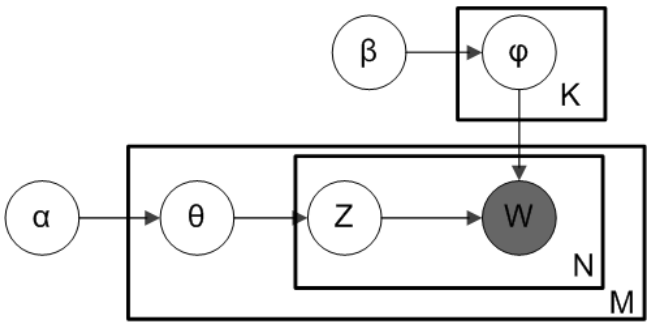
[1] Latent Semantic Analysis

- به طور تصادفی موضوعی با استفاده از توزیع احتمال مرحله قبل مشخص می‌شود.
- به طور تصادفی کلمه‌ای از واژه‌نامه با توزیع احتمال مشخص‌شده‌انتخاب می‌شود.

در این مدل احتمالی، اینکه هر سند از چندین موضوع تشکیل شده‌است، منعکس می‌شود. مرحله نخست انعکاس سهیم بودن موضوعات مختلف در یک متن با نسبت‌های متفاوت است. قسمت دوم مرحله دوم هم بیانگر این است که هر کلمه در هر سند از یکی از موضوعات استخراج شده‌است. قسمت نخست مرحلهٔ دوم هم بر این نکته تأکید دارد که این موضوع حاصل توزیع احتمال موضوعات روی اسناد انتخاب‌شده‌است. گفتنی است که در این روش همهٔ اسناد مجموعهٔ یکسانی از موضوعات را در اختیار دارند، ولی هر سند این موضوعات را با نسبت‌های متفاوتی ارائه می‌کند. این نسبت‌ها از رابطه ۳-۳ محاسبه می‌شود.

$$P(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}) = \prod P(\beta_i) \prod P(\theta_d) \left(\prod P(z_{d,n}|\theta_d)\ P(w_{d,n}|\beta_{1:K}, z_{d,n})\right) \quad (۳-۳)$$

در این رابطه، موضوعات ما $\beta_{1:k}$ هستند که هر β_i یک توزیع احتمال روی مجموعهٔ کلمات است. نسبت موضوعات برای اسناد با پارامتر θ مشخص می‌شود. بدین معنی که $\theta_{d,k}$ بیانگر نسبت موضوع k در سند d است. پارامتر z بیانگر تخصیص موضوع است. $z_{d,n}$ موضوع تخصیص داده‌شده به کلمهٔ-n اُم از سند d است. W کلمات مشاهده شده‌است که $w_{d,n}$ کلمه‌ی-n اُم از سند d است. شمای گرافیکی این الگوریتم را در شکل ۳-۳ مشاهده می‌کنید.



شکل ۳-۳ شمای گرافیکی الگوریتم LDA [90]

همان طور که مشاهده می‌شود از این رابطه می‌توان به مجموعه‌ای از وابستگی‌ها پی برد. مثلاً موضوع تخصیص داده‌شده به هر کلمه در یک سند ($z_{d,n}$) بستگی به نسبت موضوعات برای آن سند (θ_d)داشته و یا کلمهٔ مشاهده‌شده در هر سند ($w_{d,n}$) بستگی به موضوع اختصاص داده‌شده به آن کلمه ($z_{d,n}$) و همهٔ موضوعات ($\beta_{1:k}$) دارد؛ یعنی $z_{d,n}$ موضوع مربوط به آن کلمه را مشخص می‌کند و بعد احتمال حضور آن کلمه در آن موضوع یعنی $w_{d,n}$ مشخص می‌شود.

اکنون می‌خواهیم احتمال موضوعات مشخص‌شده را حساب کنیم؛ یعنی طبق گفته‌های قبل با وجود متغیرهای آشکار، متغیرهای پنهان با چه احتمالی حضور دارند. این امر با استفاده از رابطه ۴-۳ محاسبه می‌شود:

$$P(\beta_{1:K}, \theta_{1:D}, z_{1:D}|w_{1:D}) = \frac{P(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D})}{P(w_{1:D})} \qquad (۳-۴)$$

صورت کسر با توجه به رابطه ۳-۳ محاسبه می‌شود. مخرج کسر هم به‌راحتی از کلمات مشاهده‌شده به دست می‌آید. تعداد ساختارهای موضوعی پنهان بسیار زیاد است و محاسبهٔ آن به‌راحتی امکان‌پذیر نیست. این موضوع به علت صورت کسر این رابطه است. به همین علت برای محاسبهٔ این عبارت آن را تخمین می‌زنند. الگوریتم‌های مدل‌سازی موضوعی برای تخمین این عبارت، از یک توزیع احتمال دیگر استفاده می‌کنند که نزدیک به مقدار واقعی باشد. در LDA از توزیع دریکله برای این کار استفاده می‌شود.

۳-۲-۳-۱- ویژگی‌های مبتنی بر پیوستگی

در این بخش ما تاثیر برخی از ادات پیوستگی را که هزینه محاسباتی پایینی دارند در تشخیص صفحات وب‌هرز بررسی می‌کنیم. موارد بررسی شده شامل تعداد کلمات، جملات، پاراگرافها و نیز تعداد ادات ربط در متن است. لیست کامل این ویژگی‌ها در جدول ۳-۱ درج شده‌است.

جدول ۳-۱ ویژگی‌های مبتنی بر پیوستگی

توضیح	مثال
نرخ ادارت ربط پایه به همه کلمات	for, and, nor
نرخ ادات ربط به همه کلمات	and, but
نرخ کلمات فصل به همه کلمات	Or
نرخ حروف ربط وابسته به کل کلمات	after, although, as
نرخ حروف ربط همپایه به کل کلمات	yet, so, nor
نرخ کلمات وصل به کل کلمات	and, also, besides
نرخ کلمات متصل کننده جملات به کل کلمات	nonetheless, therefore, although
نرخ کلمات ترتیبی به کل کلمات	to begin with, next, first
نرخ کلمات مشخص کننده دلیل و هدف به کل کلمات	therefore, that is why, for this reason
نرخ کلمات ضدیت به کل کلمات	but, however, nevertheless
نرخ کلمات معین به کل کلمات	a, an, the
نرخ ادات ربط سببی مثبت به کل کلمات	arise, because, enabling
نرخ ادات ربط منطقی مثبت به کل کلمات	actually, after all, all in all
نرخ ادات ربط زمانی به کل کلمات	a consequence of, after, again
نرخ ادات ربط مثبت هدف و علت به کل کلمات	by, desire, desired
نرخ ادات ربط مثبت به کل کلمات	actually, after, again
نرخ ضمایر اشاره به کل کلمات	this, that, these
نرخ ادات ربط افزایشی به کل کلمات	after all, again, all in all
نرخ ادات ربط سببی به کل کلمات	although, arise, arises

۳-۲-۳-۲- ویژگی‌های مبتنی بر همبستگی

همان طور که قبلا گفته شد، یکی از روش‌های تعیین همبستگی متن، تعداد موضوعاتی است که در آن متن مورد بحث قرار گرفته است. به منظور تعیین تعداد موضوعات متن روشهای مختلفی از قبیل مدل فضای بردار (VSM[۱])، شاخص معنایی نهفته (LSI[۲])، تحلیل احتمالی معنایی نهفته (PLSA[۳]) و تخصیص دیریکله نهفته (LDA) وجود دارد. SVD یک مدل ساده بر مبنای جبر خطی است که امکان تعیین میزان ارتباط بین پرس و جو و سند را فراهم می‌کند. با این وجود این مدل ابعاد بسیار زیادی دارد زیرا از برداری استفاده می‌کند که خلوت است. به همین دلیل استفاده از

[1] Vector Space Model
[2] Latent Semantic Indexing
[3] Probabilistic Latent Semantic Analysis

شباهت کسینوس می‌تواند پراشتباه و نادرست باشد. همچنین این مدل می‌بایست با مساله ترادف و تعدد معانی روبرو شود. LSI می‌تواند مساله ترادف را تا حدی برطرف کند و سند را به یک فضای کم بعد نگاشت دهد. به منظور انجام سریع کاهش ابعاد، این روش از تجزیه ماتریس عبارت-سند استفاده می‌کند. با این حال این کار برای حل مساله تعدد معانی موثر نیست، مضاف بر اینکه به دلیل استفاده از تجزیه بردار یکه[1] (SVD)، بار محاسباتی بالایی دارد. مشکل دیگر این روش سختی به‌روزرسانی آن هنگام ظهور یک سند جدید است. در روش PLSA با استفاده از روش‌های احتمالی به جای ماتریس و در نظر گرفتن موضوع به صورت توزیعی از کلمات، مساله تعدد معانی حل شده‌است. اما مشکلی که وجود دارد این است که تعداد پارامترها به طور خطی نسبت به تعداد اسناد افزایش می‌یابد. در روش LDA به جای استفاده از ماتریس از توزیع احتمالی پیشین دیریکله به عنوان توزیع‌های موضوع-کلمه و سند-موضوع استفاده می‌شود. این امر از تناسب بیش از حد[2] جلوگیری می‌کند. این مدل بار محاسباتی پایینی دارد و می‌تواند با ظهور سند جدید به راحتی به‌روزرسانی شود [91]. بنابراین همان‌طور که پیش از این اشاره شد، در این رساله با استفاده از LDA تعداد موضوعات یک صفحه را به عنوان ویژگی برای تشخیص وب‌هرز استخراج کردیم.

قبل از استفاده از LDA، لازم است مشخص کنیم چند موضوع در مجموعه داده وجود دارد. برای تعیین تعداد بهینه موضوعاتی که باید توسط LDA استخراج شوند، معمولا از امتیاز انسجام موضوع[3] استفاده می‌شود (رابطه ۵-۳) در این رابطه w_i و w_j نشان‌دهنده ارزشمندترین[4] کلمات یک موضوع است.

$$CoherenceScore = \sum_{i<j} score(w_i, w_j) \quad (۵-۳)$$

$$Score(w_i, w_j) = log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \quad (۶-۳)$$

[1] Singular Vector Decomposition
[2] Over-Fitting
[3] Coherence Score
[4] Top

احتمالات بر مبنای تعداد همرخدادی[1] کلمات تخمین زده می‌شوند. برای این کار ابتدا تعدادی سند با استفاده از یک پنجره لغزان که روی ویکی پدیا (به عنوان یک منبع نوشته خارجی) حرکت می‌کند ساخته شده‌است (هر سند یکی از موقعیتهای پنجره است). سپس تعداد این همرخدادی‌ها در این سندها محاسبه می‌شود. پس از انتخاب تعداد موضوع مناسب، این ویژگی‌ها از مجموعه داده استخراج می‌شود:

- تعداد موضوعات با احتمال بیشتر از ۱٪
- تعداد موضوعات با احتمال بیشتر از ۱۰٪
- تعداد موضوعات با احتمال بیشتر از ۲۰٪
- تعداد موضوعات با احتمال بیشتر از ۳۰٪
- تعداد موضوعات با احتمال بیشتر از ۴۰٪

۳-۳- انتخاب ویژگی

مسأله انتخاب ویژگی تلاش می‌کنند تا از میان ۲^N زیرمجموعه ممکن از کل ویژگی‌ها، زیرمجموعه‌ای را انتخاب کنند که بتواند مقدار یک تابع ارزیابی را بهینه کند. روش‌های مختلف انتخاب ویژگی شامل دو جزء اصلی الگوریتم جستجو[2] و تابع ارزیابی‌کننده[3] هستند. در بعضی از الگوریتم‌های جستجو تمام فضای ممکن و در برخی دیگر که می‌تواند مکاشفه‌ای یا جستجوی تصادفی باشد، در ازای از دست دادن مقداری از کارایی، فضای کوچکتری جستجو می‌شود. همچنین توابع ارزیابی‌کننده متعددی نیز وجود دارد. یک تابع ارزیابی، میزان خوب بودن زیرمجموعه تولید شده را محاسبه کرده و حاصل را با مقدار بهترین زیرمجموعه قبلی مقایسه می‌کند. اگر زیرمجموعه جدید، بهتر از زیرمجموعه‌های قدیمی باشد، زیرمجموعه جدید به عنوان زیرمجموعه بهینه، جایگزین قبلی می‌شود [92]. در این رساله برای انتخاب بهترین زیرمجموعه از میان ویژگی‌های موجود، ابتدا ۱۱

[1] Co-Occurance
[2] Search Method
[3] Evaluator Function

روش جستجو و هشت تابع ارزیاب رایج مورد بررسی قرار گرفته و در نهایت یک روش انتخاب ویژگی که کارایی بهتری در این زمینه دارد برای انتخاب ویژگی استفاده شد.

۳-۳-۱- روش‌های جستجو

وظیفه الگوریتم جستجو که به آن تابع تولید کننده نیز گفته می‌شود، ایجاد زیرمجموعه‌های مختلف از کل ویژگی‌ها است. الگوریتم‌های جستجو یا بدون ویژگی و یا با مجموعه تمام ویژگی‌ها شروع به کار می‌کنند. در حالت اول ویژگی‌ها به ترتیب به مجموعه اضافه می‌شوند و زیرمجموعه‌های جدید را تولید می‌کنند. به این دسته روش‌ها، روش‌های پائین به بالا می‌گویند. در حالت دوم از یک مجموعه شامل تمام ویژگی‌ها، شروع می‌کنیم و به مرور و در طی اجرای الگوریتم، ویژگی‌ها را حذف می‌کنیم تا به زیرمجموعه دلخواه برسیم. روش‌هایی که به این صورت عمل می‌کنند، روش‌های بالا به پائین نام دارند. در ادامه به شرح برخی از این الگوریتم‌ها می‌پردازیم.

- **حریصانه**

یکی از استراتژی‌های جستجو، روش حریصانه[۱] است که فضای ویژگی را به صورت جلو رفت[۲] یا عقب‌گرد[۳] جستجو می‌کند. این الگوریتم عمل جستجو را از یک نقطه دلخواه آغاز می‌کند و نقطه توقف آن زمانی است که اضافه یا حذف یک ویژگی باعث کاهش مقدار تابع ارزیابی شود. همچنین این الگوریتم می‌تواند بر اساس پیمایش فضا از یک سو به سوی دیگر و ذخیره ترتیب ویژگی‌های انتخاب شده، لیستی از ویژگی‌ها ایجاد کند که بر اساس میزان مفید بودنشان رتبه‌بندی شده‌اند.

[1] Greedy
[2] Feed Forward
[3] Backward

- **اول بهترین**

روش جستجوی اول بهترین[1] یک روش انتخاب حریصانه است که تلاش می‌کند کارایی زیرمجموعه فعلی را با افزودن یا حذف کلیه ویژگی‌های ممکن به مجموعه بهتری تبدیل کند. این الگوریتم می‌تواند با مجموعه‌ای خالی از ویژگی شروع به جستجو نماید و به جلو رود یا با مجموعه‌ای کامل از ویژگی‌های شروع کند و عقب‌گرد داشته باشد یا از هر نقطه‌ای شروع کند و در دو جهت جلو و عقب ادامه دهد. در صورتی که هیچ محدودیتی برای ادامه الگوریتم وجود نداشته باشد، این روش یک جستجوی جامع محسوب می‌شود [92].

- **خطی رو به جلو**

الگوریتم خطی روبه جلو[2] نوعی روش اول بهترین است که ابتدا بر اساس یک ارزیاب، ویژگی‌ها را به طور انفرادی بررسی کرده و آن‌ها را رتبه‌بندی می‌کند. سپس K ویژگی اول را انتخاب کرده و بر روی آن‌ها عمل جستجو را انجام می‌دهد به این صورت که با یک مجموعه خالی شروع می‌کند، سپس در هر تکرار یک ویژگی را با استفاده از تابع ارزیابی به مجموعه جواب اضافه می‌کند. این کار را تا زمانی تکرار می‌شود که تعداد ویژگی لازم انتخاب شود [92].

- **روبه‌جلو با تعیین اندازه زیرمجموعه**

این روش توسعه یافته الگوریتم خطی رو به جلو است. در این نسخه از الگوریتم، در قسمت جستجو، یک اعتبارسنجی داخلی انجام می‌شود (نقطه شروع و تعداد دسته‌ها می‌تواند از قبل مشخص شود) تا اندازه زیرمجموعه بهینه مشخص شود (با استفاده از ارزیابی کننده‌ای که در ابتدا مشخص شده‌است). در نهایت یک بار دیگر الگوریتم خطی رو به جلو بر اساس اندازه زیرمجموعه بهینه بر روی تمام داده‌ها اعمال می‌شود [93].

[1] Best First
[2] Linear Forward Selection

- **رتبه‌بندی**

در این روش ارزش ویژگی‌ها به تنهایی و بدون در نظر گرفتن رابطه بین آن‌ها سنجیده می‌شود. خروجی این الگوریتم یک لیست رتبه‌بندی شده‌از ویژگی‌های به ترتیب اهمیت است. به همین علت این روش بیشتر در کنار توابع ارزیابی مانند ریلیف[1]، نرخ سود[2]و انتروپی[3] استفاده می‌شود [92].

- **جستجوی پراکنده**

جستجوی پراکنده[4]یک الگوریتم تکاملی است و شامل مجموعه راه‌حل‌هایی است که بر اساس مکانیزم ترکیبی بین آن‌ها تکامل یافته‌اند. این الگوریتم در بین زیرمجموعه‌های ویژگی یک جستجوی پراکنده انجام می‌دهد. شروع جستجو با تعداد زیادی زیرمجموعه‌های متنوع و حائز اهمیت است و الگوریتم زمانی پایان می‌یابد که نتیجه از مقدار آستانه مدنظر بیشتر شود یا امکان بهبود بیشتری نباشد [94].

- **الگوریتم ژنتیک**

در این روش یک دسته ویژگی از زیرمجموعه ویژگی‌های کاندید شده تولید می‌کنیم. در هر بار تکرار الگوریتم، با استفاده از عملگرهای جهش و بازترکیبی بر روی عناصر دسته قبلی، عناصر جدیدی تولید می‌کنیم. با استفاده از یک تابع ارزیابی، میزان شایستگی عناصر دسته فعلی را مشخص کرده و عناصر بهتر را به عنوان دسته نسل بعد انتخاب می‌کنیم. اگرچه یافتن بهترین جواب در این روش تضمین نمی‌شود؛ ولی همیشه یک جواب نسبتا بهینه با توجه به مدت زمانی که به الگوریتم اجازه اجرا داده باشیم، پیدا می‌شود [92].

[1] Relieff
[2] Gain Ratio
[3] Entropy
[4] Scatter

- **الگوریتم توده ذرات**

روش توده ذرات[1] از رفتار اجتماعی دسته‌های پرندگان الهام گرفته شده‌است. در این سیستم عامل‌ها به طور محلی با یکدیگر همکاری می‌نمایند و رفتار جمعیت باعث همگرایی در نقطه‌ای نزدیک به جواب بهینه سراسری می‌شود. برای نیل به این هدف لازم است تا در بعد اول همه اعضای گروه موقعیت خود را با تابعیت از بهترین فرد گروه تغییر دهند و در بعد دوم لازم است تا تک تک اعضا بهترین موقعیتی که تاکنون تجربه کرده‌اند را در حافظه نگهداری کرده، میل به سوی بهترین موقعیت خود در گذشته داشته باشند. سرعت و موقعیت ذرات تا زمان برآورده شدن شرط خاتمه (رسیدن به حد آستانه تعیین شده) با توجه به این دو مورد به طور مداوم به روز می‌شود [95].

