# ارزیابی تأثیر منشأ ویژگی‌ها بر میزان دقت تشخیص وب‌هرز توسط الگوریتم‌های طبقه‌بندی

فریبا مستشارنژاد [۱]، سیدرضا کامل [۲]

[۱] دانشجوی کارشناسی ارشد نرم افزار، دانشکده فنی، مهندسی، دانشگاه آزاد اسلامی، مشهد، ایران

[۲] استادیار گروه نرم افزار، دانشکده فنی، مهندسی، دانشگاه آزاد اسلامی، مشهد، ایران



## چکیده

**امروزه با توجه به رشد اطلاعات در وب، موتورهای جستجو به عنوان یک ابزار برای ورود به دنیای وب مورد توجه قرار گرفته‌اند. آنها فهرستی از نتایج مرتبط با پرسش کاربر را در اختیار او قرار می‌دهند. از آنجا که اکثر کاربران تنها نتایج صفحه نخست و از آن میان فقط روی سه یا پنج پیوند اولیه را مورد بازدید قرار می‌دهند، حضور یک صفحه در نتایج بالای موتورهای جستجو به معنای بازدیدکننده بیشتر و نیز درآمد بیشتر است. در این میان وب هرز یک روش غیرقانونی و غیراخلاقی به منظور افزایش رتبه صفحات اینترنتی توسط فریب الگوریتم‌های موتورهای جستجو می باشد. از آنجا که کیفیت نتایج برای موتورهای جستجو اهمیت بسیاری دارد، روش‌های مختلفی برای تشخیص صفحات وب هرز ارائه شده است.**

**تاکنون بررسی های فراوانی بر روی مجموعه داده UK-WEBSPAM-2007 صورت گرفته و الگوریتم های طبقه بندی جدید و ترکیبی به نتایج خوبی رسیده اند اما هدف ما بررسی عملکرد الگوریتم های کلاسیک بر روی این مجموعه داده است و اینکه نشان دهیم از اینگونه الگوریتم ها نمی‌توان به صورت خام برای تشخیص وب هرز استفاده کرد و روشهای ترکیبی جدید گزینه مناسب تری در این خصوص است. در این مقاله قصد داریم تاثیر روش‌های منتخب طبقه‌بندی را بر میزان تشخیص این صفحات با در نظر گرفتن چگونگی انتخاب ویژگی‌ها، بررسی نماییم. بدین منظور از مجموعه داده UK-WEBSPAM-2007 استفاده کرده و ۱۲روش مختلف طبقه‌بندی را برای تشخیص صفحات وب هرز از دیگر صفحات بر روی حالات مختلف انتخاب ویژگی اعمال کردیم. بهترین نتیجه از اعمال الگوریتم‌های طبقه‌بندی بر ترکیب ویژگی‌های مبتنی بر محتوا و ویژگی‌های مبتنی بر پیوند بدست آمد.**

## کلمات کلیدی

**وب هرز، طبقه‌بندی، ویژگی‌های مبتنی بر محتوا، ویژگی‌های مبتنی بر پیوند ، ویژگی‌های مبتنی بر پیوند تبدیل یافته**

## ۱- مقدمه

امروزه با توجه به رشد اطلاعات در وب، موتورهای جستجو به عنوان یک ابزار برای ورود به وب مورد توجه قرار گرفته‌اند. آنها فهرستی از نتایج مرتبط با پرسش کاربر را در اختیار او قرار می‌دهند. روش قانونی برای افزایش رتبه سایت‌ها در فهرست نتایج موتورهای جستجو افزایش کیفیت صفحات سایت‌ها است، اما این روش زمان بر و پرهزینه است. روش دیگر استفاده از روش‌های غیرقانونی و غیراخلاقی برای افزایش رتبه در موتورهای جستجو است.

تحقیقاتی که در مرجع [1] و [2] صورت گرفت نشان می‌دهد که تقریبا ۸۵٪ کاربران تنها نتایج صفحه نخست را مورد بازدید قرار داده و از آن میان فقط روی سه یا پنج پیوند اولیه کلیک می‌کنند. در نتیجه حضور یک صفحه در نتایج بالای موتورهای جستجو به معنای بازدیدکننده بیشتر و نیز درآمد بیشتر است. به همین دلیل موتورهای جستجو برای آنکه بهترین نتایج ممکنه را در اختیار کاربران قرار دهند از الگوریتم‌های بسیار پیچیده‌ای استفاده می‌کنند.

در این میان وب هرز یک روش غیرقانونی و غیراخلاقی به منظور افزایش رتبه صفحات اینترنتی توسط فریب الگوریتم‌های موتورهای جستجو می‌باشد. هدف وب هرزها تغییر رتبه صفحات وب در نتایج جستجو است. به این صورت که رتبه صفحات را بیش از آنچه که باید نشان دهد تا ترجیحاً در میان ۱۰ نتیجه اول ظاهر شود. وب هرزها باعث کاهش کیفیت نتایج جستجو و در نتیجه اتلاف وقت کاربر می‌شوند. افزایش تعداد این صفحات، افزایش تعداد صفحات جستجو شده توسط خزشگرها[1] و مرتب شده توسط اندیس-گزارها[2] را در پی دارد. این امر موجب اتلاف منابع موتور جستجو و افزایش زمان پاسخگویی به کاربر می شود.