- **الگوریتم کلونی مورچه**

برای پیاده‌سازی مساله انتخاب ویژگی با الگوریتم کلونی مورچه[2] دو رشته صفر و یک به طول تعداد متغیرها در نظر می‌گیریم و الگوریتم را برای تشکیل رشته‌ای از صفر و یک‌ها اجرا می‌کنیم. مسیری که هر مورچه طی می‌کند با استفاده از توابع فرومون و ابتکاری مشخص می‌شود. در نهایت بهترین مسیری که توسط مورچه‌ها انتخاب شده‌است مشخص می‌کند که کدام ویژگی می‌بایست انتخاب شود [96].

- **الگوریتم جستجوی خفاش**

الگوریتم جستجوی خفاش[3] الهام گرفته از نحوه مسیریابی خفاش‌ها است. این موجودات قدرت بینایی ضعیفی داشته و بر اساس انعکاس صدای خود مسیر حرکت را تعیین می‌کنند.در این روش جستجوی فضای ویژگی بر اساس معیارهای سرعت و مکان صورت گرفته و از تنظیم فرکانس به منظور جهش به حالت بعد استفاده می‌شود. میزان این جهش با توجه به تغییرات بلندی صدا و پالس

[1] Particle Swarm Intelligence (PSO)
[2] Ant Colony
[3] Bat Search

انتشار متفاوت است. این تغییرات قابلیت تمرکز خودکار را فراهم می‌کند به طوری که باعث استخراج راه‌حل‌های قوی‌تر می‌شود [97].

- **الگوریتم زنبورعسل**

این الگوریتم شبیه‌سازی رفتار جستجوی غذای گروه‌های زنبور عسل است. برای انتخاب ویژگی توسط این الگوریتم ابتدا زنبورها d گام به جلو می‌روند و در هر گام تصمیم می‌گیرند یک ویژگی را انتخاب کنند یا خیر. پس از اتمام d مسیر زنبورها به کندو بازگشته و نتایج خود را ارزیابی می‌کنند. در این مرحله زنبورها به دو دسته متعهد و غیرمتعهد تقسیم می‌شوند. زنبورهایی که برازندگی‌شان پس از ارزیابی از مقدار آستانه بیشتر بود به عنوان متعهد و بقیه غیرمتعهد هستند. این مفهوم به معنی وفاداری زنبور عسل به شهد کشف شده‌است. زنبورهای غیرمتعهد می‌بایست بر اساس چرخ رولت به سمت راه‌حل زنبور متعهد حرکت کنند و پس از حرکت مسیر با d گام آزادند که مسیر جدیدی را انتخاب کنند [98].

۳-۳-۲- ارزیابی ویژگی

یک تابع ارزیابی، میزان خوب بودن یک زیرمجموعه تولید شده را بررسی کرده و یک مقدار به عنوان میزان خوب بودن زیرمجموعه مورد نظر بازمی‌گرداند. این مقدار با بهترین زیرمجموعه قبلی مقایسه می‌شود. اگر زیرمجموعه جدید، بهتر از زیرمجموعه‌های قدیمی باشد، زیرمجموعه جدید به عنوان زیرمجموعه بهینه، جایگزین قبلی می‌شود. توابع ارزیابی که در این رساله بررسی شده‌است در ادامه توضیح داده شده‌است.

- **همبستگی**

ضریب همبستگی[1] ابزاری آماری برای تعیین نوع و درجه رابطه یک متغیر کمی با متغیر کمی دیگر است. ضریب همبستگی، یکی از معیارهای استفاده شده در تعیین همبستگی دو متغیر است. ضریب همبستگی شدت رابطه و همچنین نوع رابطه (مستقیم یا معکوس) را نشان می‌دهد. این ضریب بین ۱ تا ۱- است و در نبود وجود رابطه بین دو متغیر، برابر صفر است. همبستگی بین دو متغیر تصادفی X و Y به صورت رابطه (۷-۳) تعریف می‌شود که در آن E عملگر امید ریاضی، cov به معنای کوواریانس، corr نماد معمول برای همبستگی پیرسان و سیگما نماد انحراف معیار است [99].

$$corr(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E\left[(X-\mu_X)(Y-\mu_Y)\right]}{\sigma_X \sigma_Y} \quad (۷-۳)$$

- **مربع کای**

در این روش ارزش هر ویژگی با استفاده از محاسبه مقدار آماری مربع کای[2] و با توجه به کلاس مد نظر ارزیابی می‌گردد. این مقدار با استفاده از آزمون مربع کای که جزء آزمون‌های غیر پارامتری است محاسبه می‌شود. برای این کار ابتدا مربع کای را برای هر کلاس و با استفاده از فرمول (۸-۳) محاسبه می‌کنیم. در این فرمول Oi مقدار مشاهده ویژگی i و Ei مقدار مورد انتظار آن ویژگی است.

$$X^2 = \sum [(O_i - E_i)^2 / E_i] \quad (۸-۳)$$

سپس درجه آزادی دو متغیر را با استفاده از فرمول (۹-۳) به دست می‌آوریم. در این فرمول ثابت‌های r و c به ترتیب تعداد دسته‌های موجود برای دو متغیر مورد آزمایش هستند.

$$P = (r-1)(c-1) \quad (۹-۳)$$

حال با توجه به جدول توزیع احتمال مربع کای، اگر مقدار P بیشتر از ۰/۰۵ است، فرضیه مورد قبول و دو متغیر در ارتباط هستند. در غیر این صورت متغیرها هیچ وابستگی به هم ندارند [99].

[1] Correlation
[2] Chi Squared

- **پایداری**

معیار پایداری[1] قابلیت پیشگویی مقدار یک متغیر به‌وسیله یک متغیر دیگر را اندازه‌گیری می‌کند. ضریب (Coefficient) یکی از معیارهای وابستگی کلاسیک است و می‌توانیم آن را برای یافتن همبستگی بین یک ویژگی و یک کلاس به کار ببریم. اگر همبستگی ویژگی X با کلاس C بیشتر از همبستگی ویژگی Y با کلاس C باشد، در این صورت ویژگی x برای قرار گرفتن در کلاس C بر y اولویت دارد. با یک تغییر کوچک، می‌توانیم وابستگی یک ویژگی با ویژگی‌های دیگر را اندازه‌گیری کنیم. این مقدار درجه افزونگی این ویژگی را نشان می‌دهد. همه توابع ارزیابی بر پایه معیار وابستگی را می‌توانیم بین دو دسته معیارهای مبتنی بر فاصله و اطلاعات تقسیم کنیم؛ اما به خاطر اینکه این روش‌ها با دیدی متفاوت از دید اصلی، به مسأله نگاه می‌کنند، این کار را انجام نمی‌دهیم [99].

- **نرخ بهره**

در این روش ارزش یک ویژگی با توجه به میزان نرخ بهره آن با توجه به کلاس‌ها به دست می‌آید. نرخ بهره در واقع نسبت میزان بهره اطلاعاتی ویژگی به اطلاعات انشعاب[2] آن ویژگی است. ارزش اطلاعاتی انشعاب، اطلاعات بالقوه‌ای که در صورت تقسیم مجموعه داده D به v بخش ایجاد می‌شود را نشان می‌دهد که v مقادیر دیده شده ویژگی A است. لذا اطلاعات انشعاب ویژگی A با توجه به مجموعه داده D از فرمول (۱۰-۳) به دست می‌آید [99]

$$SplitInfo\ _A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times log_2\left(\frac{|D_j|}{D}\right) \qquad (۱۰-۳)$$

- **بهره اطلاعاتی**

در این روش ارزش یک ویژگی به‌وسیله بهره اطلاعاتی[3] آن به دست می‌آید. در بحث تئوری اطلاعات و یادگیری ماشین، بهره اطلاعاتی معادل واگرایی Kullback-Leibler است. فرض کنید که

[1] Consistency
[2] Spilt Information
[3] Information Gain

T یک مجموعه از داده‌های آزمایشی باشد که هر یک از این داده‌ها به شکل $(X,y) = (x_1, x_2, \ldots, x_k,$ y) است. به نحوی که x_a عضو vals(a) مقدار a امین ویژگی داده X است که به کلاس y تعلق دارد. در این صورت اگر تابع H میزان آنتروپی باشد، بهره اطلاعاتی از فرمول (۱۱-۳) محاسبه می‌شود [99].

$$IG(T,a) = H(T) - \sum_{v \in vals(a)} \frac{|\{X \in T | x_a = v\}|}{|T|} \cdot H(\{X \in T | x_a = v\}) \qquad (۱۱-۳)$$

- **طبقه‌بند OneR**

این تابع ارزیابی ویژگی‌های را بر اساس نتیجه طبقه‌بند OneR ارزیابی می‌کند. این طبقه‌بند ویژگی‌ها را بر اساس نرخ خطا (در مجموعه آموزشی) رتبه‌بندی می‌کند. به این صورت که بر اساس هر یک از ویژگی‌ها یک درخت تصمیم سطح یک ایجاد می‌کند. سپس داده‌ها را بر این اساس طبقه‌بندی کرده و نرخ خطای به دست آمده را اندازه‌گیری می‌کند [100].

- **Relief**

یک تابع ارزیابی مبتنی بر فاصله است که بر اساس تولید مکاشفه‌ای عمل می‌کند. این روش از یک راه‌حل آماری برای انتخاب ویژگی استفاده می‌کند. همچنین یک روش مبتنی بر وزن است که از الگوریتم‌های مبتنی بر نمونه الهام گرفته است. روش کار به این صورت است که از میان مجموعه نمونه‌های آموزشی، یک زیرمجموعه نمونه انتخاب می‌کند. کاربر بایستی تعداد نمونه‌ها در این زیرمجموعه را مشخص کرده باشد و آن را به عنوان ورودی به الگوریتم ارائه دهد. الگوریتم به صورت تصادفی یک نمونه از این زیرمجموعه را انتخاب می‌کند. سپس برای هر یک از ویژگی‌های این نمونه، نزدیک‌ترین برخورد[1] و نزدیکترین شکست[2] را بر اساس معیار اقلیدسی پیدا می‌کند. نزدیک‌ترین برخورد، نمونه‌ای است که کمترین فاصله اقلیدسی را در میان سایر نمونه‌های همکلاس با نمونه انتخاب شده دارد. نزدیک‌ترین شکست نیز نمونه‌ای با کمترین فاصله اقلیدسی در میان نمونه‌های غیر همکلاس با نمونه انتخاب شده‌است [99].

[1] Nearest Hit
[2] Nearest Miss

- **عدم قطعیت متقارن**

در این روش همبستگی بین دو ویژگی X و Y بر مبنای تئوری اطلاعات اندازه‌گیری می‌شود. در واقع هر چه میزان عدم قطعیت متقارن[1] یک ویژگی بیشتر باشد، ارزش آن ویژگی بیشتر است. برای به دست آوردن این مقدار از فرمول (۱۲-۳) استفاده می‌شود که در آن IG(X) میزان بهره اطلاعاتی ویژگی X و H(X) آنتروپی آن است [101].

$$SU(X,Y) = 2\frac{IG(X|Y)}{H(X)+H(Y)} \qquad (۱۲-۳)$$

۳-۳-۳- روش پیشنهادی برای انتخاب ویژگی

همان‌طور که اشاره شد، روش‌های انتخاب ویژگی شامل دو جزء اصلی استراتژی جستجو و تابع ارزیابی‌کننده هستند. در روش پیشنهادی استراتژی جستجو حذف رو به عقب و تابع ارزیابی، میزان شاخص دقت متعادل شده (IBA[2]) به دست آمده توسط طبقه‌بند Naïve Bayes است. شاخص IBA شاخصی است که به منظور ارزیابی داده‌های نامتوازن استفاده می‌شود و از آنجا که تعداد وب‌هرزها به کل صفحات وب بسیار اندک است، به منظور ارزیابی نتایج حاصل از این معیار استفاده کردیم. این معیار تلاش دارد تا توازنی بین میزان دقت کلی بایاس نشده و میزان غلبه کلاس‌ها بر یکدیگر ایجاد کند. منظور از محاسبه این مقدار این است که بدانیم به ازای هر حدس درست در تشخیص صفحات هرز، چه میزان صفحه غیرهرز را به غلط هرز تشخیص داده‌ایم [102]. توضیحات بیشتر در خصوص این شاخص در فصل آتی ارائه خواهد شد.

در روش انتخاب ویژگی پیشنهادی که Smart-BT نام دارد، طبقه‌بندی بخش مهمی از آن به شمار می‌رود و باید دو خصوصیت مهم داشته باشد. اولین خصوصیت زمان اجرا و پیچیدگی زمانی آن است که به علت تعدد استفاده به عنوان ارزیاب، می‌بایست زمان پایینی داشته باشد. دومین ویژگی به

[1] Symmetrical Uncertainty
[2] Index of Balanced Accuracy

خصیصه مهم این مجموعه داده مرتبط است و آن نامتوازن بودن داده‌ها است به طوری که کلاس کوچکتر از اهمیت بالایی برخوردار است. از سوی دیگر گوسی نبودن توزیع برخی از ویژگی ها، نیازمند انتخاب طبقه‌بندی غیرحساس به توزیع متغیرها است که همه این مسائل منجر به انتخاب Naïve Bayes شده‌است.

این طبقه‌بند براساس تئوری معروف بیز و با فرض (نه زیاد) ساده ی استقلال بین هر جفت از ویژگی‌ها طراحی شده‌است. اگرچه در عمل ویژگی‌ها به ندرت مستقل از هم هستند و این همان دلیلی است که به این روش Naive می‌گویند ولی در [103] نویسندگان نشان داده‌اند که این روش می‌تواند حتی در شرایطی که ویژگی‌ها مستقل نباشند می‌تواند بهینه باشد. این روش پیچیدگی زمانی خطی متناسب با اندازه مجموعه داده دارد و به فضای ذخیره‌سازی اندکی نیازمند است. به علاوه، این روش در بعضی موقعیت‌های واقعی مثل فیلترکردن هرزنامه‌ها و طبقه بندی متن کاملا خوب عمل میکند و چون ویژگی‌ها را مستقل در نظر می‌گیرد، توزیع احتمال هر ویژگی می‌تواند به عنوان توزیع یک بعدی به طور مستقل تخمین زده شود که به نوبه خود باعث کاهش مشکلات ناشی از ابعاد بالا می‌شود. این برای ویژگی‌های نامرتبط خیلی مهم است و زمانی که با تعداد زیادی ویژگی‌های با اهمیت یکسان روبرو هستیم از روشهایی مثل درخت تصمیم بهتر عمل می‌کند.

در روش پیشنهادی برای انتخاب ویژگی فرض بر این است که بیشتر ویژگی‌ها موثر هستند و تنها تعداد کمی از آن‌ها قابلیت تفکیک کنندگی پایینی دارند. بنابراین هدف هدف یافتن بزرگ‌ترین زیرمجموعه‌ای از ویژگی‌ها است که با حذف اعضای آن از مجموعه داده، بیشترین میزان بهبود در دقت طبقه‌بند پدیدار شود. بدین منظور، پیش از حذف هر یک از ویژگی‌ها کل مجموعه داده با استفاده از الگوریتم Naïve Bayes طبقه‌بندی شده و میزان IBA محاسبه می‌گردد. این عدد به عنوان IBA پایه در ادامه الگوریتم و به منظور بررسی میزان تاثیر ویژگی‌ها بر دقت طبقه‌بند استفاده می‌شود. سپس مجموعه‌ای از کلیه زیرمجموعه‌های i عضوی (که در ابتدا $i = 1$) ویژگی‌های موجود به نام

$Working\ Set$ ایجاد می‌شود. مجموعه $Irremovable$ که در ابتدا خالی است و به تدریج در طول اجرای الگوریتم به‌روزرسانی می‌شود، شامل زیرمجموعه‌ای از ویژگی‌ها است که در هر تکرار بعد از ایجاد $Working\ Set$ بررسی می‌شود. در این بررسی هر کدام از مجموعه‌های عضو $Working\ Set$ که حداقل یکی از اعضای $Irremovable$ زیرمجموعه‌ای از آن‌ها باشد، حذف می‌شوند.

مرحله بعد تشخیص این نکته است که حذف کدام یک از اعضای $Working\ Set$ باعث تغییر در میزان کارایی طبقه‌بند می‌شود. این کار با استفاده از تابع $Evaluate$ انجام می‌شود. این تابع ویژگی‌هایی را که به عنوان ورودی دریافت می‌کند را از مجموعه داده اولیه حذف کرده و پس از طبقه‌بندی مجموعه داده حاصل با استفاده از الگوریتم Naïve Bayes، میزان IBA به دست آمده را محاسبه می‌کند. اگر این مقدار بزرگتر یا مساوی با میزان IBA پایه (IBA حاصل از مشارکت کلیه ویژگی‌ها) باشد، به این معنا است که حذف این ویژگی‌ها باعث افزایش دقت طبقه‌بند شده‌است و نقش زیادی در تعیین دقت آن ندارد. لذا این ویژگی‌ها به مجموعه $LowInfoFeatures$ منتقل می‌شود. اما اگر این مقدار کمتر از IBA پایه باشد، نشان دهنده این است که ویژگی‌های حذف شده نقش مهمی در افزایش دقت طبقه‌بند داشته و نباید حذف شوند. این ویژگی‌ها به مجموعه $Irremovable$ که قبلا به آن اشاره شد اضافه می‌شوند. در نهایت نیز شمارنده (i) یک واحد افزایش پیدا می‌کند. این الگوریتم با حذف $Working\ Set$ فعلی و ایجاد دوباره آن براساس مقدار جدید شمارنده (i) ادامه پیدا می‌کند و زمانی خاتمه می‌یابد که حذف هیچ یک از اعضای این مجموعه باعث بهبود دقت طبقه‌بند نشود. شبه کد این الگوریتم را در شکل ۳-۴ مشاهده می‌کنید.

```
Algorithm finding unvalued features
input:  feature set of dataset
output: feature set Result such that removing it from dataset leads achieving BestIBA
Initialize the Irremovable and LowInfoFeatures set to Null, i to 1 and BestAnswer to 0
    BaseIBA = Evaluate (AllFeatures)
    /* calculate IBA of dataset using input features */
    CreateSubSet (WorkingSet, i)
    /* Create all i-member subset of features set and put it in WorkingSet */
    while ( WorkingSet(i) is not null) do
        for each ws ∈ WorkingSet do
            if (exist ir ∈ Ivvremovable that ir is a subset of ws) then
                Eliminate(ws)
                /* eliminates ws from WorkingSet */
            else
                e := Evaluate(ws)
                diff := BaseIBA - e
                if (e > 0)
                    Remove (ws, Ivvremovable)
                    /* removes ws to Ivvremovable set */
                else
                    Remove (ws, LowInfoFeatures)
                    if (e >= BaseIBA) then
                        BestIBA := e
                        Result := ws
                    end if
                end if
            end if
        end for
        i := i + 1
        CreateSubSet (WorkingSet, i)
    end while
```

شکل ۳-۴- شبه کد الگوریتم پیشنهادی Smart-BT [104]

۳-۳-۴- پیچیدگی زمانی و مکانی

به منظور محاسبه پیچیدگی زمانی روش ارائه شده، ابتدا پیچیدگی زمانی هر یک از توابع آن را محاسبه می‌کنیم:

- Evaluate()

این تابع مجموعه داده را با استفاده از ویژگی‌های ورودی و محاسبه شاخص دقت متعادل شده طبقه‌بندی می‌کند. بازسازی مجموعه داده به اندازه $O(n \times k)$ زمان احتیاج دارد که در آن n تعداد نمونه داده و k تعداد ویژگی‌های ورودی است. الگوریتم بیز ساده نیزنیازمند زمانی به اندازه $O(n \times k)$ است. بنابراین پیچیدگی زمانی این تابع $O(n \times k)$ است.

- CreateSubSet()

این تابع همه زیرمجموعه‌های k- عضوی مجموعه ویژگی را ایجاد می‌کند که برابر با انتخاب k عضو از d ویژگی است. بنابراین پیچیدگی زمانی این تابع برابر است با $O(C(k,d)) = O(\frac{d!}{k! \times (d-k)!})$ که در آن d ابعاد مجموعه داده و k تعداد عناصر موجود در زیرمجموعه است. از آنجایی که هر بار هنگامی که یک ترکیب جدید ایجاد می‌شود، می‌بایست آن را در متغیر خروجی کپی کنیم که زمانی معادل O (k) احتیاج دارد، پیچیدگی زمانی نهایی این تابع برابر با $O(C(k,d) \times k)$ است.

- Eliminate()

حذف مجموعه ورودی از WorkingSet، پیدا کردن یک مجموعه با k عضو در مجموعه‌ای از مجموعه‌های k عضوی، به اندازه $O(k \times n)$ زمان می‌برد که در آن n تعداد زیرمجموعه‌های k عضوی است. در نتیجه پیچیدگی زمانی این تابع $O(C(k,d) \times k)$ است.

- Remove()

این تابع مجموعه منبع را از WorkingSet حذف کرده و آن را در مجموعه نهایی کپی می‌کند. از آنجا که وظیفه اصلی این تابع پیدا کردن مجموعه منبع است، بنابراین پیچیدگی زمانی این تابع نیز همانند تابع Eliminate() معادل $O(C(k,d) \times k)$. است.

از آنجا که حلقه‌های "for" و "while" به منظور بررسی همه زیرمجموعه‌های ممکن از ویژگی‌ها پیاده‌سازی شده‌اند، 2^d بار تکرار می‌شوند. با توجه به بررسی شرایط در ابتدای حلقه "for"، به ازای هر زیرمجموعه کاندید، یکی از توابع Eliminate() یا Evaluate() به همراه تابع Remove() اجرا می‌شود. در بدترین حالت، پیچیدگی زمانی الگوریتم $O(2^{2d} \times n)$ است که در آن n تعداد نمونه‌های یک مجموعه داده را نشان می‌دهد و d ابعاد فضای ویژگی است. این حالت زمانی رخ می‌دهد که فقط یک ویژگی مفید باشد و بقیه حذف شوند. از آنجایی که ما یک پیش پردازش انجام می‌دهیم و مطمئن هستیم که ویژگی‌ها به اندازه‌ای خوب هستند که بیش از یک سوم آنها حذف نشود، به همین دلیل پیچیدگی زمانی الگوریتم $\Theta(2^d \times n)$ است. در بهترین حالت که همه

ویژگی‌ها مفید باشند و هیچ کدام از آنها قابلیت حذف نداشته باشند، پیچیدگی زمانی $\Omega(n \times d)$ است. از آنجا که مرتبه زمانی این الگوریتم نمایی و وابسته به ابعاد فضای ویژگی است، لذا کاهش تعداد ویژگی‌ها در مرحله پیش پردازش بسیار با اهمیت است.

۳-۴- طبقه بندی

در این رساله از تعدادی الگوریتم طبقه بندی جهت ارزیابی عملکرد استفاده شده‌است که به شرح ذیل می‌باشند:

- **C4.5**: در روش درخت تصمیم، نمونه‌ها به یک الگوریتم C4.5 که یکی از درختهای تصمیم‌گیری است اعمال می‌شوند. سپس با توجه به درخت هرس شده حاصل، ویژگی-هایی که در آن وجود دارد را بعنوان جواب مساله باز می‌گردانیم. الگوریتم C4.5 تکمیل شده‌الگوریتم ID3 می باشد.
- **Naive Bayes**: تئوری بیز یکی از روش‌های آماری برای رده بندی به شمار می‌آید. در این روش کلاس‌های مختلف، هر کدام به شکل یک فرضیه دارای احتمال در نظر گرفته می‌شوند. هر رکورد آموزشی جدید، احتمال درست بودن فرضیه‌های پیشین را افزایش و یا کاهش می‌دهد. در نهایت، فرضیاتی که دارای بالاترین احتمال باشند، به عنوان یک کلاس در نظر گرفته شده و برچسبی بر آن‌ها در نظر گرفته می‌شود..
- **SVM**: ماشین بردار پشتیبان یک گروه از الگوریتم‌های طبقه‌بندی نظارت‌شده هستند، که پیش‌بینی می‌کند یک نمونه در کدام کلاس یا گروه قرار می‌گیرد. این الگوریتم برای تفکیک دو کلاس از هم، از یک صفحه استفاده می‌کند به طوری که این صفحه از هر طرف بیشترین فاصله را تا هر دو کلاس داشته باشد. نزدیک‌ترین نمونه‌های آموزشی به این صفحه "بردارهای پشتیبان" نام دارند .

- **Adaboost**: این الگوریتم یکی از معروف‌ترین الگوریتم‌های برتر داده کاوی محسوب می‌شود. در واقع یک متا الگوریتم است که بمنظور ارتقاء عملکرد، و رفع مشکل رده‌های نامتوزان همراه دیگر الگوریتم‌های یادگیری استفاده می‌شود. در این الگوریتم طبقه‌بند در هر مرحله، به نفع نمونه‌های غلط طبقه‌بندی شده در مراحل قبل، تنظیم می‌گردد. این الگوریتم از کل مجموعه داده به منظور آموزش هر طبقه‌بند استفاده می‌کند، اما بعد از هر بار آموزش، بیشتر بر روی داده‌های سخت تمرکز می‌کند تا به درستی طبقه‌بندی شوند.
- **KNN**: یکی از بهترین، ساده‌ترین و متداول‌ترین طبقه‌بندی کننده‌ها بر پایه یادگیری نمونه، طبقه‌بند K نزدیکترین همسایه است. روش K نزدیک‌ترین همسایه یک گروه شامل K رکورد از مجموعه رکوردهای آموزشی که نزدیک‌ترین رکوردها به رکورد آزمایشی باشند را انتخاب کرده و بر اساس برتری رده یا برچسب مربوط به آن‌ها در مورد دسته رکورد آزمایشی مزبور تصمیم‌گیری می‌نماید. به عبارت ساده‌تر این روش رده‌ای را انتخاب می‌کند که در همسایگی انتخاب شده بیشترین تعداد رکورد منتسب به آن دسته باشند. بنابراین رده‌ای که از همه رده‌ها بیشتر در بین K نزدیک‌ترین همسایه مشاهده شود، به عنوان رده رکورد جدید در نظر گرفته می‌شود.
- **MLP**: یکی از مرسوم ترین انواع شبکه های عصبی، شبکه عصبی پرسپترون چند لایه(MLP) است که به طور موفقیت آمیزی در بازه وسیعی ازکاربردها از جمله طبقه بندی داده مورد استفاده قرار گرفته است .هنگام کار با شبکه‌های عصبی MLP با دو مسئله روبه رو هستیم : انتخاب معماری مناسب و انتخاب الگوریتم آموزشی مناسب. معماری مناسب به معنی انتخاب بهینه تعداد لایه ها، تعداد نرون‌ها در هر لایه و نوع تابع تحریک هر نرون می باشد و معماری بهینه شبکه های عصبی مبتنی بر مجموعه داده‌ها و ویژگی‌های آن‌ها است. الگوریتم‌های آموزشی متنوعی جهت آموزش شبکه های عصبی به کار می رود. متداول ترین الگوریتم آموزشی این شبکه ها، الگوریتم پس انتشار خطا می

باشد. در الگوریتم پس انتشارخطا در هر مرحله مقدار خروجی محاسبه شده جدید، با مقدار واقعی مقایسه شده و با توجه به خطای به دست آمده به اصلاح وزن های شبکه پرداخته می شود .به نحوی که در انتهای هر تکرار اندازه خطای حاصله کمتر از میزان به دست آمده در تکرار قبلی باشد.

- **Random Forest**: این طبقه‌بند یک روش بر مبنای مدل دسته جمعی است. این مدل، روش یادگیری برای طبقه بندی و رگرسیون است که با ساخت بسیاری از درخت‌های تصمیم گیری در زمان آموزش و انتخاب بهترین درخت از میان درختان تولید شده کار می کند.مدل دسته جمعی بر مبنای دقت و میزان اهمیت متغیرها کار می کند.
- **Classification & Regression Trees (CART)**: درخت تصمیم گیری CART یک روش تقسیم بندی بازگشتی باینری است. در این روش درختان به حداکثر رشد خود می‌رسند و سپس اصلاح می‌شوند که کار پیچیده ای است. هدف این مکانیزم تولید تنها یک درخت نیست بلکه تولید یک سری درختان اصلاح شده تو در تو است که همه آن درختان بهینه داوطلب هستند و انتخاب درخت بهینه تنها پس از ارزشیابی داده‌ها صورت می‌گیرد.
- **Logistic Regression**: رگرسیون لجستیک یکی از ابزارهای آماری است که به منظور مدل سازی و تحلیل داد ه‌ها از آن استفاده می شود. رگرسیون لجستیک دارای شکل کلی زیر است:

$$\log\left(\frac{\pi}{1-\pi}\right) = \alpha + \sum_{i=1}^{k} \beta_i . x_i \qquad (۱۲-۳)$$

در این مدل، π احتمال تعلق فرد به سطح اول متغیر وابسته است، xi متغیر مستقل i ام و βi ضریب برآورد شده مدل برای متغیر مستقل i ام است .از مزایای استفاده از مدل رگرسیون لجستیک

علاوه بر مدلسازی مشاهده ها، امکان پیش بینی احتمال تعلق هر فرد به هر یک از سطوح متغیر وابسته و هم چنین امکان محاسبه مستقیم نسبت شانس با استفاده از ضرایب مدل است.

- **Logit boost**: الگوریتم افزایندگی (Logit boost) به طریقی عمل می‌کند که همانند یک روش گروهی خود را برای انجام هدف اصلی تجهیز می‌کند و به لحاظ مفهوم شبیه به Bagging است. افزایندگی، آموزش دهنده ضعیف بعدی را بر اساس خطاهای آموزش دهنده قبلی آموزش می‌دهد. در شرایط عادی، افزایندگی نسبت به Bagging عملکرد بهتری دارد.
- **Logistic Model Tree**: یا به اختصار LMT یک مدل طبقه بندی تحت نظارت است که از ترکیب رگرسیون لجستیک و درخت تصمیم استفاده می‌کند. ایده این روش به این صورت است که به جای اینکه در برگهای درخت از مقادیر ثابت استفاده شود، با استفاده از الگوریتم LogitBoost برای هر یک از آنها یک مدل رگرسیون خطی ایجاد شود. این الگوریتم به جای استفاده از مقادیر تصادفی، در هر فراخوانی از نتایج گره والد خود برای شروع استفاده می‌کند.سپس این گره‌ها با استفاده از الگوریتم ۴/۵C تقسیم شده و در نهایت هرس می‌شوند. همچنین برای پیدا کردن تعداد تکرارهای مناسب برای اجرای LogitBoost از cross validation استفاده می‌شود.
- **Least Absolute Deviation Tree**: درختان رگرسیون LAD از به حداقل رساندن انحراف مطلق (Absolute Deviation) بین مقدار پیش بینی شده و مقدار واقعی، به عنوان معیار انتخاب استفاده می‌کنند. استفاده از این معیار باعث می‌شود که درخت نسبت به حضور داده‌های پرت مقاوم تر باشند.

در این رساله از این الگوریتم‌ها به منظور ارزیابی روش پیشنهادی استفاده شده‌است و برای تشخیص وب‌هرز از یک روش طبقه‌بندی مبتنی بر ایمنی مصنوعی استفاده شده که در مقایسه با روشهای اشاره شده، نتایج بهتری در تشخیص وب‌هرز دارد.

۳-۴-۱- سیستم ایمنی مصنوعی

الگوریتم سیستم ایمنی مصنوعی از سیستم ایمنی بدن انسان الهام گرفته شده‌است. سیستم ایمنی بدن انسان دو بخش ذاتی و اکتسابی دارد. سیستم ایمنی ذاتی، همان‌طور که از نامش برمی‌آید، با گذر زمان تغییر نمی‌کند و به همین دلیل در AIS[1] به طور گسترده مورد استفاده قرار نگرفته است.

ایمنی ذاتی هنگامی که فعال می‌شود، برای چند روزی فعال می‌ماند. درحالی‌که ایمنی اکتسابی پس از شروع فعالیت، هفته‌ها فعال می‌ماند. وظیفهٔ ایمنی اکتسابی است که با ضعیف شدن یا ناکارا بودن ایمنی ذاتی، پاتوژن‌ها را از بین می‌برد. ناکارا بودن ایمنی ذاتی به این علت است که ایمنی ذاتی نمی‌تواند پاسخی خاص برای یک پاتوژن مهاجم ایجاد کند. در این حالت سیستم ایمنی اکتسابی وارد عمل می‌شود. برخلاف ایمنی ذاتی، پاسخ ایمنی اکتسابی خاص است. ایمنی اکتسابی حافظه دارد و درنتیجه پاتوژنی را که یک بار حمله کرده و سیستم برای آن پاسخی تولید کرده‌است، به یاد می‌آورد و در برخوردهای بعدی، برای مقابله با آن پاتوژن پاسخ سریع‌تری تولید می‌کند. برای سیستم ایمنی مصنوعی یک چهارچوب ارائه شده‌است که هسته آن شامل سه عامل است:

۱. فرم نمایش اجزای سیستم.
۲. مجموعه‌ای از مکانیزم‌ها برای سنجش ارتباطات بین اجزاء با محیط و یکدیگر (یک تابع شباهت).
۳. شیوهٔ سازگاری.

شیوهٔ نگاشت یک مسئلهٔ مهندسی به AIS رویکردی لایه‌لایه است. بر اساس دامنهٔ مسئلهٔ مورد بحث، نوع نمایش و به تبع آن انتخاب تابع پیوند تعیین می‌شود. ما نیز از این چهارچوب برای نگاشت مسئلهٔ خود به AIS استفاده خواهیم کرد.

[1] Artificial Immune System

بسته به اینکه از کدام بخش سیستم ایمنی بدن برای روابط میان اجزاء الهام گرفته باشیم، الگوریتم‌های متنوعی تحت عنوان سیستم ایمنی مصنوعی ایجاد می‌شود که از مهم‌ترین آن‌ها می‌توان به انتخاب منفی، انتخاب کلونی و تئوری خطر اشاره کرد. پیشنهاد ما بر این است که از الگوریتم انتخاب منفی به‌منظور تشخیص صفحات وب‌هرز استفاده کنیم.

۳-۴-۱-۱- انتخاب منفی/ مثبت

در بدن انسان وظیفه سلول T تشخیص آنتی‌ژن‌های سلول‌های خودی از آنتی‌ژن‌های عوامل بیماری زا (غیر خودی) است. این سلول‌ها در مغز استخوان تولید و از طریق جریان خون وارد غده تیموس می‌شوند. در این غده میل ترکیبی سلول‌های T نابالغ با سلول‌های خودی موجود در تیموس بررسی می‌شود و هرکدام از سلول‌های T نابالغ میل ترکیبی زیادی با یکی از سلول‌ها داشته باشند حذف می‌شوند و سایر سلول‌های T وارد جریان خون می‌شوند. به این شکل از انتخاب، که داشتن تطابق باعث مرگ (حذف) سلول می‌شود، انتخاب منفی می‌گویند. در صورتی که عدم تطابق باعث مرگ شود، به آن انتخاب مثبت گفته می‌شود. در سال ۱۹۹۴ فورست برداشتی آزاد از انتخاب منفی بیولوژیک ارائه کرد که بر تولید ردیاب‌های[1] متغیر تمرکز دارد. در این الگوریتم فرض می‌شود که داده‌ها رشته‌هایی دودویی بوده و S مجموعه داده‌های خودی است.

۱. ایجاد الگوهای تصادفی P

۲. هر الگوی P که میل ترکیبی آن با حداقل یک الگوی S بیشتر از آستانه شناسایی است، حذف می‌شود.

۳. الگوهای باقیمانده به مجموعه A اضافه می‌شود.

[1] Detector

در نهایت از مجموعه A در مرحله نظارت یا اجراء استفاده می‌شود. در این مرحله هر الگوی ورودی با الگوهای جدیددر مجموعه A مقایسه می‌شود و در صورتی که میل ترکیبی آنها از یک حد آستانه‌ای بیشتر شود، الگوی ورودی به عنوان یک الگوی غیرخودی حذف می‌شود. می‌توان الگوریتم را به شکلی تغییر داد که الگوهای خودی را پوشش دهند. به این حالت انتخاب مثبت می‌گویند.

۳-۴-۱-۲- انتخاب جامعه

در سیستم ایمنی بدن، زمانی که یک سلول B آنتی‌ژنی را شناسایی می‌کند، شروع به تکثیر کرده، تعداد زیادی سلول B یکسان و مشابه تولید می‌کند. هر چه میل پیوندی بین سلول B و آنتی‌ژن بیشتر شود نرخ تکثیر بیشتر خواهد شد، در نتیجه سلول‌های B با میل پیوندی بالاتر، همانند بیشتری تولید می‌کنند که اصل انتخاب جامعه نام دارد. بخش‌های اصلی این الگوریتم شامل شناسایی آنتی‌ژن، تکثیر و جهش سلول‌های B (آنتی‌بادی‌ها) و ایجاد سلول‌های حافظه است. در این الگوریتم سلول‌هایی که میل ترکیبی بیشتری دارند، بیشتر تکثیر می‌شوند. از طرف دیگر سلول‌هایی که میل ترکیبی کمتری داشته‌اند بیشتر تغییر می‌یابند. مراحل این الگوریتم به شرح ذیل می‌باشد

۱- ایجاد N آنتی‌بادی اولیه

۲- برای هر آنتی‌ژن:

۱- N1 آنتی‌بادی که میل ترکیبی بیشتری با آنتی‌ژن دارند انتخاب می‌شوند.

۲- همه آنتی‌بادی‌های انتخاب شده تکثیر می‌شوند. هرچه میل ترکیبی آنتی‌بادی بیشتر باشد، بیشتر تکثیر می‌شود.

۳- با احتمال زیاد آنتی‌بادی‌های موجود جهش داده می‌شود. هرچه میل ترکیبی بیشتر باشد جهش کمتر می‌شود.

۴- N2 آنتی‌بادی با بیشترین میل ترکیبی انتخاب می‌شوند.

۵- تعدادی آنتی‌بادی تصادفی ایجاد شده، جایگزین آنتی بادهایی که میل ترکیبی کمتری دارند می‌شوند.

۳- در صورت برقرار نبودن شرط خاتمه، مرحله ۲ تکرار شود.