گاهی وب سایت‌های اسپم به عنوان وسیله‌ای برای گسترش بدافزار، دسترسی به محتوای ممنوعه و حملات فیشینگ به کار گرفته می‌شود. به عنوان مثال در مرجع[3] حدود ۱۰۰ میلیون صفحه با استفاده از الگوریتم PageRank رتبه‌دهی شدند و مشخص شد ۱۱ صفحه از ۲۰ نتیجه ارائه شده توسط موتورهای جستجو، سایت‌های هرزه‌نگاری بودند که به خاطر تغییر محتوا و لینک رتبه بالایی گرفته بودند. این امر باعث می‌شود که شرکت سازنده موتور جستجو مقادیر زیادی از منابع محاسباتی و ذخیره‌سازی را هدر دهد.

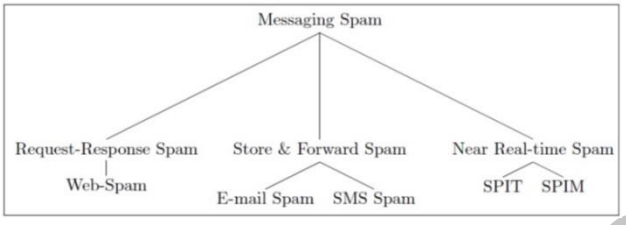
در تحقیقی که در سال ۲۰۰۵ صورت گرفت[4]، تخمین زده شد که وب هرزها سالانه حدود ۵۰ میلیون دلار باعث اتلاف منابع مالی می شوند. این مقدار در تحقیقی مشابه[5] که در سال ۲۰۰۹ صورت گرفت به ۱۳۰ میلیون دلار افزایش پیدا کرد. از جمله دلایل این رشد سریع تعداد وب هرزها می توان به ساده شدن ابزار ایجاد وب (از قبیل ویکی وب‌های رایگان، بسترهای بلاگ‌نویسی و ...) و کاهش هزینه نگهداری وب سایت اشاره کرد. تشخیص وب هرزها برای موتورهای جستجو از اهمیت بسزایی برخوردار بوده و در بازار رقابتی با سایر موتورهای جستجو جزو اولویت‌های کاری آنها محسوب می-شود.

تاکنون بررسی‌های فراوانی بر روی مجموعه داده UK-WEBSPAM-2007 صورت گرفته و الگوریتم های طبقه بندی جدید و ترکیبی به نتایج خوبی رسیده اند اما هدف ما بررسی عملکرد الگوریتم های کلاسیک بر روی این مجموعه داده است و اینکه نشان دهیم از اینگونه الگوریتم ها نمی‌توان به صورت خام برای تشخیص وب هرز استفاده کرد و روش‌های ترکیبی جدید گزینه مناسب تری در این خصوص است.

در بخش ۲ ابتدا انواع روش‌هایی که صفحات وب هرزها برای افزایش رتبه خود در موتورهای جستجو استفاده می‌کنند مورد بررسی قرار گرفته است. در بخش ۳ انواع ویژگی‌های قابل استخراج از صفحات وب توضیح داده شده است. در بخش ۴ توضیح مختصری از ۱۲ روش منتخب طبقه‌بندی استفاده شده بیان می‌شود. در بخش ۵ معیارهای ارزیابی مورد استفاده معرفی شده‌اند. در بخش ۶ روش‌های طبقه‌بندی مورد آزمایش قرار گرفته و نتایج بدست آمده از اعمال آنها بر حالات مختلف انتخاب ویژگی نشان داده شده است و در نهایت در بخش ۷ نتیجه گیری بررسی‌های انجام شده، عنوان می شود.

## ۲- بخش‌بندی مقاله

رایج‌ترین شکل اسپم، هرزنامه[3] است. جایگاه وب هرزها در دسته‌بندی انواع پیام‌های ناخواسته در شکل (۱) نشان داده شده است.




**شکل (۱) : رده‌بندی سطح بالای هرزنامه‌ها**

در سال ۱۹۹۵، Lycos اولین موتور جستجوی تجاری وارد دنیای وب شد. از همان زمان مبارزه با وب هرزها تحت عنوان spamdexing و با استفاده از روش‌های یادگیری ماشین تبدیل به یکی از مباحث مورد علاقه مجامع دانشگاهی شد [6] . از سال ۲۰۰۵، کارگاه‌های آموزشی AIRWeb به محلی برای تبادل اطلاعات در این زمینه تبدیل شد[7].