۳-۴-۲- روش پیشنهادی جهت طبقه بندی صفحات وب بر اساس سیستم ایمنی مصنوعی

در روشی که در این رساله به منظور تشخیص صفحات وب‌هرز ارائه می‌شود ابتدا با استفاده از الگوریتم انتخاب مثبت و جامعه، به ایجاد سلول‌های حافظه به منظور تشخیص سلول‌های خودی می‌پردازیم. سپس داده ورودی در صورت تشخیص توسط سلول‌ها، غیرهرز و در غیر این صورت هرز در نظر گرفته می‌شود. در واقع مسئله تشخیص صفحات وب‌هرز از صفحات غیرهرز را می‌توان به مسئله شناخت خودی و غیرخودی در سیستم ایمنی مصنوعی نگاشت کرد (جدول ۳-۲). در این صورت صفحات غیرهرز سلول‌های خودی هستند که می‌بایست از سلول‌های غیرخودی (عوامل بیماری‌زا) متمایز شوند. از آنجا که تعداد صفحات غیرهرز بسیار بیشتر از صفحات وب‌هرز است، به دنبال تشکیل سلول‌های حافظه‌ای هستیم که بتواند سلول‌های خودی را تشخیص دهد. به همین دلیل برای تشکیل سلول‌های حافظه از الگوریتم انتخاب جامعه و برای انتخاب بهترین سلول از الگوریتم انتخاب مثبت استفاده شده‌است؛

جدول ۳-۲ نگاشت AIS به مسئله تشخیص صفحات وب‌هرز

اجزاء سیستم ایمنی مصنوعی	معادل آن در مساله
آنتی‌ژن	بردار ویژگی داده
آنتی‌بادی	ساختاری با سه جزء شامل ۱- بردار ویژگی ۲- شعاع شناسایی ۳- پرچم
شعاع شناسایی	بیشترین فاصله مجاز یک آنتی‌بادی تا نزدیکترین عضو غیرخودی
کلون	آنتی‌بادی جهش‌یافته
شناسایی سلولی	حالتی که در آن فاصله بردارهای ویژگی آنتی‌بادی و آنتی‌ژن کمتر از شعاع آنتی‌بادی باشد

الگوریتم پیشنهادی برای انجام این کار به این صورت است:

۱. مجموعه داده آموزشی را به دو دسته خودی و غیرخودی تقسیم می‌کنیم (داده‌هایی که برچسب هرز دارند غیرخودی و داده‌هایی که برچسب غیرهرز دارند خودی در نظر گرفته می‌شوند).

۲. فرض می‌کنیم S مجموعه اعضای خودی و s نشانگر یک عضو از اعضای این مجموعه باشد. در مقابل N مجموعه اعضا غیرخودی و n نشانگر یک عضو آن باشد. برای کلیه sهای عضو S یک پرچم شناسایی با مقدار اولیه false در نظر می‌گیریم. برای کلیه sها این مراحل انجام می‌شود:

a. پرچم s بررسی می‌شود. در صورت false بودن این مراحل صورت می‌گیرد:

i. آنتی‌بادی معادل آنتی‌ژن s ایجاد می‌شود.

ii. با توجه به مجموعه N، شعاع شناخت آنتی‌بادی را تعیین می‌شود.

iii. به تعداد نسل‌ها (که از قبل تعیین شده‌است) این مراحل صورت می‌گیرد:

۱. به تعداد کلون‌ها (که از قبل تعیین شده‌است) این مراحل صورت می‌گیرد:

a. تا زمانی که کلون بتواند s را شناسایی کند:

i. آنتی‌بادی جهش‌یافته، به عنوان کلون نگهداری می‌شود.

ii. شعاع کلون محاسبه می‌شود.

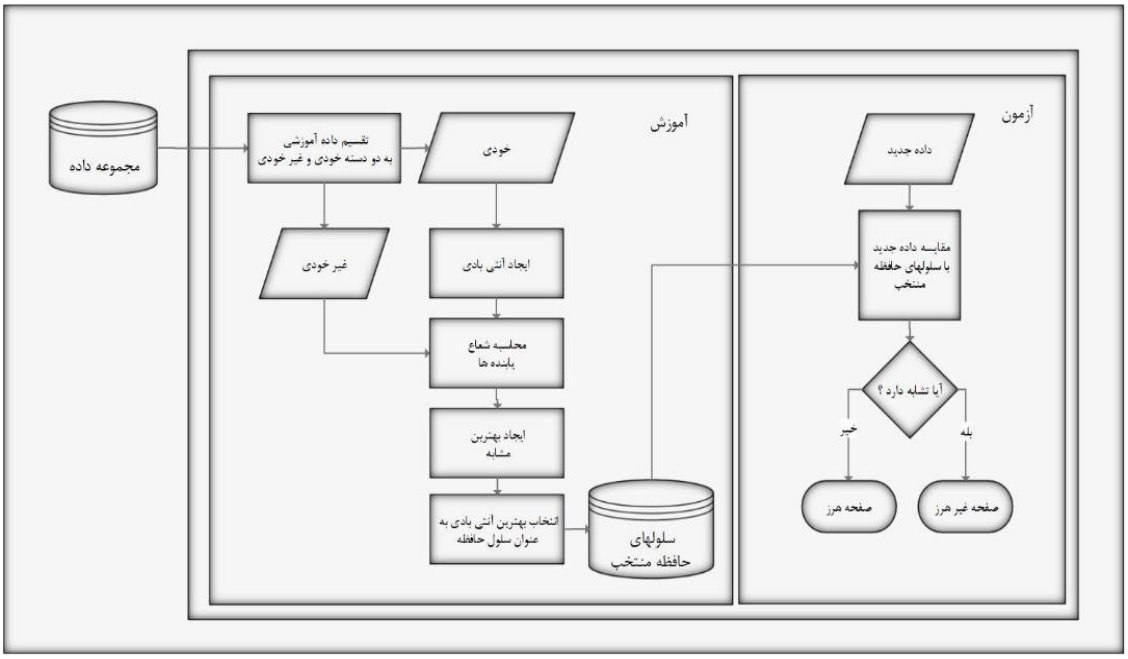
b. کلون با آنتی‌بادی اولیه مقایسه شده، در صورت عملکرد بهتر (شناسایی تعداد خودی‌های بیشتر یا شعاع بزرگتر) به عنوان بهترین کلون حفظ می‌شود.

۲. بهترین کلون را به عنوان آنتی‌بادی در نظر می‌گیریم

iv. بهترین آنتی‌بادی را به عنوان سلول حافظه در نظر می‌گیریم.

b. کلیه sهای عضو S توسط سلول جدید بررسی شده، در صورت شناسایی وضعیت پرچمشان به true تغییر می‌کند.

۳. مجموعه آنتی بادی‌های تولید شده ذخیره و در مرحله تست به منظور تشخیص صفحات وب‌هـرز مورد استفاده قرار می‌گیرند.

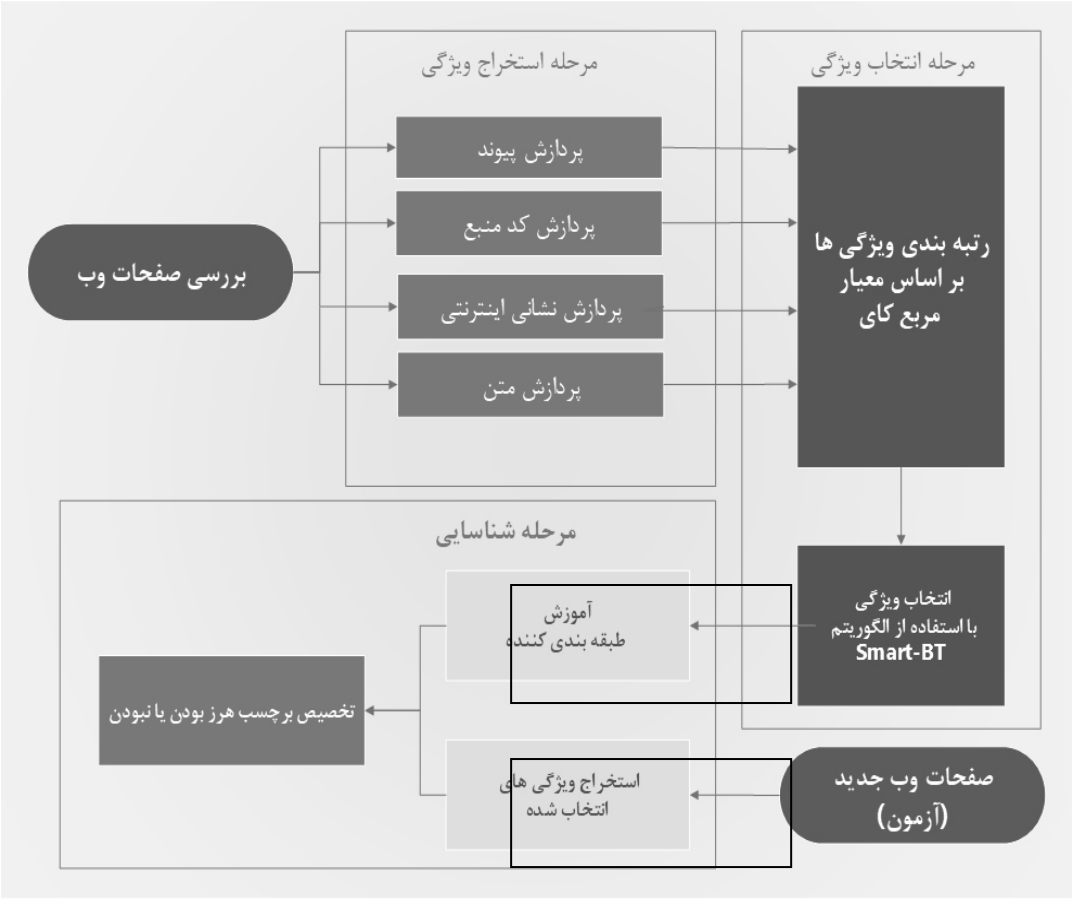


شکل ۳-۵- فلوچارت روش طبقه‌بندی پیشنهادی

در اینجا مرحله یادگیری به پایان می‌رسد. حاصل کار این بخش ایجاد سلول‌های حافظه‌ای است که می‌توانند سلول‌های خودی را به خوبی شناسایی کنند. حال برای تشخیص هرز بودن یک صفحه وب جدید، ابتدا فاصله بردار ویژگی داده جدید با اولین سلول حافظه محاسبه می‌شود. اگر این فاصله کمتر از شعاع سلول باشد، صفحه به عنوان غیرهرز در نظر گرفته می‌شود. در غیر این صورت فاصله آن با سلول بعدی محاسبه می‌شود. در صورتی که هیچ یک از سلول‌های حافظه موفق به شناسایی داده ورودی نشوند، آن صفحه به عنوان وب‌هرز تشخیص داده می‌شود.

۳-۵- مدل پیشنهادی

همان طور که در بخش‌های قبلی اشاره شد، مدل پیشنهادی در این رساله جهت تشخیص وب‌هرز شامل استخراج ویژگی، انتخاب ویژگی و طبقه بندی است. در مرحله اول، ما یک گروه از ویژگیهای کم هزینه بر مبنای ساختار آدرس اینترنتی و برچسب‌های HTML معرفی کردیم. سپس با استفاده از روش انتخاب ویژگی مبتنی بر حذف رو به عقب (Smart-BT) تعداد ویژگی‌ها را کاهش دادیم. نمودار این الگوریتم در شکل ۳-۶ نمایش داده شده‌است. در این مدل، ورودی صفحه وب است که شامل ۴ بخش است. گراف رابطه، کد منبع، آدرس اینترنتی و متن قابل مشاهده. هر بخش منبع خوبی از اطلاعات است و به منظور استخراج ویژگی‌های مناسب مورد بررسی قرار می‌گیرد. بعد از استخراج ویژگی، آنها بر حسب امتیازی که توسط معیار ارزیابی کای به دست می‌آورند، رتبه بندی می‌شوند. در مرحله بعدی، یک زیرمجموعه از n-بهترین ویژگی به عنوان ورودی به الگوریتم Smart-BT سپرده می‌شود. خروجی الگوریتم، مجموعه ویژگی نهایی است که برای طبقه‌بندی صفحات وب جدید استفاده می‌شود. که در این مدل از سیستم ایمنی مصنوعی برای طبقه‌بندی صفحات وب استفاده شده‌است. در بخش بعدی ویژگی هایی که انتخاب شدند و نتایج حاصل از آن بررسی می‌شود.



شکل ۳-۶ نمودار مدل پیشنهادی

۳-۵-۱- پیچیدگی زمانی

همان طور که گفته شد، این مدل از دو بخش استخراج و انتخاب ویژگی و طبقه بندی تشکیل شده‌است. از آنجا که انتخاب ویژگی تنها یک بار استفاده می‌شود، پیچیدگی زمانی آن تاثیری در زمان اجرا ندارد. در بخش استخراج ویژگی نیز سه دسته ویژگی ارائه شده که پیچیدگی زمانی هر یک را محاسبه می‌کنیم:

- ویژگی‌های مبتنی بر آدرس اینترنتی: از آنجا که تعداد کاراکترهای یک URL تعداد محدودی است و به عنوان مثال طول دامنه نمی‌تواند بیش از ۶۳ کاراکتر باشد، لذا ویژگی‌هایی که از این منبع استخراج می‌شود از مرتبه $O(1)$ است.

- ویژگی‌های مبتنی بر کد HTML: اگر منبع هر صفحه وب شامل n کاراکتر باشد، آنگاه پیچیدگی زمانی ویژگی‌های استخراج شده‌از مرتبه O (n) است.
- ویژگی‌های مبتنی بر خوانایی متن: اگر مجموعه داده شامل N صفحه باشد و هر صفحه حاوی n کاراکتر، از آنجا که مدل LDA تنها یک بار ایجاد می‌شود و نیازی به تکرار آن نیست، پیچیدگی زمانی ویژگی‌های مستخرج از آن O (Nn) است.

در بخش تشخیص یا طبقه‌بندی نیز از آنجا که یابنده‌ها یکبار در ابتدا ایجاد می‌شوند، تنها مرتبه زمانی فرآیند تشخیص مورد نیاز است که اگر N ویژگی و d یابنده داشته باشیم، مرتبه زمانی $O\ (Nd)$ است. لذا مرتبه زمانی نهایی برابر است با $O\ (Nd + Nn)$ است که چون N و d ثابت هستند، می‌توان آن را به صورت $O\ (n)$ خلاصه سازی کرد.

۴

نتایج

معرفی مجموعه داده

نتایج به دست آمده

۴-۱- مقدمه

همان‌طور که گفته شد، ازآنجاکه موتور جستجو یکی از تأثیرگذارترین عناصر دنیای اینترنت به شمار می‌رود، وجود یک سامانه قوی شناسایی تشخیص صفحات وب‌هرز از دیگر صفحات باعث افزایش اعتبار و اقبال به موتور جستجو می‌شود. دستیابی به دقت مناسب و پایین آوردن نرخ خطا و نیز تصمیم‌گیری در کمترین زمان ممکن هدف نهایی در این مسئله است. از آنجا که ایجادکنندگان صفحات هرز همواره در حال تغییر روش‌های مورد استفاده برای فریب موتورهای جستجو هستند، روشی که بتواند پایداری بیشتری نسبت به این تغییرات از خود نشان دهد مطلوب تر است. به همین جهت در فصل گذشته مدلی برای انجام این کار ارائه کردیم.

در این رساله ابتدا یک سری ویژگی جدید معرفی و به ویژگی‌های رایج در این حوزه اضافه شد. سپس با استفاده از معیار مربع کای و رتبه بندی، تعدادی از آنها انتخاب شدند. این مجموعه به عنوان داده ورودی در الگوریتم ابداعی Smart-BT مورد استفاده قرار گرفت و تعداد ویژگی‌های نهایی کمتر شد. در نهایت این ویژگی‌ها از فایل‌های ورودی جدید استخراج و با استفاده از سیستم ایمنی مصنوعی، وب‌هرز بودن یا نبودن صفحه جدید مشخص می‌شود. نتایج حاصل از این مدل پیشنهادی در ادامه فصل بررسی خواهد شد.

۴-۲- مجموعه داده

مجموعه داده استاندارد برای شناسایی وب‌هرز در زبان انگلیسی، WEBSPAM-UK2007 است [105]. این مجموعه داده شامل ۱۰۵،۸۹۶،۵۵۵ صفحه از ۱۱۴،۵۲۹ میزبان در دامنه uk. است که به‌وسیلهٔ گروهی از افراد داوطلب برچسب‌گذاری شده‌است. این برچسب‌ها عبارتند از غیرهرز[1]، مرزی[2] و هرز که به ترتیب مقادیر صفر، ۰/۵ و یک به آن‌ها تعلق گرفته است. داده‌ها در

[1] Nonspam
[2] Borderline

ماه می[1] سال ۲۰۰۷ به‌وسیله آزمایشگاه الگوریتم‌های وب دانشگاه میلان بارگزاری شده‌است. این مجموعه داده شامل سه بخش است:

- برچسب‌های هرز/ غیرهرز
- آدرس‌های اینترنتی و پیوندها
- محتوای صفحات در قالب HTML

همچنین مجموعه‌ای از ویژگی‌های از پیش محاسبه‌شده برای این مجموعه داده وجود دارد که شامل ۹۶ ویژگی محتوامحور، ۴۴ ویژگی پیوندمحور و ۱۳۸ ویژگی مبتنی بر تبدیلات عددی ویژگی‌های پیوندمحور[2] است.

۹۶ ویژگی مبتنی بر محتوا در واقع ۲۴ ویژگی هستند که به چهار طریق محاسبه می‌شوند. یعنی هر یک از این ویژگی‌ها برای صفحه اصلی یک وبسایت و صفحه‌ای که دارای بالاترین امتیاز براساس الگوریتم PageRank است محاسبه می‌شود. همچنین میانگین و واریانس ویژگی‌ها در همه صفحات وب‌سایت نیز به عنوان ویژگی در نظر گرفته می‌شود. در نتیجه از ۲۴ ویژگی اولیه، ۹۶ ویژگی ایجاد می‌شود. برخی از این ویژگی‌ها عبارتند از: تعداد کلمات در صفحه، تعداد کلمات در عنوان، میانگین طول کلمات، سهم متون لنگر از کل متن، سهم متون قابل مشاهده به کل متن، نرخ فشرده سازی، مشابهت سه‌گانه مستقل[3] و انتروپی سه‌گانه ها.

مجموعه داده دیگری که در این زمینه وجود دارد Webb Spam Corpus 2011 است. این مجموعه داده توسط آقای وانگ و به عنوان بخشی از یک پروژه تحقیقاتی در دانشگاه جورجیا جمع‌آوری شده و حاوی بیش از ۳۵۰۰۰۰ صفحه وب‌هرز و فاقد هرگونه صفحه غیرهرز است [106].

[1] May
[2] Transformed Link-based Features
[3] Independent trigram likelihood

در مورد زبان فارسی تاکنون مجموعه داده مناسبی برای شناسایی وب‌هرزهای فارسی تهیه و استاندارد نشده‌است. برای تهیه چنین مجموعه داده‌ای به همکاری پروژه‌های موتور جستجو نیاز است.

۴-۳- فرض‌های مسئله

در این رساله فرض بر این است که صفحات متنی و در قالب HTML و فاقد تصویر و ویدئو باشند. مجموعه داده مسئله در حوزه زبان انگلیسی و با استفاده از مجموعه داده‌های WEBSPAM-UK2007 و Webb Spam Corpus 2011 خواهد بود. تشخیص در حالت برون خط[1] صورت می‌گیرد و محدودیتی در مورد زمان اجرا وجود ندارد.

۴-۴- معیارهای ارزیابی

یکی از ساده ترین و پرکاربردترین معیارهای ارزیابی مدل‌های طبقه‌بند استفاده از نمونه‌هایی است که به اشتباه طبقه‌بندی شده‌اند. بدین منظور می‌توان برای مشخص کردن میزان خطای مدل طبقه‌بندی از جدول درهم‌ریختگی[2] استفاده کرد (شکل ۴-۱).

شکل ۴-۱ جدول درهم‌ریختگی

		برچسب پیش‌بینی شده	
		مثبت	منفی
برچسب شناخته شده	مثبت	TP	FN
	منفی	FP	TN

حال بر اساس این مقادیر می‌توان معیارهای مختلف ارزیابی طبقه‌بند و اندازه‌گیری دقت را تعریف کرد. پارامتر دقت، متداول‌ترین، اساسی‌ترین و ساده‌ترین معیار اندازه‌گیری کیفیت یک طبقه‌بند است و عبارت است از میزان تشخیص صحیح طبقه‌بند در مجموع دو دسته. این پارامتر در واقع نشان‌گر

[1] Offline
[2] Confusion Matrix

میزان الگوهایی است که درست تشخیص داده شده‌اند و بر اساس ماتریس شکل ۴-۱، به صورت رابطه (۴-۱) تعریف می‌شود:

$$Accuracy = (TP+TN) / (TP+FN+FP+TN) \quad (۴-۱)$$

این معیار تنها یک دورنما از عملکرد طبقه‌بند را نشان می‌دهد و برای داده‌های متعادل مناسب است. در این مسئله که مجموعه داده‌ای به شدت نامتعادل دارد، دقت معیار مناسبی نیست زیرا حتی اگر کلیه داده‌ها غیرهرز تشخیص داده شوند، دقتی برابر با ۹۴٪ به دست می‌آید. به همین دلیل از معیارهای دیگری مثل صحت و یادآوری استفاده می‌شود.

$$Precision = TP / (TP+FN) \quad (۴-۲)$$

$$Recall = TN / (TN+FP) \quad (۴-۳)$$

معیار صحت بازگوکننده این نکته است که طبقه‌بند، به چه اندازه در تشخیص تمام نمونه‌های مثبت موفق بوده‌است. همانگونه که از رابطه فوق مشخص است، تعداد نمونه‌های منفی که توسط طبقه‌بند به اشتباه مثبت تشخیص داده شده‌اند، اثربخشی در محاسبه این پارامتر ندارد. در مقابل ممکن است در مواقعی صحت تشخیص کلاس منفی حائز اهمیت باشد که در این صورت از یادآوری استفاده می‌شود. واضح است که پیش‌بینی عالی، پیش‌بینی است که مقادیر صحت و یادآوری مربوط به آن، هر دو صد درصد باشند؛ اما احتمال وقوع این اتفاق در واقعیت بسیار کم است و همیشه یک حداقل خطایی وجود دارد. این دو معیار بنا بر ماهیتی که دارند همواره در رقابت با یکدیگر هستند. یعنی افزایش یکی با کاهش دیگری همراه است و برعکس. همین وضعیت منجر به معرفی معیار دیگری به نام F-measure یا F-Score برای ارزیابی طبقه‌بندها شده‌است. این معیار در واقع میانگین هارمونیک دو معیار صحت و یادآوری است.

$$F\text{-measure}= 2 * (Recall * Precision) / (Recall + Precision) \quad (۴-۴)$$

علاوه بر معیارهای فوق، از آنجا که مجموعه داده مورد استفاده توزیع نامتوازنی دارد (۹۴٪ غیرهرز و ۶٪ هرز) به منظور ارزیابی هر چه بهتر ویژگی‌های از معیار دیگری به نام IBA استفاده کردیم. همان‌طور که پیشتر توضیح داده شد، این معیار تلاش دارد تا توازنی بین میزان دقت کلی بایاس نشده و میزان غلبه کلاس‌ها بر یکدیگر ایجاد کند. بدین منظور از مفهومی به نام معیار غلبه استفاده می‌شود. این معیار به رابطه بین کلاس‌ها از لحاظ میزان غلبه اشاره دارد و عددی در بازه [-۱,+۱] است.

$$Dom = TP_{rate} - TN_{rate} \quad (۵-۴)$$

شاخص دقت متوازن (IBA) با استفاده از مفهوم معیار نفوذ[1] معرفی شده و برای محیط‌های نامتوازن مفید است. فاکتور وزن نتایجی را مدنظر قرار می‌دهد که دارای نرخ طبقه‌بندی نسبتاً بهتری بر روی کلاس اقلیت است. فرمول IBA به صورت رابطه (۶-۴) است:

$$IBA_{\alpha}(M) = (1+\alpha . Dom)M \quad (۶-۴)$$

(1+α.Dom) فاکتوری به منظور وزن‌دار کردن معیار کارایی (M) است که از رابطه (۷-۴) محاسبه می‌شود:

$$M = TP \times (1 - FP) \quad (۷-۴)$$

در این رابطه، M، سطح زیر یک نمودار دو بعدی است که یک محور آن مربع میانگین هندسی دقت کلاس‌ها و محور دیگر آن، اختلاف علامت‌دار بین دقت کلاس‌ها است. ضریب α نیز به منظور وزن‌دهی به معیار غلبه استفاده می‌شود. در [107] نشان داده شده که ۰/۱ مقدار مناسبی برای این ضریب است.

[1] Dominance

۴-۵- نتایج به دست آمده

۴-۵-۱- مقایسه نحوه تشخیص صفحات وب‌هرز توسط طبقه‌بندهای رایج

در این بخش از رساله هدف بررسی عملکرد طبقه‌بندهای معروف در تشخیص صفحات وب‌هرز است. در این بررسی نوع ویژگی‌ها نیز در نظر گرفته شده و تلاش بر این بوده که اثربخشی نوع ویژگی‌های مورد استفاده در میزان صحت عملکرد طبقه‌بند مورد توجه قرار گیرد. بدین منظور از نرم افزار weka [108] و دادگان WEBSPAM-UK2007 استفاده شده‌است. الگوریتم‌های طبقه بندی که مورد بررسی قرار گرفته‌اند شامل درخت تصمیم، ماشین بردار پشتیبان، AdaBoost، نزدیکترین همسایگی، بیز ساده، درختهای طبقه بندی و رگرسیون، رگرسیون لجستیک، Logit Boost، مدل درختی، شبکه عصبی پرسپترون چندلایه، جنگل تصادفی و درخت LAD است. جدول ۴-۱ نشان دهنده طبقه بندی داده‌ها به دو دسته هرز و غیر هرز با توجه به ویژگی‌های محتوا محور توسط این طبقه‌بندها است. همان طور که مشاهده می‌شود، بهترین نرخ یادآوری مربوط به بیز ساده است. بهترین صحت را شبکه عصبی پرسپترون و بهترین معیار F را جنگل تصادفی ایجاد می‌کند. طبقه‌بندی که بیشترین تعادل را بین صحت و یادآوری ایجاد کرده نزدیکترین همسایگی است. از آنجا که مهم است نمونه‌های کلاس کوچکتر شناسایی شوند، نتایج به دست آمده توسط شبکه عصبی چندان جالب توجه نیست. مضاف بر اینکه اکثر طبقه‌بندها، کل نمونه‌ها را به کلاس بزرگتر نسبت داده‌اند و اگرچه دقت بالایی دارند، برای تشخیص وب‌هرز مناسب نیستند.

جدول ۴-۱- نرخ تشخیص وب اسپم با ویژگی‌های مبتنی بر محتوا.

روش	Accuracy	Precision	Recall	F-Measure	IBA
Naïve Bayes	۱۴/۳۲	۰/۰۶	**۰/۹۰**	۰/۱۱	۰/۰۰۱
SVM	۹۴/۲۰	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰۰
Logistic Regression	۹۳/۴۳	۰/۳۲	۰/۱۱	۰/۱۷	۰/۱۰۰
MLP	**۹۴/۶۶**	**۰/۶۶**	۰/۱۷	۰/۲۷	۰/۱۶۳
KNN	۹۲/۹۷	۰/۳۶	۰/۲۷	۰/۳۱	**۰/۲۴۴**
AdaBoost	۹۴/۲۰	۰/۰۰	۰/۰۰	**۰/۰۰**	۰
Logit Boost	۹۴/۲۰	۰/۵۰	۰/۱۰	۰/۱۶	۰/۰۹۱
۴/۵C	۹۲/۲۵	۰/۲۹	۰/۲۳	۰/۲۶	۰/۱۹۵
Model Tree	۹۴/۶۱	۰/۶۴	۰/۱۶	۰/۲۶	۰/۱۵۴
Random Forest	۹۴/۶۱	۰/۵۸	۰/۲۵	**۰/۳۵**	۰/۲۱۶

در مرحله بعد، ویژگی‌های مبتنی بر پیوند را جایگزین ویژگی‌های مبتنی بر متن کرده و با استفادها از همان الگوریتم‌ها آن‌ها را طبقه‌بندی می‌کنیم. هدف از این کار بررسی میزان اثربخشی ویژگی‌ها در طبقه‌بندی است که نتایج به دست آمده در جدول ۴-۲ ثبت شده‌است. این بار الگوریتم نزدیکترین همسایگی بهترین نرخ یادآوری و بیشترین تعادل میان صحت و یادآوری را داشته است. همچنین جنگل تصادفی بیشترین نرخ صحت و معیار F را به دست آورده است. هرچند باید توجه کرد نتایج به دست آمده ضعیفتر از نتایج حاصل از استفاده از ویژگی‌های مبتنی بر متن می‌باشد. به عنوان مثال بهترین نرخ یادآوری در حالت قبل ۰/۹۰۳ بوده که این بار به ۰/۱۰۷ کاهش یافته. همین طور معیار F از ۰/۳۴۸ به ۰/۱۱۷ تقلیل پیدا کرده که نشان می‌دهد ویژگی‌های مبتنی بر متن دقت تفکیک کنندگی بیشتری دارند.

جدول ۴-۲ نرخ تشخیص وب اسپم با ویژگی‌های مبتنی بر پیوند.

روش	Accuracy	Precision	Recall	F-Measure	IBA
Naïve Bayes	۹۲/۴۶	۰/۱۳	۰/۰۵	۰/۰۷	۰/۰۳
SVM	۹۴/۰۶	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
Logistic Regression	**۹۳/۸۲**	۰/۲۲	۰/۰۲	۰/۰۳	۰/۰۱
MLP	۹۲/۴۱	۰/۰۸	۰/۰۳	۰/۰۴	۰/۰۰۶
KNN	۸۹/۹۳	۰/۱۲	**۰/۱۱**	۰/۱۱	**۰/۰۵۶**
AdaBoost	۹۴/۰۶	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
Logit Boost	۹۴/۰۶	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
۴/۵C	۹۳/۷۲	۰/۲۳	۰/۰۳	۰/۰۴	۰/۰۲
LAD Tree	۹۳/۴۸	۰/۲۷	۰/۰۶	۰/۱۰	۰/۰۳
Model Tree	۹۴/۰۶	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
Random Forest	۹۳/۳۸	**۰/۲۸**	۰/۰۷	**۰/۱۲**	۰/۰۴۵

در مرحله سوم، از ویژگی‌های مبتنی بر پیوند که برخی توابع ریاضیاتی بر آنها اعمال شده، استفاده کرده‌ایم (جدول ۴-۳). بیشترین صحت به وسیله جنگل تصادفی و بیشترین یادآوری باز هم توسط بیز ساده به دست آمده است. همچنین این الگوریتم بیشترین توازن میان صحت و یادآوری را داشته است. این میزان توازن که برابر با ۰/۱۹۲ است، از حالت قبلی بهتر و از حالت اولیه کمتر است. این امر نشانگر آن است که ویژگی‌های مبتنی بر پیوند در حالت اصلی خود تفکیک کنندگی پایینی دارند و باید تحت تبدیلات ریاضیاتی تغییر کنند تا میزان تفکیک کنندگی شان بالا رود.

جدول ۴-۳ نرخ تشخیص وب اسپم با ویژگی‌های مبتنی بر پیوند تبدیل یافته.

روش	Accuracy	Precision	Recall	F-Measure	IBA
Naïve Bayes	۸۲/۴۸	۰/۱۳	**۰/۳۴**	**۰/۱۹**	**۰۱۹**
Svm	۹۴/۰۶	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
logistic regression	۹۳/۳۳	۰/۱۱	۰/۰۲	۰/۰۳	۰/۰۱
MLP	۹۱/۹۷	۰/۰۸	۰/۰۳	۰/۰۵	۰/۰۱
KNN	۹۰/۸۵	۰/۱۷	۰/۱۴	۰۱۵	۰/۱۰
Adaboost	۹۴/۰۶	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
logit boost	۹۳/۹۷	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
۴/۵C	۹۲/۷۵	۰/۱۵	۰/۰۵	۰/۰۸	۰/۰۳
LAD Tree	۹۳/۶۳	۰/۲۶	۰/۰۴	۰/۰۷	۰/۰۸
model tree	۹۴/۰۶	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
random forest	۹۳/۵۳	**۰/۲۸**	۰/۰۶	۰/۱۰	۰/۰۸

با توجه به اینکه ویژگی‌های مبتنی بر متن بیشترین نرخ تشخیص را دارند، بررسی می‌کنیم که آیا افزودن دو دسته ویژگی دیگر به آن موجبات بهبود آن فراهم خواهد آمد یا خیر. جدول ۴-۴ نتایج حاصل از افزودن ویژگی‌های مبتنی بر پیوند به ویژگی‌های مبتنی بر متن را نشان می‌دهد. همان طور که مشاهده می‌شود، نرخ یادآوری از ۰/۹۰۳ به ۰/۸۵۸ کاهش و صحت از ۰/۶۵۵ به ۰/۸۸۹ و IBA از ۰/۲۴۰ به ۰/۲۸۰ افزایش پیدا کرده‌است. معیار F نیز بدون تغییر باقی مانده است. شواهد نشان می‌دهد که افزودن ویژگی‌های مبتنی بر پیوند، اندکی باعث بهبود نتایج شده‌است.

جدول ۴-۴ نرخ تشخیص وب اسپم با ترکیب ویژگی‌های مبتنی بر محتوا و مبتنی بر پیوند.

روش	Accuracy	Precision	Recall	F-Measure	IBA
Naïve Bayes	۲۲/۶۴	۰/۰۶	**۰/۸۶**	۰/۱۱	۰/۰۵
Svm	۹۴/۲۰	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
logistic regression	۹۳/۳۳	۰/۳۰	۰/۱۲	۰/۱۷	۰/۱۰
MLP	۹۴/۵۱	۰/۵۹	۰/۱۷	۰/۲۶	۰/۱۶
KNN	۹۳/۰۷	۰/۳۸	۰/۳۱	۰/۳۴	**۰/۲۸**
Adaboost	۹۴/۵۶	**۰/۸۹**	۰/۰۷	۰/۱۳	۰/۰۷
logit boost	۹۴/۵۶	۰/۶۸	۰/۱۲	۰/۲۰	۰/۱۱
۴/۵C	۹۲/۳۰	۰/۲۸	۰/۲۱	۰/۲۴	۰/۱۸
LAD Tree	۹۳/۸۴	۰/۳۹	۰/۱۱	۰/۱۷	۰/۱۱
model tree	۹۴/۵۱	۰/۶۱	۰/۱۵	۰/۲۴	۰/۱۴
random forest	۹۴/۹۷	۰/۷۰	۰/۲۳	**۰/۳۵**	۰/۲۱

به همین دلیل این بار ویژگی‌های مبتنی بر پیوند تبدیل یافته را به ویژگی‌های متنی اضافه می‌کنیم. جدول ۴-۵ نشان می‌دهد که این بار نیز نتایج رضایت‌بخش نیست و بازهم شاهد کاهش نرخ یادآوری، معیار F و IBA هستیم. هرچند صحت افزایش داشته است. با توجه به اینکه هدف ما تشخیص کلاس کوچکتر است، این افزایش مطلوب نیست.

جدول ۴-۵ نرخ تشخیص وب اسپم با ترکیب ویژگی‌های مبتنی بر محتوا و مبتنی بر پیوند تبدیل یافته.

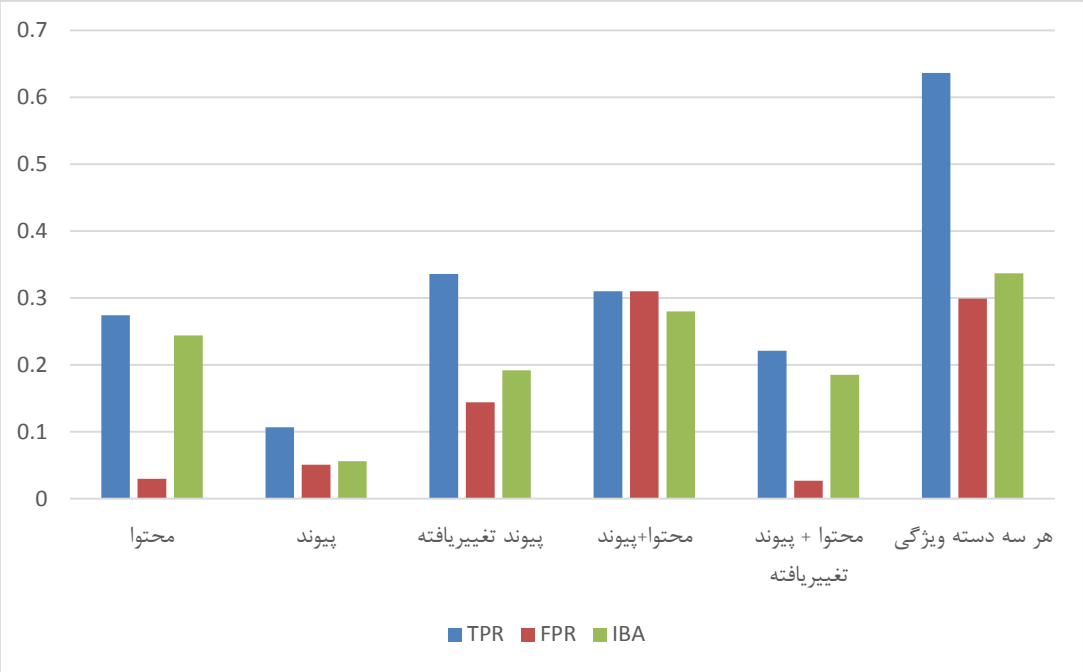
روش	Accuracy	Precision	Recall	F-Measure	IBA
Naïve Bayes	۴۳/۸۹	۰/۰۸	**۰/۷۶**	۰۱۴	۰/۱۸
SVM	۹۴/۲۰	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
logistic regression	۹۲/۹۲	۰/۳۶	۰/۲۲	**۰/۲۷**	**۰/۱۹**
MLP	۹۳/۴۳	۰/۳۳	۰/۱۳	۰/۱۹	۰/۱۲
KNN	۹۱/۶۸	۰/۲۰	۰/۱۴	۰/۱۷	۰/۱۱
Adaboost	۹۴/۵۶	۰/۸۹	۰/۰۷	۰/۱۳	۰/۰۷
logit boost	۹۴/۶۱	۰/۷۵	۰/۱۱	۰/۱۹	۰/۱۰
۴/۵C	۹۱/۶۳	۰/۲۵	۰/۲۱	۰/۲۳	۰/۱۷
LAD Tree	۹۳/۹۴	۰/۴۲	۰/۱۱	۰/۱۸	۰/۰۹
model tree	۹۴/۴۶	۰/۵۹	۰/۱۴	۰/۲۳	۰/۱۴
random forest	۹۴/۷۱	۰/۶۹	۰/۱۶	۰/۲۶	۰/۱۲

اگرچه شاید به نظر برسد باید ویژگی‌های مبتنی بر پیوند را کنار گذاشت و تنها به ویژگی‌های متنی اکتفا کرد اما به عنوان آخرین آزمایش از همه ویژگی‌ها در کنار هم استفاده می‌کنیم تا از صحت این فرضیه مطمئن شویم. همان طور که در جدول ۴-۶ مشخص شده، طبقه‌بند بیز ساده با استفاده از هر سه دسته ویژگی توانست بهترین توازن میان صحت و یادآوری را ایجاد کند. اگرچه به ظاهر دقت این طبقه‌بند پایین است، اما این طبقه‌بند توانسته بیشترین تعداد صفحات وب‌هرز (۶۴٪ صفحات) را به قیمت کمترین میزان هرز در نظر گرفتن صفحات غیرهرز (۳۰٪) شناسایی کند.