محققان وب هرزها را به سه دسته اصلی تقسیم می کنند:

- اسپم‌های مبتنی بر محتوا
- اسپم‌های مبتنی بر پیوند
- اسپم‌های مبتنی بر پنهان‌سازی صفحه

### ۲-۱- اسپم‌های مبتنی بر محتوا

اسپم محتوا اولین و گسترده‌ترین شکل وب هرزها است. دلیل گستردگی این شکل از وب هرز در این حقیقت نهفته است که موتورهای جستجو برای رتبه‌دهی به صفحات وب از مدل‌های بازیابی اطلاعات مبتنی بر محتوای صفحه از قبیل مدل فضای بردار [8]، BM25 [9] و مدل‌های آماری زبان[10] استفاده می‌کنند. به همین دلیل اسپمرها نقاط ضعف این مدل‌ها را تحلیل کرده و از آنها سوءاستفاده می کنند.

اسپم محتوا را به ۵ زیرگروه تقسیم می‌کنیم[11] .

- اسپم بدنه[4]: در این روش عبارات اسپم در بدنه صفحه قرار می‌گیرند. به عنوان نمونه اگر یک اسپمر بخواهد در صورت جستجوی یک سری کلمات محدود و از پیش تعیین شده به رتبه بالایی دست پیدا کند، کافی است آن کلمات و مشتقات آن را در صفحه تکرار کند.

- اسپم عنوان[5]: برخی موتورهای جستجو وزن بیشتری برای عنوان اسناد قائل هستند. به همین دلیل اسپمرها عبارات اسپم را در عنوان صفحه قرار می دهند.
- اسپم فرابرچسب[6]: توضیحات فرابرچسب‌های HTML این امکان را برای طراحان صفحات فراهم می کند که توضیحات کوتاهی راجع به صفحه قرار دهند. اگر کلمات نامربوط در این مکان قرار بگیرند و الگوریتم‌های موتورهای جستجو بر اساس این توضیحات صفحه را مورد بررسی قرار دهند، آنگاه این صفحات به خاطر کلمات نامربوطی که دارند، رتبه بالاتری دریافت می کنند.
- اسپم آدرس اینترنتی: برخی الگوریتم‌ها از عبارات موجود در URL سایت به منظور محاسبه رتبه استفاده می کنند. به همین دلیل اسپمرها برای

- صفحات، لینک‌های طولانی شامل کلمات عبارت جستجو شده ایجاد می کنند.
- اسپم متن Anchor: بخش Anchor یک لینک حاوی خلاصه‌ای از متن صفحه‌ای است که به آن اشاره می‌کند. بنابراین اسپمرها با قرار دادن عبارات دلخواه خود (که معمولا ارتباطی با محتوای صفحه ندارد) می‌توانند الگوریتم‌هایی که برای عبارات این بخش ارزش بیشتری قائل می شوند را فریب دهند و رتبه خود را افزایش دهند.

با ظهور الگوریتم‌های رتبه دهی پیوند محور، تا حدود زیادی بر مسئله اسپم-های محتوا محور غلبه شد. هر چند اسپمرها نیز به زودی راهکار مقابله با این روش را در ایجاد مزارع پیوند و دیگر روش‌ها یافتند. با این حال این جنگ و گریز همچنان ادامه دارد.

## ۲-۲- اسپم‌های مبتنی بر پیوند

دو دسته عمده از لینک اسپم ها وجود دارد: اسپم لینک‌های خروجی و اسپم لینک‌های ورودی.

- اسپم لینک‌های خروجی: این روش آسان‌ترین و ارزان‌ترین روش افزایش رتبه صفحات وب است. از آنجا که اسپمر مالک صفحه وب خود است و محتوای آن را می تواند تغیر دهد، به راحتی می تواند لینک هر صفحه‌ای که بخواهد را در صفحه خود قرار دهد یا حتی محتوای سایت‌هایی از قبیل DMOZ و Yahoo! Directory که حاوی فهرستی از آدرس صفحات اینترنتی هستند را در صفحه خود کپی کند. این روش الگوریتم‌های رتبه‌دهی همچون HITS را نشانه گرفته و در پی افزایش رتبه قطب خود هستند.
- اسپم لینک‌های ورودی: در این روش اسپمرها الگوریتم رتبه دهی PageRank را نشانه گرفته‌اند و بنابراین سعی در افزایش تعداد لینک‌های ورودی دارند. در اینجا بسته به اینکه اسپمر مالک صفحه باشد یا تنها به آن دسترسی داشته باشد، روش‌ها متفاوت است.

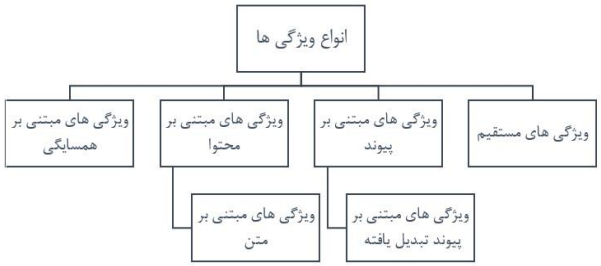
## ۲-۳- اسپم‌های مبتنی بر پنهان‌سازی صفحه

در این روش وب هرزها محتوای متفاوتی نسبت به آنچه کاربر می‌بینند را به موتورهای جستجو نمایش می‌دهند. انواع روش‌های آن عبارتند از:

- پنهان‌سازی : در این روش درخواست مشاهده وب سایت بررسی می-شود. اگر از سوی خزشگرهای موتورهای جستجو بود، محتوای متفاوتی نسبت به کاربر عادی به آن نمایش داده می‌شود.
- تغییر مسیر: وقتی کاربر از کلیک کردن بر روی پیوند نتایج موتور جستجو درخواست مشاهده صفحه را می‌کند، به صفحه دیگری منتقل می‌شود.