جدول ۴-۶ نرخ تشخیص وب اسپم با ترکیب هر سه دسته ویژگی.

روش	Accuracy	Precision	Recall	F-Measure	IBA
Naïve Bayes	۶۹/۶۸	۰/۱۶	**۰/۶۴**	۰۲۰	**۰/۳۴**
SVM	۹۴/۲۱	۰/۰۰	۰/۰۰	۰/۰۰	۰/۰۰
Logistic Regression	۹۲/۲۰	۰/۲۷	۰/۲۰	۰/۲۳	۰/۱۵
MLP	۹۴/۲۱	۰/۵۰	۰/۰۴	۰/۰۷	۰/۰۴
KNN	۹۱/۷۲	۰/۱۹	۰/۱۳	۰/۱۶	۰/۱۰
Adaboost	۹۴/۵۳	۰/۸۸	۰/۰۷	۰/۱۲	۰/۰۶
Logit Boost	۹۴/۵۹	۰/۷۳	۰/۱۰	۰/۱۸	۰/۱۰
C4.5	۹۱/۶۱	۰/۲۵	۰/۲۲	**۰/۲۴**	۰/۱۸
LAD Tree	۹۳/۹۴	۰/۱۴	۰/۱۱	۰/۱۸	۰/۰۷
Model Tree	۹۴/۳۷	۰/۵۶	۰/۱۴	۰/۲۲	۰/۱۳
Random Forest	۹۴/۸۶	۰/۸۳	۰/۱۴	**۰/۲۴**	۰/۱۴

اگر بخواهیم نگاهی کلی به سیر تغییرات میزان شاخص دقت متعادل شده بر اثر نوع ویژگی‌های استفاده شده جهت طبقه‌بندی داشته باشیم، شکل ۴-۲ به خوبی بیانگر این موضوع است.



شکل ۴-۲ روند تغییرات میزان IBA بر اثر تغییر نوع ویژگی‌های استفاده شده جهت طبقه‌بندی

۴-۵-۲- بررسی تاثیر تعداد ویژگی‌ها بر عملکرد طبقه‌بند

همان طور که در بخش قبل مشاهده شد، هر سه دسته ویژگی دارای خصایصی هستند که باعث بهبود عملکرد طبقه‌بند می‌شوند. اما تعداد این ویژگی‌ها زیاد است (۲۷۵=۱۳۸+۴۱+۹۶) و استخراج آن‌ها زمانبر خواهد بود. به همین علت در این بخش میزان اثربخشی کاهش ویژگی را بر عملکرد طبقه‌بند بررسی می‌کنیم. بدین منظور روش‌های انتخاب ویژگی رایجی که در فصل ۲-۳ توضیح داده شد را با استفاده از نرم‌افزار weka بر روی مجموعه داده WEBSPAM-UK2007 اعمال کردیم. سپس تشخیص صفحات وب‌هرز با استفاده از ویژگی‌های انتخاب شده توسط الگوریتم بیز ساده صورت گرفت. نتایج در جدول ۴-۷ نشان داده شده‌است.

جدول ۴-۷ میزان IBA حاصل از روش‌های مختلف انتخاب ویژگی

روش ارزیابی ویژگی	روش جستجو	تعداد ویژگی‌های انتخاب شده	IBA
Correlation	Forward Selection	۴۴	۰/۳۳۷
	Best first	۶۶	۰/۳۶۲
	LForward Selection	۱۲	۰/۲۳۲
	Greedy Stepwise	۶۶	۰/۳۶۲
	Genetic Search	۶۳	۰/۲۷۳
Consistency	Forward Selection	۱۷	۰/۱۰۹
	Best first	۸	۰/۲۴۵
	LForward Selection	۲۱	۰/۱۸۳
	Greedy Stepwise	۵	۰/۱۶۲
	Genetic Search	۱۵	۰/۲۰۸
Chi Square	**Ranker**	**۳۲**	**۰/۳۶۹**
InfoGain		۲۹	۰/۳۶۳
Gain Ratio		۳۹	۰/۳۵
SymUncertainty		۳۰	۰/۳۲۵
RelifF		۵۱	۰/۲۰۴
OneR		۱۹	۰/۰۲۷
Naïve Bayes Classifier	Genetic Search	۳۰	۰/۲۱
Naïve Bayes Classifier	PSO	۱۹	۰/۳۰۷
Naïve Bayes Classifier	ABC	۱۱۰	۰/۳۰۷
Naïve Bayes Classifier	ACO	۱۸۴	۰/۳۰۷

همان طور که مشاهده می‌شود، کاهش ویژگی با استفاده از رتبه بندی ویژگی‌ها براساس معیار مربع کای به علاوه بر کاهش یک هشتمی تعداد ویژگی‌ها (از ۲۷۵ به ۳۲) باعث بهبود عملکرد طبقه‌بند شده و معیار IBA را از ۰/۳۳۷ به ۰/۳۶۹ افزایش داده است. حال که الگوریتم بیز ساده و رتبه بندی براساس معیار مربع کای بهترین روش‌های رایج جهت طبقه‌بندی و کاهش ویژگی هستند، بار دیگر عملکرد آنها را در برابر انواع ویژگی و این بار در مقایسه با روش انتخاب ویژگی Smart-BT بررسی می‌کنیم.

جدول ۴-۸ مقایسه نتایج حاصل از اعمال الگوریتم Smart-BT بر روی انواع ویژگی‌ها و اثربخشی آن بر نرخ تشخیص وب‌هرز

مجموعه ویژگی	روش انتخاب ویژگی	تعداد ویژگی انتخاب شده	Accuracy	Recall	IBA
محتوا	-	۹۶	٪۱۴	۰/۹۰	۰/۰۰۲
	Chi squared + ranker	۷۶	٪۱۴	۰/۹۱	۰/۰۰۶
	Smart-BT	۵۷	٪۱۵	**۰/۹۲**	۰/۰۲۰
پیوند	-	۴۱	**٪۹۲**	۰/۰۵	۰/۰۲۹
	Chi squared + ranker	۲۲	٪۹۲	۰/۰۴	۰/۰۱۴
	Smart-BT	۱۰	٪۹۲	۰/۰۴	۰/۰۱۷
پیوند تبدیل یافته	-	۱۳۸	٪۸۲	۰/۳۴	۰/۱۹۲
	Chi squared + ranker	۴۷	٪۸۲	۰/۴۳	۰/۲۴۸
	Smart-BT	۴۰	٪۸۳	۰/۴۲	۰/۲۴۳
کل ویژگی‌ها	-	۲۷۵	٪۷۰	۰/۶۴	۰/۳۳۷
	Chi squared + ranker	۳۲	٪۷۲	۰/۶۳	۰/۳۶۹
	Smart-BT	۲۷	٪۷۵	۰/۶۴	**۰/۴۱۱**

همان‌طور که مشاهد می‌شود استفاده از روش پیشنهادی با استفاده از ویژگی‌های کمتر باعث حصول نتایج مشابه و حتی بهتر می‌شود. اهمیت این امر در کاهش زمان محاسبات و افزایش سرعت اجرای برنامه است. نکته دیگری که وجود دارد این است که همچنان ترکیبی از ویژگی‌های محتوا و پیوند محور نتیجه بهتری در پی دارد و در ازای شناسایی یک دسته از داده ها، دسته دیگر را حذف نمی‌کند. نکته بعدی در مورد ویژگی‌های انتخاب شده‌است که اگرچه پس از انتخاب ویژگی، تعداد آنها کاهش یافته است، اما نه تنها باعث کاهش عملکرد طبقه‌بند شده، بلکه آن را افزایش داده است. این ویژگی‌ها در جدول ۴-۹ فهرست شده‌اند.

جدول ۴-۹- لیست ویژگی‌های انتخاب شده

مخفف ویژگی	توضیح ویژگی
HST_16 HST_19 HST_20	درصدی از کلمات عضو Q که در hp میزبان قابل مشاهده هستند (,100 = Q 200).
HMG_43	درصدی از کلمات عضو Q که در mp میزبان قابل مشاهده هستند (100 = Q).
HMG_33	درصدی از کلمات صفحه mp میزبان که عضو K هستند (500 = K).
AVG_53	درصد میزان متن قابل رؤیت (مقدار میانگین برای همه صفحات یک میزبان)
AVG_64, AVG_65, AVG_66	میانگین نسبت تعداد کلمات موجود از مجموعه Q در کلیه صفحات یک میزبان (q = 200, 500, 1000)
AVG_67, AVG_68	میانگین درصدی از کلمات عضو Q که در کلیه صفحات میزبان قابل مشاهده هستند (q = 100, 200).
STD_73	انحراف معیار تعداد کلمات در کلیه صفحات میزبان
STD_75	انحراف معیار طول کلمه در کلیه صفحات میزبان
STD_76	انحراف معیار نسبت Anchor Text به کل متن صفحه برای کلیه صفحات میزبان
STD_77	نسبت متن قابل رویت در کلیه صفحات
STD_95	انتروپی کلیه صفحات میزبان
STD_96	انحراف معیار میزان Likelihood مستقل 3-gramهای کلیه صفحات میزبان
truncatedpagerank_1_hp	TruncatedPageRank با فاصله‌ی یک صفحه‌ی خانه میزبان
truncatedpagerank_2_mp, truncatedpagerank_3_mp, truncatedpagerank_4_mp	TruncatedPageRank صفحه mp میزبان با فاصله‌ی دو، سه و چهار
L_truncatedpagerank_1_mp, L_truncatedpagerank_2_mp, L_truncatedpagerank_3_mp, L_truncatedpagerank_4_mp	لگاریتم TruncatedPageRank صفحه mp میزبان با فاصله‌ی یک، دو، سه و چهار.
outdegree_hp, outdegree_mp	تعداد پیوندهای خروجی از hp و mp میزبان
L_outdegree_hp, L_outdegree_mp	لگاریتم تعداد پیوندهای خروجی از hp و mp میزبان
PageRank_mp , L_PageRank_mp	Pagerank صفحه mp میزبان و لگاریتم آن
K: مجموعه‌ای k عضوی از پرتکرارترین عبارات موجود در مجموعه داده (به‌استثنای کلمات توقف) Q: مجموعه‌ای q عضوی از پرتکرارترین عبارات جستجو شده در شبکه داخلی دانشگاه تهیه‌کنندگان مجموعه داده. Mp: صفحه‌ای که بالاترین میزان page rank را در بین همه صفحات یک میزبان دارد Hp: صفحه‌ی اصلی (خانه) میزبان	

بررسی خواستگاه این ویژگی‌ها نیز جالب توجه است (جدول ۴-۱۰). از آنجا که ویژگی‌های تغییرشکل یافته، همان ویژگی‌های مبتنی بر پیوند هستند که اعمال ریاضی از قبیل لگاریتم بر آنها

اعمال شده‌است، بنابراین اگر ویژگی‌ها مشابه را در نظر نگیریم این جدول نمایانگر آن است که متن یک صفحه وب دارای ویژگی‌های موثرتری به منظور تشخیص صفحات هرز و غیرهرز است. از جمله این ویژگی‌ها می‌توان به میزان استفاده از پرس‌وجوهای پرتکرار در متن، میزان استفاده از کلمات پرتکرار در متن، میزان متن قابل مشاهده در صفحه، تعداد کلمات، طول کلمات و میزان استفاده از Anchor Text اشاره کرد.

جدول ۴-۱۰ توزیع انواع ویژگی‌های انتخاب شده در حالت‌های مختلف

مجموعه ویژگی	محتوا محور	پیوند محور	پیوند تغییرشکل یافته
اولیه (۲۷۵)	۹۶	۴۱	۱۳۸
پیش پردازش(۳۲)	۱۸	۷	۷
الگوریتم ارائه شده (۲۸)	**۱۴**	۷	۷

همان‌طور که دیده می‌شود در مجموعه داده اولیه بیشتر ویژگی‌ها مبتنی بر پیوند هستند. بعد از استفاده از روش انتخاب ویژگی رتبه‌بندی، میزان ویژگی‌های مبتنی بر محتوا افزایش می‌یابد. در نهایت توزیع این دو نوع ویژگی در روش پیشنهادی به تعادل می‌رسد.

در نهایت نتایج به‌دست آمده با مقاله سیلوا و همکاران [50] و پاتیل [38] در جدول ۴-۱۱ مقایسه شده‌است. همان‌طور که مشاهده می‌شود مقدار IBA در هر سه روش کم و بیش یکسان است اما نکته‌ای که آنها را از هم متمایز می‌کند تعداد ویژگی‌های مورد استفاده جهت دستیابی به این مقدار است که بسیار کمتر است.

جدول ۴-۱۱ مقایسه نتایج به‌دست آمده

	تعداد ویژگی	طبقه‌بندی کننده	Recall	F-Measure	IBA
نتیجه اولیه	۲۷	Naïve Bayes	**۰/۶۴**	۰/۲۲۶	۰/۴۱
مقاله سیلوا [50]	۱۳۷	MDLClassifier	-	۰/۲۲۵	۰/۴۰
مقاله پاتیل [38]	۲۹۶	SVM	-	**۰/۴۴**	**۰/۴۳**
مقاله فدز [48]	۲۷۵	SVM+REGX	۰/۴۴	۰/۴۱	۰/۴۱

۴-۵-۳- بررسی نحوه تاثیر ویژگی‌های معرفی شده بر نرخ تشخیص وب‌هرز

همان طور که در فصل گذشته توضیح داده شد، سه دسته ویژگی شامل ویژگی‌های مستخرج از ساختار آدرس اینترنتی، کد HTML و ویژگی‌های ریشه گرفته از محتوای صفحه معرفی شدند. برای پیاده‌سازی این ویژگی‌ها از پایتون نسخه ۳ استفاده شد. برای انتخاب ویژگی و طبقه‌بندی نیز از کتابخانه‌های نرم‌افزار weka استفاده شد. برای طبقه‌بندی از الگوریتم Naïve Bayes و برای تست از اعتبارسنجی متقابل ۱۰ لایه[1] استفاده کردیم.

برای هر نمونه داده، ۱۳ ویژگی مبتنی بر URL (تعداد نقاط مسیر، تعداد نویسه‌های خاص مسیر، تعداد ارقام مسیر، طول مسیر، طول بزرگترین و کوچکترین توکن مسیر و میانگین طول آنها، بیشترین، کمترین و میانگین نرخ توالی حروف در توکن‌های مسیر، کمترین، بیشترین و میانگین شباهت توکن‌های مسیر به وبسایت‌های معروف) در سه حالت بیشترین میزان، کمترین میزان و میانگین بین کل صفحات یک دامنه خاص استخراج شد (۳۹ ویژگی). همچنین ۹ ویژگی شامل تعداد نقطه‌های موجود در دامنه، تعداد نویسه‌های خاص موجود در دامنه، تعداد اعداد دامنه، طول دامنه، طول بزرگترین توکن دامنه، کمترین نرخ توالی حروف در توکن‌های دامنه، بیشترین میزان شباهت دامنه به دامنه‌های معروف، استفاده از عبارت "www" یا "HTTPS" یا IP در ساختار آدرس اینترنتی نیز به ۳۹ ویژگی قبلی افزوده و تعداد آن را به ۴۹ افزایش می‌دهیم.

در مورد ویژگی‌های مبتنی بر کد HTML نیز تعداد ۲۴ برچسب شامل برچسب‌های comment، a، div، embed، iframe، img، link، extern، script، object، map، layer، video، menu، form، figure، details، command، code، button، audio، article، canvas، applet در سه حالت بیشترین میزان، کمترین میزان و میانگین بین کل صفحات یک دامنه خاص استخراج شد (۷۲ ویژگی).

[1] 10-fold cross validation

در مورد ویژگی‌های مبتنی بر پیوستگی، تعداد کلمات، جملات و پاراگرافهای یک صفحه برای کل صفحات یک دامنه در سه حالت کمترین، بیشترین و میانگین محاسبه شده‌است (۹ ویژگی). همچنین ۱۹ ویژگی جدول ۳-۱ نیز برای صفحه اصلی و برای کل صفحات دامنه به صورت میانگین محاسبه شده‌است (۳۸ ویژگی). در بخش ویژگی‌های مبتنی بر همبستگی متن نیز پنج ویژگی در مورد تعداد موضوعات مورد بحث در صفحه معرفی شد. در نتیجه در این بخش در مجموع ۵۲ ویژگی استخراج شده‌است.

اگر بدون کاهش ویژگی و فقط با استفاده از ویژگی‌های اولیه اقدام به تشخیص صفحات وب‌هرز نماییم، نتیجه به شکل جدول ۴-۱۲ خواهد بود. همان‌طور که مشاهده می‌شود، افزودن ویژگی‌های معرفی شده به ویژگی‌های قبلی باعث افزایش دقت، نرخ یادآوری و همینطور IBA شده‌است. اکنون با توجه به اثربخشی انتخاب ویژگی که در بخش قبلی مشاهده شد، اقدام به کاهش تعداد ویژگی‌ها می‌کنیم. بدین منظور ابتدا با استفاده از رتبه‌بندی بر اساس مربع کای تعداد ویژگی‌ها را کاهش داده، سپس الگوریتم Smart-BT را استفاده می‌کنیم. بهترین نتیجه با کاهش تعداد ویژگی‌ها از ۴۴۸ به ۴۸ به دست می‌آید که مقدار IBA را به ۰/۶۹۴ افزایش می‌دهد.

جدول ۴-۱۲ مقایسه نتایج حاصل از طبقه‌بندی داده با استفاده از ویژگی‌های مختلف

مجموعه ویژگی	تعداد ویژگی‌ها	Accuracy	Recall	IBA
محتوا محور	۹۶	٪۱۴	۰/۹۰	۰/۰۱
پیوند محور	۴۱	٪۹۲	۰/۰۵	۰/۰۳
پیوند محور تغییریافته	۱۳۸	٪۸۲	۰/۳۴	۰/۱۹
هر سه دسته فوق	۲۷۵	٪۷۰	۰/۶۴	۰/۳۴
مبتنی بر ساختار نشانی اینترنتی	۴۸	٪۸۰	۰/۳۳	۰/۱۷
مبتنی بر برچسب‌های کد منبع	۷۲	٪۳۶	۰/۴۲	۰/۲۲
مبتنی بر انسجام متن	۵۲	٪۸۲	۰/۲۱	۰/۳۱
همه ویژگی ها	۴۴۸	٪۷۶	۰/۶۵	۰/۳۶
ویژگی‌های انتخاب شده توسط Smart-BT	۴۸	٪۹۶	۰/۷۶	۰/۶۹

حال که ویژگی‌های مورد نظر را استخراج کردیم، به جای Naïve Bayes که تاکنون بهترین طبقه‌بند رایج بود، از یک روش طبقه‌بندی مبتنی بر سیستم ایمنی مصنوعی که در فصل گذشته معرفی شد استفاده می‌کنیم. این روش با زبان برنامه‌نویسی Visual C++ و در محیط Microsoft Visual Studio 2010 بر روی سیستم ۶۴ بیتی با پردازنده Intel Core2Dual با فرکانس GHz۲/۶۷ پیاده‌سازی شد که نتایج به دست آمده در جدول ۴-۱۳ نشان داده شده‌است.

جدول ۴-۱۳- نتایج به دست آمده توسط سیستم ایمنی مصنوعی در مقایسه با سایر روش‌های طبقه‌بندی

روش طبقه‌بندی	Accuracy	Precision	Recall	IBA
NaiveBayes	٪۹۶	۰/۷۹	۰/۷۵	۰/۶۹
NN	٪۹۵	۰/۸۴	۰/۳۵	۰/۳۵
J48	٪۹۵	۰/۹۲	۰/۵۴	۰/۵۲
Random Tree	٪۹۰	۰/۴۵	۰/۶۲	۰/۵۴
Random Forest	٪۹۶	**۰/۹۲**	۰/۴۷	۰/۴۷
SVM	٪۹۵	۰/۸۹	۰/۳۱	۰/۳۰
KNN	٪۹۳	۰/۶۳	۰/۵۳	۰/۴۸
Immunos	٪۹۴	۰/۳۴	۰/۷۲	۰/۶۱
روش ارائه شده	**٪۹۸**	۰/۸۳	**۰/۹۰**	**۰/۸۷**

همان‌طور که مشاهده می‌شود روش پیشنهادی در مقایسه با دیگر روش‌ها از جمله Immunos که یکی از روش‌های طبقه‌بندی رایج ایمنی مصنوعی است، از نرخ یادآوری بالاتری برخوردار است که در این مسئله خاص از اهمیت بیشتری برخوردار است. همین‌طور این جدول به خوبی نشان می‌دهد که IBA معیار مناسبی جهت سنجش کارایی الگوریتم‌ها در فضای داده‌های نامتوازن است. در ادامه روش پیشنهادی را با چند مقاله دیگر مقایسه کرده و نتایج را در جدول ۴-۱۴ نشان داده‌ایم.

جدول ۴-۱۴ مقایسه نتایج به دست آمده با دیگران

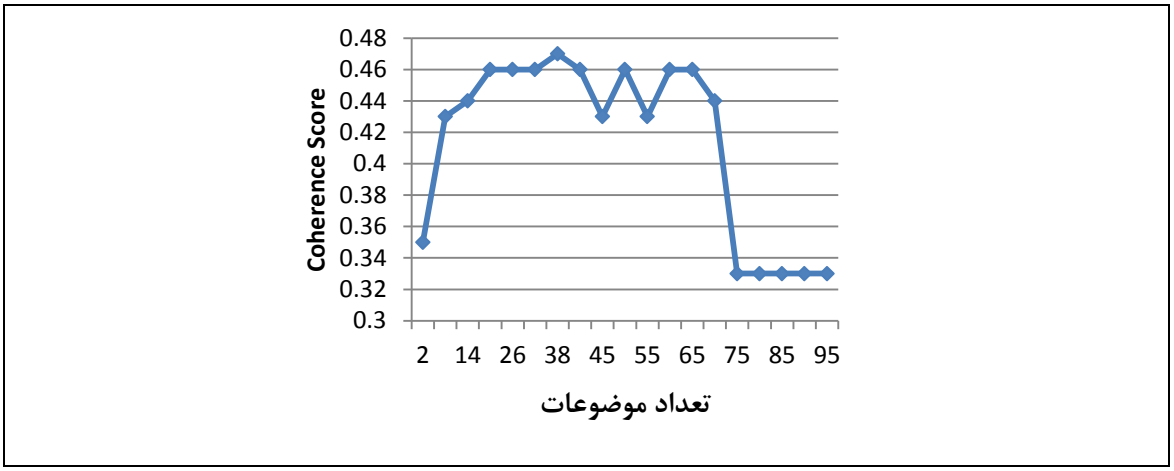
روش	تعداد ویژگی	Accuracy	Precision	Recall	F1	IBA
روش پیشنهادی	۴۸	**٪۹۸**	۰/۸۳	**۰/۹۰**	**۰/۸۵**	**۰/۸۷**
مقاله [109]	۲۷۵	–	–	۰/۴۴	۰/۴۱	–
مقاله[110]	۹۶	٪۹۳	–	–	۰/۳۴	–
مقاله [111]	۱۳۷	–	–	۰/۸۱	۰/۸۱	۰/۷۱
مقاله [112]	۷۰	٪۸۲	**۰/۸۳**	۰/۶۳	–	–
مقاله [42]	**۱۳**	٪۷۷	–	–	۰/۲۷	–
مقاله [113]	۵۹	–	–	–	۰/۷۸	–
مقاله [115]	۴۱	٪۹۵	–	–	–	–

همان‌طور که مشاهده می‌شود، مدل ارائه شده با تعداد ویژگی‌های کمتر، به نتایج بهتری دست پیدا کرده‌است. بخشی از این ویژگی‌ها شامل ویژگی‌های جدیدی که در این رساله معرفی شدند و افزودن آنها به ویژگی‌های قبلی باعث افزایش دقت طبقه‌بندی گردید به شرح جدول ۴-۱۵ می‌باشد.

جدول ۴-۱۵ لیست ویژگی‌های انتخاب شده‌از بین ویژگی‌های معرفی شده

نوع	شرح ویژگی
ساختار نشانی اینترنتی	حداکثر تعداد نقاط در دامنه
	تعداد نویسه‌های خاص در دامنه
	کمترین و میانگین تعداد ارقام در نشانی اینترنتی
	حداقل و میانگین طول نشانی اینترنتی
	حداقل و میانگین طول توکن‌های نشانی اینترنتی
	میانگین نرخ پیوستگی نویسه‌های الفبایی در توکن‌های نشانی اینترنتی
	بیشترین میزان شباهت توکن‌های دامنه به دامنه‌های معروف
برچسب‌های کد HTML	حداقل تعداد برچسب‌های <a>
	حداکثر تعداد برچسب‌های
	میانگین تعداد برچسب‌های <img>
	کمترین تعداد برچسب <script>
	تعداد برچسب‌های <iframe>
	تعداد برچسب‌های <link>
خوانایی متن	تعداد اسامی اشاره
	تعداد موضوعات با احتمال وجود بیشتر از یک درصد
	تعداد موضوعات با احتمال وجود بیشتر از ده درصد
	تعداد موضوعات با احتمال وجود بیشتر از چهل درصد

شایان ذکر است تعداد کل موضوعات در بخش ویژگی‌های مبتنی بر محتوای متن در این دادگان ۳۸ مورد است که براساس بیشترین میزان پیوستگی[1] انتخاب شده‌است (شکل ۴-۳).



شکل ۴-۳ تاثیر تعداد موضوعات بر میزان پیوستگی مدل LDA

۴-۶- جمع بندی

در این رساله فصل راهکار پیشنهادی را به بوته آزمایش گذاشتیم تا اثربخشی آن را بررسی کنیم. به این منظور ابتدا ویژگی‌های رایج در سه دسته محتوا محور و پیوند محور را مورد بررسی قرار دادیم. بهترین نرخ IBA با استفاده از مجموع ویژگی‌های هر دو دسته (۲۷۵ ویژگی) و الگوریتم Naïve Bayes به میزان ۰/۳۴ به دست آمد. سپس الگوریتم Smart-BT به منظور کاهش ویژگی معرفی شد. این روش به دنبال تشکیل بزرگترین زیرمجموعه از ویژگی‌ها است که حذف آن‌ها نه تنها باعث کاهش دقت طبقه‌بند نشود، بلکه آن را افزایش دهد. Smart-BT به عنوان پیش پردازش، ویژگی‌ها را با استفاده از معیار مربع کای رده‌بندی می‌کند. این رده‌بندی باعث کاهش بار محاسباتی الگوریتم می‌شود. استفاده از این دو روش ضمن کاهش تعداد ویژگی‌ها از ۲۷۵ به ۲۸، نرخ IBA را نیز به ۰/۴۱ افزایش داد. سپس تعدادی ویژگی جدید جهت افزایش نرخ تشخیص وب‌هرز شامل تعیین میزان پیوستگی مطالب صفحه با استفاده از نرخ حضور ضمایر اشاره در متن قابل رویت، نحوه توزیع موضوعات در صفحه، میانگین نرخ پیوستگی نویسه‌های الفبایی در توکن‌های مسیر و نرخ استفاده از

[1] Coherence Score

برخی برچسب‌های HTML مانند iftame، div، link و a نرخ IBA را به ۰/۶۹ افزایش داد. در نهایت با استفاده از مفهوم یابنده‌ها و سلول‌های حافظه در سیستم ایمنی مصنوعی، روشی برای تشخیص صفحات هرز ارائه شد که طبقه‌بندی داده‌ها با این روش به جای Naïve Bayes، میزان IBA را به ۰/۸۷ افزایش داد.

۵

جمع‌بندی و کارهای آتی

نتیجه گیری

کارهای آتی

۵-۱- مقدمه

در این رساله به منظور تشخیص صفحات وب‌هرز مدلی شامل دو بخش پیش‌پردازش و طبقه‌بندی ارائه شد. در بخش پیش‌پردازش دو بحث انتخاب و استخراج ویژگی مورد توجه قرار گرفت. در بخش اول الگوریتم Smart-BT که از نوع بسته‌بند[1] و بر اساس ارزیابی زیرمجموعه‌ها عمل می‌کند معرفی شده‌است. این روش به دنبال تشکیل بزرگترین زیرمجموعه از ویژگی‌ها است که حذف آن‌ها نه تنها باعث کاهش دقت طبقه‌بند نشود، بلکه آن را افزایش دهد. با اینکه این روش جزو روش‌های جامع[2] قرار می‌گیرد، از آنجا که در هر مرحله از نتایج مرحله قبل استفاده می‌شود، بار محاسباتی کمی دارد. در بخش دوم تعدادی ویژگی مبتنی بر کد منبع و محتوا معرفی شد. هدف از این کار توجه به ساختار یک صفحه غیرهرز و استخراج ویژگی هایی است که بیانگر میزان مفید بودن آن صفحه باشد تا نسبت به تغییر روشهای ایجاد صفحات هرز، پایداری بیشتری داشته باشد. در این زمینه ابتدا مفهوم انسجام را معرفی کردیم. سپس به بررسی برخی نشانه‌های آن از قبیل انسجام ساختاری و لغوی و نیز انسجام معنایی پرداختیم. برای بررسی انسجام لغوی، از نظریات هالیدی زبانشناس معروف و برای تشخیص انسجام معنایی از مدلسازی موضوعی به عنوان راهکار استفاده شد.

در نهایت در قسمت طبقه‌بندی با توجه به توزیع نامتوازن داده‌ها، از مفهوم یابنده با شعاع متغیر بر مبنای الگوریتم‌های انتخاب مثبت و جامعه سیستم ایمنی مصنوعی جهت طراحی طبقه‌بند استفاده شد. آزمایش‌های صورت گرفته نشان می‌دهد که این مدل در مقایسه با دیگر روش‌ها، علاوه بر استفاده از تعداد ویژگی‌های کمتر از عملکرد بهتری نیز برخوردار است.

[1] Wrapper
[2] Comprehensive

۵-۲- نتایج و دستاوردها

اهم دستاوردهای این رساله به شرح ذیل است:

- مقایسه الگوریتم‌های رایج طبقه‌بندی و انتخاب بهترین آنها به منظور طبقه‌بندی داده‌های نامتوازن
- مقایسه روشهای رایج انتخاب ویژگی و انتخاب بهترین آنها به منظور پیش پردازش داده ها
- ارائه الگوریتم Smart-BT که یک روش جدید انتخاب ویژگی است و خروجی آن نزدیک به بهینه‌ترین حالت ممکن است
- خاص منظوره کردن این روش برای مجموعه داده‌های نامتوازن به خصوص برای شناسایی وب‌هرز
- معرفی ویژگی‌های مبتنی بر کد منبع یک صفحه وب
- معرفی ویژگی‌های مبتنی بر انسجام لغوی متن
- معرفی ویژگی‌های مبتنی بر انسجام موضوعی متن
- بررسی ویژگی‌های رایج در شناسایی وب‌هرز و ویژگی‌های معرفی شده و انتخاب بهترین زیرمجموعه از آنها که قدرت تفکیک بالاتری دارند.
- ارائه یک روش طبقه‌بندی مبتنی بر سیستم ایمنی مصنوعی
- ارائه یک مدل کامل شامل مجموعه ویژگی و طبقه‌بند به منظور تشخیص صفحات وب‌هرز که در قیاس با دیگر روشها با هزینه محاسباتی پایین، عملکرد بالایی دارد.

۵-۳- کارهای آتی

در سالهای اخیر شناسایی ناهنجاری با استفاده از یادگیری عمیق رواج پیدا کرده‌است. به عنوان یکی از کارهای آتی می‌توان عملکرد این روش را در شناسایی صفحات وب‌هرز بررسی کرد. همچنین با توجه به اینکه روشهای هرزفرست‌ها هر روز جدیدتر می‌شود، تهیه مجموعه داده به روز و بررسی روش‌های رایج بر روی آن یکی دیگر از اقدامات آتی است. همچنین با توجه به اینکه برخی ویژگی‌های ارائه شده مختص زبان انگلیسی است، اگر بتوان مجموعه داده فارسی جمع آوری کرد و معادل این ویژگی‌ها را برای زبان فارسی به دست آورد، بسیار مفید می‌باشد. انسجام متن که در این رساله به آن پرداخته شد می‌تواند دقیق تر مورد بررسی قرار گیرد و ویژگی‌های دیگری به منظور تشخیص آن ارائه شود. نکته دیگری که میتوان به آن اشاره کرد، توجه به محتوای چند رسانه‌ای صفحات است. در این رساله این عناصر حذف شدند و تنها به متن استناد شده‌است. در کارهای آتی می‌توان محتوای آنها را نیز در نظر گرفت.

منابع

[1] D. Loiz, "Advanced Web Ranking," Caphyon, 5 June 2018. [Online]. Available: https://www.advancedwebranking.com/ctrstudy/. [Accessed 5 July 2018].

[2] P. N. Gupta, P. Singh, P. Singh, P. K. Singh and D. Sinha, "A Novel Architecture of Ontology based Semantic Search Engine," *International Journal of Science and Technology,* vol. 1, no. 12, pp. 650-700, 2012.

[3] S. Das, "Instant Messaging Spam Detection In Long Term Evolution Networks," Concordia Institute for Information Systems Engineering, Montreal, Quebec, Canada, 2013.

[4] B. Davison, "Recognizing nepotistic links on the web," in *Workshop on Artificial Intelligence for Web Search*, Austin, TX, 2000.

[5] M. Najork, Web Spam Detection, Berlin: Springer, 2009.

[6] Gyongyi, Z. and H. Garcia-Molina, "WebSpamTaxonomy," in *First International Workshop onAdversarial Information Retrieval on the Web (AIRWeb)*, Chiba, Japan, 2005.

[7] B. Wu and B. Davison, "Eliminating The Impact of Link Plagiarism On Web Search Rankings," in *2006 ACM symposium on Applied computing*, Dijon, France, 2006.

[8] N. Eiron, K. S. McCurley, and J. A. Tomlin, "Ranking The Web Frontier," in *13th International Conference on World Wide Web*, *WWW'04*, New York, NY, 2004.

[9] D. Fetterly, M. Manasse, and M. Najork, "Detecting Phrase-level Duplication on The World Wide Web," in *28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'05*, Salvador, Brazil, 2005.

[10] R. Jennings, "Cost of Spam is Flattening — Our 2009 Predictions," Ferris Research, San Francisco, CA, USA, 2009.

[11] A. A. Benczur, K. Csalogany, T. Sarlos, and M. Uher, "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*, Chiba, 2005.

[12] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri, "Know Your

Neighbors: Web Spam Detection Using The Web Topology," in *30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Amsterdam, The Netherlands, 2007.

[13] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly, "Detecting Spam Web Pages Through Content Analysis," in *15th International Conference on World Wide Web*, Edinburgh, Scotland, 2006.

[14] G. Salton, A. Wong, and C. S. Yang, "A Vector Space Model for Automatic Indexing," *Communications of the ACM,* vol. 18, no. 11, pp. 613-620, 1975.

[15] S. Robertson, H. Zaragoza, and M. Taylor, "Simple BM25 Extension to Multiple Weighted Fields," in *13th ACM International Conference on Information and Knowledge Management*, Washington, D.C., 2004.

[16] C. Zhai, Statistical Language Models for Information Retrieval, MA, USA: Hanover Inc, 2008.

[17] Luca Becchetti , Carlos Castillo , Debora Donato , Stefano Leonardi , Ricardo Baeza-Yates, "Link-Based Characterization and Detection of Web Spam," in *2nd International Workshop on Adversarial Information Retrieval on the Web*, Seattle, Washington, USA, 2006.

[18] N. Spirin and J. Han, "Survey on Web Spam Detection: Principles And Algorithms," *ACM SIGKDD Explorations Newsletter,* vol. 13, no. 2, pp. 50-64, 2011.

[19] M. Danandeh Oskuie and S. N. Razavi, "A Survey of Web Spam Detection Techniques," *International Journal of Computer Applications Technology and Research*, vol. 3, no. 3, pp. 180 - 185, 2014.

[20] L. Becchetti, C. Castillo, D. Donato, S. Leonardi and R. Baeza-Yates, "Using Rank Propagation and Probabilistic Counting for Link-based Spam Detection," in *The SIGKDD Workshop on Web Mining and Web Usage Analysis*, 2006.

[21] P. Boldi, M. Santini and S. Vigna, "PageRank as a Function of the Damping Factor," in *The 14th International Conference on World Wide Web*, Chiba, Japan, 2005.

[22] Z. Gyöngyi, H. Garcia-Molina and J. Pedersen, "Combating Web Spam with TrustRank," in *The 13th International Conference on Very Large Databases*, Toronto, Canada, 2004.

[23] V. Krishnan, "Web Spam Detection with Anti-trust Rank," in *The 2nd International Workshop on Adversarial Information Retrieval on the Web*, 2006.

[24] Lawrence Page , Sergey Brin , Rajeev Motwani , Terry Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford InfoLab, Texas, 1999.

[25] J. Kleinberg, "Authoritative Sources In a Hyperlinked Environment," *Journal of the ACM(JACM),* vol. 46, no. 5, pp. 604-632, 1999.

[26] T. Haveliwala, "Topic-Sensitive PageRank," in *11th International Conference on World Wide Web*, 2002.

[27] Z. Gyongyi, "Link Spam Detection Based On Mass Estimation," in *The 32th International Conference on Very Large Data Bases*, Seoul, Korea, 2006.

[28] M. Sobek, "PR0 - Google's PageRank 0 Penalty," eFactory, 2002. [Online]. Available: http://pr.efactory.de/e-pr0.shtml. [Accessed 2 January 2015].

[29] D. F. a. B. Racz, "Towards Scaling Fully Personalized PageRank," in *3rd Workshop on Algorithms and Models for the Web-Graph*, Rome, Italy, 2004.