## ۳- انواع ویژگی‌ها

به طور کلی ویژگی های قابل استخراج از صفحات وب به ۴ دسته اصلی تقسیم می شوند که در شکل (۲) نشان داده شده است.




**شکل (۲) : دسته بندی ویژگی های مختلف صفحات وب**

دادگان مورد استفاده در این مقاله UK-WEBSPAM-2007 [12] می‌باشد که معتبرترین مجموعه داده در زمینه شناسایی صفحات وب هرز است. این دادگان شامل مجموعه صفحاتی است که از نتایج یک خزشگر در دامنه uk. بدست آمده است و شامل ۱۰۵/۹ میلیون صفحه از ۱۱۴۵۲۹ میزبان می باشد. از این صفحات سه دسته ویژگی استخراج شده است. ویژگی های مبتنی بر محتوا، ویژگی‌های مبتنی بر پیوند و ویژگی‌های مبتنی بر پیوند تبدیل‌یافته.

## ۳-۱- ویژگی های مبتنی بر محتوا

این دسته از ویژگی‌ها بر روی محتوای صفحات وب تمرکز دارند. برخی از این ویژگی‌ها در (جدول ۱) بیان شده است. در زبان انگلیسی برای ویژگی‌های مبتنی بر محتوا ۹۶ ویژگی استخراج گردیده است.

**جدول (۱) : برخی از ویژگی‌های مبتنی بر محتوا**

| ردیف | نام ویژگی |
|---|---|
| ۱ | تعداد کلمات در صفحه |
| ۲ | تعداد کلمات در عنوان |
| ۳ | میانگین طول کلمه |
| ۴ | بخشی از متن قابل مشاهده |
| ۵ | نرخ تراکم(فشرده سازی) صفحه |
| ۶ | ۱۰۰ فراخوانی برتر دیتاست |
| ۷ | بخشی از متن لینک داده شده |
| ۸ | بخشی از متن لینک داده شده |
| ۹ | ۲۰۰ دقت (درستی) برتر دیتاست |

## ۳-۲- ویژگی‌های مبتنی بر پیوند

این دسته از ویژگی‌ها بر روی پیوندهای موجود در صفحات وب تمرکز دارند. برخی از این ویژگی‌ها در (جدول ۲) بیان شده است. این ویژگی‌ها هم در صفحه اصلی میزبان[7] و هم در صفحه‌ای از میزبان که بالاترین رتبه[8] را دارد

محاسبه می‌شوند. در زبان انگلیسی برای ویژگی‌های مبتنی بر پیوند ۴۱ ویژگی استخراج گردیده است.

**جدول (۲) : برخی از ویژگی‌های مبتنی بر پیوند**

| ردیف | نام ویژگی |
|---|---|
| ۱ | تعداد پیوندهای خروجی صفحه اصلی |
| ۲ | تعداد پیوندهای ورودی صفحه اصلی |
| ۳ | نسبت تعداد پیوندهای خروجی داخلی به کل پیوندهای خروجی |
| ۴ | نسبت پیوند های خروجی صفحات دیگر به صفحه اصلی |
| ۵ | صفحاتی که به فاصله ۲ پیوند با صفحه اصلی هستند |
| ۶ | صفحاتی که به فاصله ۳ پیوند با صفحه اصلی هستند |
| ۷ | صفحاتی که به فاصله ۴ پیوند با صفحه اصلی هستند |
| ۸ | تعداد میزبان های متفاوت که در فاصله ۲ پیوند با صفحه اصلی هستند |

## ۳-۳- ویژگی‌های مبتنی بر پیوند تبدیل‌یافته

این دسته از ویژگی‌ها شامل تبدیلات عددی ساده و ترکیب ویژگی‌های مبتنی بر پیوند می‌شود. برخی از این ویژگی‌ها در (جدول ۳) بیان شده است. در زبان انگلیسی برای ویژگی‌های مبتنی بر پیوند تبدیل‌یافته ۱۳۸ ویژگی استخراج گردیده است.

**جدول (۳) : برخی از ویژگی‌های مبتنی بر پیوند تبدیل یافته**

| ردیف | نام ویژگی |
|---|---|
| ۱ | لگاریتم پیوندهای ورودی به صفحه اصلی |
| ۲ | لگاریتم پیوندهای خروجی از صفحه اصلی |
| ۳ | لگاریتم میانگین پیوندهای ورودی از صفحات خروجی به صفحه اصلی |
| ۴ | نسبت بین Indegree/PageRank |
| ۵ | نسبت بین TrustRank/PageRank |
| ۶ | لگاریتم ویژگی‌های مختلف |

## ٤- الگوریتم‌های طبقه‌بندی

در این مقاله ۱۲ الگوریتم طبقه‌بندی مورد استفاده قرار گرفته است که اکثر آنها جز برترین الگوریتم‌های طبقه‌بندی شناخته شده می‌باشند [13] و به شرح ذیل معرفی می‌گردند:

- C4.5: در روش درخت تصمیم، نمونه‌ها به یک الگوریتم C4.5 که یکی از درخت‌های تصمیم‌گیری است اعمال می‌شوند، سپس درخت هرس شده حاصل از الگوریتم C4.5 را گرفته و کلیه ویژگی‌هایی که در آن وجود دارد را بعنوان جواب مساله باز می‌گردانیم. الگوریتم C4.5 تکمیل شده الگوریتم ID3 می‌باشد.
- Naive Bayes: تئوری بیز یکی از روش‌های آماری برای رده‌بندی 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 و در نهایت، فرضیاتی که دارای بالاترین احتمال شوند، به عنوان یک کلاس در نظر گرفته شده و برچسبی بر آن ها زده می‌شود [14] .
- 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 در واقع یک متا الگوریتم است که بمظور ارتقاء عملکرد، و رفع مشکل رده‌های نامتوزان همراه دیگر الگوریتم‌های یادگیری استفاده می‌شود. در این الگوریتم، طبقه‌بند هر مرحله جدید به نفع نمونه‌های غلط طبقه‌بندی شده در مراحل قبل تنظیم می‌گردد. این الگوریتم از کل مجموعه داده به منظور آموزش هر دسته‌کننده استفاده می‌کند، اما بعد از هر بار آموزش، بیشتر بر روی داده‌های سخت تمرکز می‌کند تا به درستی کلاسه بندی شوند.
- KNN: 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 شبکه‌های عصبی، شبکه عصبی پرسپترون چند لایه(MLP) است. هنگام کار با شبکه های 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 شبکه‌های عصبی مبتنی بر مجموعه داده‌ها و ویژگی های آنهاست. الگوریتم‌های آموزشی متنوعی جهت آموزش شبکه‌های عصبی به کار می‌رود. متداول‌ترین الگوریتم آموزشی این شبکه‌ها، الگوریتم پس انتشار خطا می‌باشد. 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 وزن‌های شبکه پرداخته می‌شود. به نحوی که در انتهای هر تکرار اندازه خطای حاصله کمتر از میزان به دست آمده در تکرار قبلی باشد.
- Random Forest: این دسته‌بند یک روش بر مبنای مدل دسته جمعی است. این مدل، روش یادگیری برای طبقه‌بندی و رگرسیون است که با ساخت بسیاری از درخت‌های تصمیم‌گیری 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 می‌کند.
- 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 می-شوند که کار پیچیده‌ای 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 داده ها صورت می‌گیرد.

- logistic regression: رگرسیون لجستیک یکی از ابزارهای آماری است که به منظور مدل‌سازی و تحلیل داده‌ها از آن استفاده می‌شود. رگرسیون لجستیک دارای شکل کلی زیر است :

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

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 مشاهده‌ها، امکان پیش‌بینی احتمال تعلق هر فرد به هر یک از سطوح متغیر وابسته و همچنین امکان محاسبه‌ی مستقیم نسبت شانس با استفاده از ضرایب مدل است.

- 

- Logit boost: الگوریتم افزایندگی (Logit boost) به طریقی عمل می کند 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 طبقه‌بندی 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 C4.5 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 می کنند. استفاده از این معیار باعث می‌شود که درخت نسبت به حضور داده‌های پرت مقاوم تر باشند.

## ۵- معیارهای ارزیابی

پس از اعمال الگوریتم‌های طبقه‌بندی بر ویژگی‌های مورد نظر شاخص‌های حساسیت یا نرخ یادآوری[۹]، نرخ پیش بینی صحیح[۱۰]، دقت[۱۱]، معیار F [۱۲]و سطح زیر منحنی[۱۳] قابل اندازه گیری خواهند بود.

$$TPR\,(Recall) = \frac{TP}{(TP + FN)} \qquad (2)$$

$$PPR\,(Precision) = \frac{TP}{(TP + FP)} \qquad (3)$$

$$Accuracy = \frac{(TP + TN)}{(P + N)} \qquad (4)$$

$$F - Measure = \frac{2TP}{(2TP + FP + FN)} \qquad (5)$$

در روابط مذکور، TP تعداد موارد طبقه‌بندی مثبت صحیح (ماشین یادگیری به درستی طبقه‌بندی می‌کند)، TN تعداد موارد طبقه‌بندی منفی صحیح (ماشین یادگیری به درستی طبقه‌بندی می‌کند)، FP تعداد موارد طبقه-بندی مثبت نادرست (ماشین یادگیری به درستی طبقه‌بندی نمی‌کند)، FN تعداد موارد طبقه‌بندی منفی نادرست (ماشین یادگیری به درستی طبقه‌بندی نمی‌کند)، سطح زیر منحنی ROC است که عددی بین صفر و یک است و هرچه به یک نزدیکتر باشد بهتر است.