[30] S. Gupta, A. K. Gupta, N. Tayal, "Spam Detection By Mass Estimation And Content Analysis," *International Journal Of Advanced Research In Engineering And Science*, vol. 1, no. 1, pp. 1-9, 2014.

[31] J. Caverlee and L. Liu, "Countering Web Spam With Credibility-Based Link Analysis," in *Twenty-Sixth Annual ACM Symposium on Principles of Distributed Computing*, Portland,Oregon, 2007.

[32] Bharat, K. and M.R. Henzinger, "Improved Algorithms for Topic Distillation In A Hyperlinked Environment," in *21st Annual International ACM SIGIR Conference on Research and Development In Information Retrieval*, Melbourne, Australia, 1998.

[33] R. Lempel and S. Moran, "The Stochastic Approach For Link-Structure Analysis (SALSA) And The TKC Effect," *Computer Networks*, vol. 33, no. 1, pp. 387- 401, 2000.

[34] G. Roberts and J. Rosenthal, "Downweighting Tightly Knit Communities In World Wide Web Rankings," *Advances and Applications in Statistics*, vol. 3, pp. 199-216, 2003.

[35] L. Li, Y. Shang and W. Zhang, "Improvement of HITS-Based Algorithms On Web Documents," in *11th International Conference On World Wide Web*, Honolulu, 2002.

[36] D. Zhou, O. Bousquet, T. Lal and J. Weston, "Learning with Local and Global Consistency," in *Advances in Neural Information Processing Systems 16*, 2003.

[37] Q. Gan and T. Suel, "Improving Web Spam Classification Using Link Structure," in *International Workshop on Adversarial Information Retrieval on the Web*, Banff, Alberta, 2007.

[38] R. C. Patil and D. R. Patil, "Web Spam Detection Using SVM Classifier," in *9th International Conference on Intelligent Systems and Control*, Coimbatore,

India, 2015.

[39] K. L. Goh and A. K. Singh, "Comprehensive Literature Review on Machine Learning Structures for Web Spam Classification," *Procedia Computer Science,* vol. 70, pp. 434-441, 2015.

[40] A. A. Torabi, K. Taghipour and S. Khadivi, "Web Spam Detection: New Approach with Hidden Markov Models," in *Asia Information Retrieval Technology*, 2013.

[41] L. Araujo and J. Martinez-Romo, "Web Spam Detection: New Classification Features Based on Qualified Link Analysis and Language Models," *IEEE Transactions on Information Forensics and Security,* vol. 5, no. 3, pp. 581 - 590, 2010.

[42] X. Y. Lu, M. S. Chen, J. L. Wu, P. C. Chang and M. H. Chen, "A Novel Ensemble Decision Tree Based on Under-sampling and Clonal Selection for Web Spam Detection," *Pattern Analysis and Applications,* pp. 1-14, 2017.

[43] H. Jelodari, Y. Wang, C. Yuan and X. Jiang, "A Systematic Framework to Discover Pattern For Web Spam Classification," in *The 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference*, Vancouver, BC, Canada, 2017.

[44] T. Singh, M. Kumari and S. Mahajan, "Feature Oriented Fuzzy Logic Based Web Spam Detection," *Information and Optimization Sciences,* vol. 38, no. 6, pp. 999-1015, 2017.

[45] S. Kumar, X. Gao, I. Welch and M. Mansoori, "A Machine Learning Based Web Spam Filtering Approach," in *International Conference on Advanced Information Networking and Applications*, Crans-Montana, Switzerland, 2016.

[46] A. Chandra, M. Suaib and R. Beg, "Web Spam Classification Using Supervised Artificial Neural Network Algorithms," *International Journal of Advanced Computational Intelligence,* vol. 2, no. 1, pp. 21-30, 2015.

[47] Y. Li, X. Nie and R. Huang, "Web Spam Classification Method Based on Deep Belief Networks," *Expert Systems With Applications,* vol. 96, no. 1, pp. 261-270, 2018.

[48] J. Fdez-Glez, D. Ruano-Ordas, J. R. Méndez, F. Fdez-Riverola, R. Laza and R. Pavón, "A Dynamic Model for Integrating Simple Web Spam Classification Techniques," *Expert Systems with Applications,* vol. 42, no. 21, p. 7969–7978, 2015.

[49] H. Li, "Web Spam Detection Based on Improved Tri-training," in *International Conference on Progress in Informatics and Computing*, Shanghai, China, 2014.

[50] R. M. Silva, T. A. Almeida and A. Yamakami, "Towards Web Spam Filtering Using a Classifier Based on the Minimum Description Length Principle," in *15th IEEE International Conference on Machine Learning and Applications*, Anaheim, CA, USA, 2016.

[51] J. Karimpour, A. Noroozi and A. Abadi, "The Impact of Feature Selection on Web Spam," *Intelligent Systems and Applications*, vol. 4, no. 9, pp. 61-67, 2012.

[52] Rungsawang, A., A. Taweesiriwate, and B. Manaskasemsak, "Spam Host Detection Using Ant Colony Optimization," in *3rd International Conference on Information Technology Convergence and Services*, Netherlands, 2011.

[53] Silva, R.M., A. Yamakami, and T.A. Alimeida, "An Analysis of Machine Learning Methods for Spam Host Detection," in *11th International Conference on Machine Learning and Applications (ICMLA)*, Campinas, Brazil, 2012.

[54] N. V. Tuc, A. C. Tcoi, and M. Hagenbuchner, "Cost-Sensitive Cascade Graph Neural Networks," in *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN*, Bruges, Belgium, 2013.

[55] J. S. Jegadeesh, P. L. Jacob, J. J. Spencer, and C. S. DevaKumar, "Web Spam Detection Using Fuzzy Clustering," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 1, no. 12, pp. 928-938, 2013.

[56] A. A. Soni, and A. Mathur, "Content Based Web Spam Detection Using Naive Bayes With Different Feature Representation Technique," *International Journal of Engineering Research and Applications*, vol. 3, no. 5, pp. 198-205, 2013.

[57] A. Keyhanipour, and B. Moshiri, "Designing a Web Spam Classifier Based on Feature Fusion in the Layered Multi Population Genetic Programming Framework," in *16th International Conference on Information Fusion (FUSION)*, Istanbul, 2013.

[۵۸] ف. اصدقی و ع. سلیمانی, "بررسی تاثیر کاهش ویژگی بر افزایش نرخ دقت تشخیص صفحات وب‌هرز," در *کنفرانس پردازش سیگنال و سیستم‌های هوشمند*, شاهرود, ۱۳۹۶.

[59] Y. Liu, F. Chen, W. Kong, H. Yu, M. Zhang, S. Ma and L. Ru, "Identifying Web Spam with the Wisdom of the Crowds," *ACM Transactions on the Web*, vol. 6, no. 1, 2012.

[60] C. Dong and B. Zhou, "Effectively Detecting Content Spam on the Web Using Topical Diversity Measures," in *IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology*, 2012.

[61] M. Luckner, M. Gad and P. Sobkowiak, "Stable Web Spam Detection Using Features Based on Lexical Items," *Computers & Security*, vol. 46, pp. 79-93,

2014.

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

[63] S. Wei and Y. Zhu, "Cleaning Out Web Spam by Entropy-Based Cascade Outlier Detection," in *International Conference on Database and Expert Systems Applications*, Lyon, France, 2017.

[64] A. Alarifi and M. Alsaleh, "Web Spam: a Study of the Page Language Effect on the Spam Detection Features," in *11th International Conference on Machine Learning and Applications*, Boca Raton, FL, USA, 2012.

[65] P. K. Karunakaran and S. Kolkur, "Language Model Issues in Web Spam Detection," *Enhanced Research in Management & Computer Applications*, vol. 3, no. 2, pp. 39-43, 2014.

[66] K. Hunagund and S. Kumar, "Spam Web Page Detection based on Content and Link Structure of the Site," *Advanced Research in Computer and Communication Engineering*, vol. 4, no. 8, pp. 348-351, 2015.

[67] S. Kumar, X. Gao and I. Welch, "Novel Features for Web Spam Detection," in *28th International Conference on Tools with Artificial Intelligence*, San Jose, CA, USA, 2016.

[68] Y. Suhara, H. Toda, S. Nishioka and S. Susaki, "Automatically Generated Spam Detection Based on Sentence-level Topic Information," in *22nd International Conference on World Wide Web*, Rio de Janeiro, Brazil, 2013.

[69] J. Wan, M. Liu and J. Yi, "Detecting Spam Webpages Through Topic and Semantics Analysis," in *Global Summit on Computer & Information Technology*, Sousse, Tunisia, 2015.

[70] M. S. I. Mamun, M. A. Rathore, A. Habibi Lashkari, N. Stakhanova and A. A. Ghorbani, "Detecting Malicious URLs Using Lexical Analysis," in *Network and System Security*, 2016.

[71] Mohammed A. Saleh , Hesham N. El mahdy , Talal Saleh, "Improvement of Arabic Spam Web pages Detection Using New Robust Features," *IOSR Journal of Computer Engineering*, vol. 16, no. 2, pp. 24-35, 2014.

[72] W. Wang, G. Zeng and D. Tang, "Using Evidence Based Content Trust Model for Spam Detection," *Expert System with Applications*, vol. 37, no. 8, pp. 5599-5606, 2010.

[73] J. Piskorski, M. Sydow, and D. Weiss, "Exploring Linguistic Features for Web Spam Detection: A Preliminary Study," in *4th International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'08*, Beijing, China, 2008.

[74] M. Sydow, J. Piskorski, D. Weiss, and C. Castillo, "Application of Machine Learning in Combating Web Spam," 2007.

[75] A. Benczur, K. Csalogany, and T. Sarlos, "Link-based Similarity Search to Fight Web Spam," in *Second Workshop on Adversarial Information Retrieval on the Web, AIRWeb'06*, Seattle, Washington, 2006.

[76] B. Wu, V. Goel, and B. D. Davison, "Propagating Trust and Distrust to Demote Web Spam," in *Workshop on Models of Trust for the web*, Edinburgh, Scotland, 2006.

[77] W. Wang and G. Zeng, "Content Trust Model for Detecting Web," in *The International Federation for Information Processing*, Boston, 2007.

[78] L. Araujo and J. Martinez-Romo, "Web Spam Identification Through Language Model analysis," in *5th International Workshop on Adversarial Information Retrieval on the Web*, Madrid, Spain, 2009.

[79] A. Pavlov and B. Dobrov, "Detecting Content Spam on the Web through Text Diversity Analysis," in *Seventh Spring Researchers Colloquium on Databases and Information Systems*, Moscow, Russia, 2011.

[80] V. M. Prieto, M. Álvarez, R. López-García and F. Cacheda, "Analysis and Detection of Web Spam by Means of Web Content," in *Multidisciplinary Information Retrieval*, Berlin, Heidelberg, 2012.

[81] M. Al-Kabi, H. Wahsheh and I. Alsmadi, "OLAWSDS: An Online Arabic Web Spam Detection System," *International Journal of Advanced Computer Science and Applications,* vol. 5, no. 2, pp. 105-110, 2014.

[82] L. Zhang, Y. Zhang, Y. Zhang and X. Li, "Exploring Both Content And Link Quality For Anti-Spamming," in *Computer and Information Technology, 2006. CIT'06. The Sixth IEEE International Conference on*, Seoul, 2006.

[83] Tian, Y., G.M. Weiss, and Q. Ma, "A Semisupervised Approach for Web Spam Detection Using Combinatorial Feature-Fusion," in *Graph Labelling Workshop and Web Spam Challemge at European Conference on Machine Learning and Principles and Practice of Knowledge Discovery*, 2007.

[84] S. Chakrabarti, M. Joshi and V. Tawde, "Enhanced Topic Distillation Using Text, Markup Tags, and Hyperlinks," in *The 24th Annual International ACM SIGIR Conference On Research and Development In Information Retrieval*, New Orleans, 2001.

[85] Y. Liu, M. Zhang, S. Ma, and L. Ru, "User Behavior Oriented Web Spam Detection," in *17th International Conference on World Wide Web, WWW'08*, Beijing, China, 2008.

[86] J. Ma, L. K. Saul, S. Savage and G. M. Voelker, "Identifying Suspicious URLs: An Application of Large-scale Online Learning," in *26th Annual*

International Conference on Machine Learning, Montreal, Quebec, Canada, 2009.

[87] F. Asdaghi, A. Solimani and M. Zahedi, "A Novel Set of Contextual Features for Web Spam Detection," *International Journal of Nonlinear Analysis And Application,* vol. 11, no. 1, pp. 321-339, 2020.

[88] M. A. K. Halliday and R. Hasan, Cohesion in English, London: Longman, 1976.

[89] S. Hoenisch, "Coherence and Cohesion in Text Linguistics," [Online]. Available: https://criticism.com/da/coherence.php. [Accessed 14 7 2019].

[90] D. Blei, A. Ng and M. Jordan, "Latent Dirichlet Allocation," *Machine Learning Research,* vol. 3, pp. 993-1022, 2003.

[91] B. V. Barde and A. M. Bainwad, "An Overview of Topic Modeling Methods and tools," in *International Conference on Intelligent Computing and Control Systems*, Madurai, India, 2017.

[92] G. Chandrashekar and F. Sahin, "A Survey On Feature Selection Methods," *Computers & Electrical Engineering,* vol. 40, no. 1, pp. 16-28, 2014.

[93] M. Gütlein, E. Frank, M. A. Hall and A. Karwath, "Large-Scale Attribute Selection Using Wrappers," in *The IEEE Symposium on Computational Intelligence and Data Mining*, 2009.

[94] . F. G. López, . M. G. Torres, . B. M. Batista, J. M. Pérez and M. Moreno-vega, "Solving Feature Subset Selection Problem by a Parallel Scatter Search," *European Journal of Operational Research,* vol. 169, no. 2, pp. 477-489, 2006.

[95] A. Moraglio, C. Di Chio and R. Poli, "Geometric Particle Swarm Optimisation," in *European Conference on Genetic Programming*, 2007.

[96] M. Deriche, "Feature Selection using Ant Colony Optimization," in *6th International Multi-Conference on Systems, Signals and Devices*, Djerba, Tunisia, 2009.

[97] A. S. S. Rani and R. R. Rajalaxmi, "Unsupervised Feature Selection Using Binary Bat Algorithm," in *2nd International Conference on Electronics and Communication Systems*, Coimbatore, India, 2015.

[98] M. Schiezaro and H. Pedrini, "Data Feature Selection Based on Artificial Bee Colony Algorithm," *Image and Video Processing,* no. 1, p. 47, 2013.

[99] L. C. Molina, L. Belanche and À. Nebot, "Feature Selection Algorithms: A Survey and Experimental Evaluation," in *International Conference on Data Mining*, 2002.

[100] J. Novakovic, "The Impact of Feature Selection on the Accuracy of Bayes

Classifier," in *18th Telecommunications Forum*, 2010.

[101] G.-A. Madara, P. Inese and A. Ludmila, "The Impact of Feature Selection on the Information Held in Bioinformatics Data," *Information Technology and Management Science,* vol. 18, no. 1, pp. 115-121, 2015.

[102] V. García, R. A. Mollineda and J. S. Sánchez, "Index of Balanced Accuracy: A Performance Measure for Skewed Class Distributions," in *4th Iberian Conference on Pattern Recognition and Image Analysis*, Portugal, 2009.

[103] P. Domingos and M. Pazzani, "On the Optimality of the Simple Bayesian Classifier under Zero-One Loss," *Machine Learning,* vol. 29, no. 2-3, pp. 103-130, 1997.

[104] F. Asdaghi and A. Soleimani, "An Effective Feature Selection Method for Web Spam Detection," *Knowledge-Based Systems,* vol. 166, pp. 198-206, 2019.

[105] "Web Spam Collection," Laboratory of Web Algorithmics, University of Milan, [Online]. Available: http://chato.cl/webspam/datasets/.

[106] D. Wang, D. Irani and C. Pu, "Evolutionary Study of Web Spam: Webb Spam Corpus 2011 Versus Webb Spam Corpus 2006," in *8th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, Pittsburgh, Pennsylvania, United States, 2012.

[107] V. García, R. A. Mollineda and J. S. Sánchez, "Theoretical Analysis of a Performance Measure for Imbalanced Data," in *International Conference on Pattern Recognition*, Istanbul, Turkey, 2010.

[108] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, "The WEKA Data Mining Software: An Update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10-18, 2009.

[109] J. Fdez-Glez, D. Ruano-Ordás, R. Laza, J. R. Méndez, R. Pavón and F. Fdez-Riverola, "WSF2: A Novel Framework for Filtering Web Spam," *Scientific Programming*, vol. 2016, pp. 1-18, 2016.

[110] M. Danandeh Oskouei and S. N. Razavi, "An Ensemble Feature Select Method to Detect Web Spam," *Information Technology and Multimedia,* vol. 7, no. 2, pp. 99-113, 2018.

[111] Y. Li, X. Nie and R. Huang, "Web Spam Classification Method Based on Deep Belief Networks," *Expert Systems With Applications*, vol. 96, no. 1, pp. 261-270, 2018.

[112] M. Shipra and J. Akanksha, "Feature Selection-Model-Based Content Analysis for Combating Web Spam," *Computer Science & Information Technology*, 2016.

[113] A. Makkar, M. Obaidat and N. Kumar, "FS2RNN: Feature Selection Scheme for Web Spam Detection Using Recurrent Neural Networks," in *IEEE Global*

Communications Conference (GLOBECOM), 2018.

[114] M. Luckner, "Practical Web Spam Lifelong Machine Learning System With Automatic Adjustment to Current Lifecycle Phase," *Security and Communication Networks*, 2019.

[115] A. Makkar and N. Kumar, "An Efficient Deep Learning-Based Scheme for Web Spam Detection In IoT Environment," *Future Generation Computer Systems*, 2020.

Abstract

Identifying Web Spam is one of the major challenges for search engines, and various methods have been proposed to identify them. Some of these methods focus on extracting the appropriate features from the webs and others emphasize on providing the appropriate classifier to increase accuracy. Apart from discussing the content, communication and validity of the pages, another group has considered the web graph as the basis of their diagnostic methods.

One of the most important challenges in this area is the constant change and updating of ranking algorithm deception techniques. For this reason, the use of features that are less counterfeit, plays an important role in increasing the detection rate and leverage. Also, due to the fact that new weeds are created every day, designing a ranking algorithm that can be trained and improved according to this data will increase the efficiency of the ranking algorithm.

For this purpose, in this dissertation, we intend to present a model including feature extractor and classifier in order to identify Web Spam pages. In this model, in addition to using some common features, some features have been extracted from some sources such as page address, html code and lexical and conceptual content of the page. Then, in order to increase the speed and reduce the size of the data, a feature selection algorithm called Smart-BT was developed. Finally, using the concept of detectors and memory cells in the artificial Safety system, a method for detecting webs is presented.

In designing this model, only the source code of web pages is used and the multimedia content that is in them (photos, videos, etc.) is not considered. The results of using this model in detecting spam web pages related to the WEBSPAM-UK data set, which is the most famous data set in this field, show an improvement in the balanced accuracy to the extent of 16% and reduction of the number of features to 65%.

Key word: Search Engine, Web Spam, Link-Based Feature, Content-Based Feature, Feature Selection, Text Coherence, Topic Modeling, Artificial Immune System.

Shahrood University of Technology

Faculty of computer engineering

Ph.D. Thesis in Artificial Intelligence

Web Spam Detection using Qualification Features

By: Faeze Asdaghi

Supervisor:

Dr. Ali Soleimani

Advisor:

Dr. Morteza Zahedi

October, 2020