## ٦- پیاده‌سازی و نتایج

الگوریتم‌های اشاره شده در بخش ۴ ، توسط نرم افزار weka [14] بر روی دادگان مورد بحث اعمال شده است. مجموعه داده مورد استفاده WEBSPAM-UK2007 می‌باشد که در آن مجموعه تست از مجموعه آموزشی جدا است. در این مقاله سعی شده است علاوه بر استفاده از ویژگی-های مبتنی بر محتوا و ویژگی‌های مبتنی بر پیوند از دسته سوم ویژگی‌ها که مبتنی بر پیوند تبدیل یافته و همچنین ترکیب ویژگی‌ها استفاده شود. ۹۶ ویژگی مبتنی بر محتوا و ۴۱ ویژگی مبتنی بر پیوند و ۱۳۸ ویژگی مبتنی بر پیوند تبدیل یافته در مجموعه داده ذکر شده مد نظر قرار گرفته است.

جدول (۴) : نرخ تشخیص وب هرز با ویژگی‌های مبتنی بر محتوا

| روش | Accuracy | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| C4.5 | 92.25% | 0.289 | 0.23 | 0.256 | 0.57 |
| SVM | 94.20% | 0 | 0 | 0 | 0.5 |
| AdaBoost | 94.20% | 0 | 0 | 0 | 0.782 |
| KNN | 92.97% | 0.36 | 0.274 | 0.312 | 0.622 |
| Naïve Bayes | 14.32% | 0.058 | **0.903** | 0.109 | 0.653 |
| CART | 94.20% | 0 | 0 | 0 | 0.5 |
| Logistic Regression | 93.43% | 0.317 | 0.115 | 0.169 | 0.737 |
| Logit Boost | 94.20% | 0.5 | 0.097 | 0.163 | 0.78 |
| Model Tree | 94.61% | 0.643 | 0.159 | 0.255 | **0.794** |
| MLP | **94.66%** | **0.655** | 0.168 | 0.268 | 0.758 |
| Random Forest | 94.61% | 0.583 | 0.248 | **0.348** | 0.788 |
| LAD Tree | 94.35% | 0.579 | 0.097 | 0.167 | 0.762 |

**جدول (۵) : نرخ تشخیص وب هرز با ویژگی‌های مبتنی بر پیوند.**

| روش | Accuracy | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| Naïve Bayes | 92.46% | 0.133 | 0.049 | 0.072 | 0.643 |
| SVM | 94.06% | 0 | 0 | 0 | 0.5 |
| Logistic Regression | **93.82%** | 0.222 | 0.016 | 0.031 | 0.619 |
| MLP | 92.41% | 0.075 | 0.025 | 0.037 | 0.037 |
| KNN | 89.93% | 0.117 | **0.107** | 0.112 | 0.528 |
| AdaBoost | 94.06% | 0 | 0 | 0 | 0.702 |
| Logit Boost | 94.06% | 0 | 0 | 0 | **0.734** |
| C4.5 | 93.72% | 0.231 | 0.025 | 0.044 | 0.609 |
| LAD Tree | 93.48% | 0.269 | 0.057 | 0.095 | 0.71 |
| Model Tree | 94.06% | 0 | 0 | 0 | 0.609 |
| Random Forest | 93.38% | **0.281** | 0.074 | **0.117** | 0.615 |

| | | | | | |
|---|---|---|---|---|---|
| CART | 94.06% | 0 | 0 | 0 | 0.5 |

**جدول (٦) : نرخ تشخیص وب هرز با ویژگی‌های مبتنی بر پیوند تبدیل یافته**

| روش | Accuracy | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| Naïve Bayes | 82.48% | 0.128 | **0.336** | **0.186** | 0.683 |
| Svm | **94.06%** | 0 | 0 | 0 | 0.5 |
| logistic regression | 93.33% | 0.105 | 0.016 | 0.028 | 0.716 |
| MLP | 91.97% | 0.078 | 0.033 | 0.046 | 0.595 |
| KNN | 90.85% | 0.17 | 0.139 | 0.153 | 0.548 |
| Adaboost | **94.06%** | 0 | 0 | 0 | 0.673 |
| logit boost | 93.97% | 0 | 0 | 0 | **0.723** |
| C4.5 | 92.75% | 0.154 | 0.049 | 0.075 | 0.631 |
| LAD Tree | 93.63% | 0.263 | 0.041 | 0.071 | 0.696 |
| model tree | **94.06%** | 0 | 0 | 0 | 0.629 |
| random forest | 93.53% | **0.28** | 0.057 | 0.095 | 0.634 |
| CART | 94.06% | 0 | 0 | 0 | 0.5 |

**جدول (۷) : نرخ تشخیص وب هرز با ترکیب ویژگی‌های مبتنی بر پیوند و مبتنی بر پیوند تبدیل یافته**

| روش | Accuracy | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| Naïve Bayes | 84.24% | 0.134 | **0.304** | **0.186** | 0.689 |
| Svm | 94.10% | 0 | 0 | 0 | 0.5 |
| logistic regression | 92.97% | 0.133 | 0.035 | 0.055 | **0.709** |
| MLP | 92.25% | 0.154 | 0.07 | 0.096 | 0.628 |
| KNN | 91.12% | 0.17 | 0.13 | 0.148 | 0.545 |
| Adaboost | 94.10% | 0 | 0 | 0 | 0.71 |
| logit boost | 93.99% | 0 | 0 | 0 | 0.724 |
| C4.5 | 93.63% | 0.154 | 0.017 | 0.031 | 0.636 |
| LAD Tree | **93.69%** | 0.25 | 0.035 | 0.061 | 0.702 |
| model tree | 94.10% | 0 | 0 | 0 | 0.625 |
| random forest | 93.38% | **0.281** | 0.078 | 0.122 | 0.644 |
| CART | 94.10% | 0 | 0 | 0 | 0.5 |

**جدول (۸) : نرخ تشخیص وب هرز با ترکیب ویژگی‌های مبتنی بر محتوا و مبتنی بر پیوند تبدیل یافته**

| روش | Accuracy | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| Naïve Bayes | 43.89% | 0.075 | **0.761** | 0.136 | 0.668 |
| Svm | 94.20% | 0 | 0 | 0 | 0.5 |
| logistic regression | 92.92% | 0.333 | 0.221 | **0.266** | 0.725 |
| MLP | 93.43% | 0.333 | 0.133 | 0.19 | 0.691 |
| KNN | 91.68% | 0.198 | 0.142 | 0.165 | 0.553 |
| Adaboost | 94.56% | **0.889** | 0.071 | 0.131 | 0.787 |
| logit boost | 94.61% | 0.75 | 0.106 | 0.186 | 0.779 |
| C4.5 | 91.63% | 0.245 | 0.212 | 0.227 | 0.502 |
| LAD Tree | 93.94% | 0.419 | 0.115 | 0.181 | **0.788** |
| model tree | 94.46% | 0.593 | 0.142 | 0.229 | 0.762 |
| random forest | **94.71%** | 0.692 | 0.159 | 0.259 | 0.76 |
| CART | 93.89% | 0.375 | 0.08 | 0.131 | 0.599 |

**جدول (۹) : نرخ تشخیص وب هرز با ترکیب ویژگی‌های مبتنی بر محتوا و مبتنی بر پیوند**

| روش | Accuracy | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| Naïve Bayes | 22.64% | 0.061 | **0.858** | 0.114 | 0.674 |
| Svm | 94.20% | 0 | 0 | 0 | 0.5 |
| logistic regression | 93.33% | 0.302 | 0.115 | 0.167 | 0.733 |
| MLP | 94.51% | 0.594 | 0.168 | 0.262 | 0.696 |
| KNN | 93.07% | 0.38 | 0.31 | 0.341 | 0.639 |
| Adaboost | 94.56% | **0.889** | 0.071 | 0.131 | **0.787** |
| logit boost | 94.56% | 0.684 | 0.115 | 0.197 | 0.779 |
| C4.5 | 92.30% | 0.282 | 0.212 | 0.242 | 0.526 |

| | | | | | |
|---|---|---|---|---|---|
| LAD Tree | 93.84% | 0.387 | 0.106 | 0.167 | 0.773 |
| model tree | 94.51% | 0.607 | 0.15 | 0.241 | 0.75 |
| random forest | **94.97%** | 0.703 | 0.23 | **0.347** | 0.753 |
| CART | 94.10% | 0.429 | 0.053 | 0.094 | 0.599 |

**جدول (۱۰) : نرخ تشخیص وب هرز با ترکیب هر سه دسته ویژگی.**

| روش | Accuracy | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|
| Naïve Bayes | 69.68% | 0.115 | **0.636** | 0.195 | 0.678 |
| Svm | 94.21% | 0 | 0 | 0 | 0.5 |
| logistic regression | 92.20% | 0.266 | 0.196 | 0.226 | 0.709 |
| MLP | 93.88% | 0.35 | 0.065 | 0.11 | 0.612 |
| KNN | 91.72% | 0.189 | 0.131 | 0.155 | 0.548 |
| Adaboost | 94.53% | **0.875** | 0.065 | 0.122 | 0.787 |
| logit boost | **94.59%** | 0.733 | 0.103 | 0.18 | 0.782 |
| C4.5 | 91.61% | 0.25 | 0.224 | **0.236** | 0.511 |
| LAD Tree | 93.94% | 0.414 | 0.112 | 0.176 | **0.788** |
| model tree | 94.37% | 0.556 | 0.14 | 0.224 | 0.759 |
| random forest | 94.26% | 0.517 | 0.14 | 0.221 | 0.755 |
| CART | 93.88% | 0.375 | 0.084 | 0.137 | 0.593 |

می توان عنوان داشت که Naïve Bayes در صورتی که بر روی ویژگی‌های محتوا محور اعمال شود (جدول ۴)، بالاترین نرخ Recall (۹۰٪) را در میان دیگر روش‌ها دارد. این نرخ بالا به بهای نرخ بسیار اندک precision (۶٪) و دقت (۱۴٪) به دست آمده است. بالاترین نرخ precision و F-Measure نیز از اعمال روش Random Forest بر روی ترکیب ویژگی‌های محتوا و پیوند محور به دست آمده است (جدول ۹) که به ترتیب ۹۵٪ و ۰/۳۵ می‌باشد اما در قبال نرخ Recall ۲۳٪ به دست آمده است.

در مورد کارایی ویژگی‌های انتخاب شده نیز با نگاهی به نتایج می‌توان مشاهده کرد که ویژگی‌های پیوند محور به خودی خود دارای ارزش تفکیک کنندگی بالایی نیستند، اما در کنار ویژگی‌های محتوا محور منجر به افزایش دقت آن می‌شوند و از آنجا که توزیع داده در این دادگان یکسان نیست و تعداد نمونه‌های کلاس صفحات نرمال بیش از ۱۷ برابر نمونه‌های کلاس صفحات اسپم است، حتی اگر کلیه صفحات اسپم به اشتباه نرمال تشخیص داده شوند، دقت طبقه‌بند بیش از ۹۴٪ خواهد شد.

## ۷- نتیجه گیری

در این مقاله انواع روش هایی که وب هرزها برای بالا بردن رتبه خود در موتورهای جستجو استفاده می کنند را معرفی کردیم. سپس انواع ویژگی های استخراج شده از صفحات وب به منظور تشخیص وب هرزها را بررسی کردیم و در نهایت ۱۲ روش طبقه بندی منتخب را در جستجوی بهترین نتیجه، بر روی مجموعه داده WEBSPAM-UK2007 پیاده سازی کردیم. از آنجا که ویژگی های مورد استفاده نقش مهمی در بالا بردن دقت طبقه بند دارند، این روش ها به تفکیک بر روی ویژگی های مبتنی بر محتوا ، مبتنی بر پیوند و مبتنی بر پیوند تبدیل یافته و همچنین ترکیب این ویژگی ها آزمایش شد. بر اساس بررسی های صورت گرفته، توزیع داده در این دادگان یکسان نیست و تعداد نمونه های کلاس صفحات نرمال بیش از ۱۷ برابر نمونه های کلاس صفحات اسپم است؛ بنابراین حتی اگر کلیه صفحات اسپم به اشتباه نرمال تشخیص داده شوند، دقت طبقه بند بیش از ۹۴٪ خواهد شد. به همین دلیل نتایج بدست آمده کمی دور از انتظار است و برخی از الگوریتم ها مثل SVM به نتایج خوبی نرسیدند و در نتیجه Accuracy به تنهایی ملاک خوبی برای

ارزیابی روشهای طبقه بندی نخواهد بود. در نهایت مشاهده می شود که ویژگی ها و روش های کلاسیک طبقه بندی راه حل مناسبی برای تشخیص صفحات وب هرز نیست و می بایست از روش و ویژگی های جدیدی برای حل آن استفاده کرد.

## مراجع


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

[2] T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay, *"Accurately interpreting click through data as implicit feedback"*. In Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'05, Salvador, Brazil, 2005.

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

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

[5] R. Jennings, *"Cost of spam is fattening our 2009 predictions"*, Ferris Research, 2009.

[6] Davison, B.D., *"Recognizing nepotistic links on the web. Artificial Intelligence for Web Search"*, 2000:p. 23-28.

[7] Najork, M., Web Spam Detection. Encyclopedia of Database Systems, 1: p. 3520-3523, 2009.

[8] G. Salton, A. Wong, and C. S. Yang, *"A vector space model for automatic indexing"*. Common. ACM, Vol.18, Nov. 1975.

[9] S. Robertson, H. Zaragoza, and M. Taylor, *"Simple bm25 extension to multiple weighted fields"*. In Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, CIKM'04, Washington, D.C., 2004.

[10] C. Zhai, *"Statistical Language Models for Information Retrieval"*. Now Publishers Inc., Hanover, MA, USA, 2008.

[11] Gyongyi, Z. and H. Garcia-Molina, WebSpamTaxonomy, in First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb 2005). 2005.

[12] Carlos Castillo, Debora Donato, Luca Becchetti, Paolo Boldi, Stefano Leonardi, Massimo Santini and Sebastiano Vigna, *"A Reference collection for Web Spam"*, ACM SIGIR Forum, Vol. 40, No. 2, pp. 11-14, 2006.

[13] XindongWu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, Angus Ng, Bing Liu, Philip S. Yu, Zhi-Hua Zhou, Michael Steinbach, David J. Hand, Dan Steinberg, *"Top 10 algorithms in data mining"*, Springer, 1-37,2008.

[14] George H. John, Pat Langley, *"Estimating Continuous Distributions in Bayesian Classifiers"*. In: Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, 338-345, and 1995.

[15] http://www.cs.waikato.ac.nz/~remco/weka.pdf.


## زیرنویس‌ها

[1] Crawler
[2] Indexer
[3] Email Spam
[4] Body Spamming
[5] Title Spamming
[6] Meta Tag
[7] Home Page
[8] PageRank
[9] Recall
[10] Precision
[11] Accuracy
[12] Accuracy
[13] AUC