الگوریتمی برای مدیریت و دسته بندی نامه های الکترونیکی

آزاده قندهاری[۱] و حمید بیگی[۲]

چکیده

میزان دریافت نامه های الکترونیکی، پیوسته در حال افزایش می باشد و روزانه وقت زیادی جهت فیلترنمودن و سازماندهی آنها در پوشه ها صرف می گردد، بنابراین نیاز به سیستمی احساس می شود که بطور اتوماتیک جهت طبقه بندی و مدیریت نامه های الکترونیکی مورد استفاده قرار گیرد. سیستمهای طبقه بندی مختلفی توسط پژوهشگران، جهت دسته بندی اتوماتیک نامه های الکترونیکی ارائه شده اند. در این مقاله الگوریتمی جهت دسته بندی نامه های الکترونیکی برای نرم افزار Microsoft Outlook ارائه شده که در آن از طبقه بندی کننده نزدیکترین همسایه استفاده شده است. این الگوریتم به مدیریت inbox کاربر کمک میکند تا نامه ها را بر اساس پوشه های تعریف شده توسط کاربر، دسته بندی کند. مزیت اصلی این سیستم، عدم نیاز به داده های آموزشی زیاد، بار محاسباتی کم و دقت بالا می باشد.

کلمات کلیدی : مدیریت نامه های الکترونیکی، دسته بندی، طبقه بندی کننده نزدیکترین همسایه

An Algorithm for Email Categorization and Management

Azadeh Ghandehari and Hamid Beigy

Abstract

The volume of e-mail that we get is constantly growing. We are spending more and more time for filtering e-mails and organizing them into folders. Therefore, a system that can automatically learn how to classify e-mails into a set of folders is highly desirable. Many researchers have proposed classification systems that automatically classify emails. In this paper, we describe the implementation of an algorithm for Microsoft Outlook 2000 that intends to help manage the inbox by automatically classifying email based on user folders by using k-nearest neighbor classifier. The main advantages of this system are low computational time with high accuracy in classification and no need to have lots of training examples.

Keywords: E-mails management, classification, K-Nearest Neighbor classifier

[1] دانشجوی کارشناسی ارشد دانشگاه صنعتی امیر کبیر، دانشکده مهندسی کامپیوتر ، a_ghandehari@aut.ac.ir
[2] دانشگاه صنعتی شریف، دانشکده مهندسی کامپیوتر ، beigy@ce.sharif.edu

۱- مقدمه

زیاد بودن اطلاعات و نامه های الکترونیکی، مشکل عمده و قابل ملاحظه ای در طی چندین سال اخیر بوده است و برای کاربران، دریافت روزانه صدها نامه الکترونیکی، عادی می باشد. امروزه، میزان دریافت نامه های الکترونیکی افزایش چشمگیری را داشته و وقت زیادی برای فیلتر کردن و طبقه بندی این نامه ها جهت تسهیل و کمک به بازیابی آنها صرف می شود. اکثر بسته های نرم افزاری که برای دسته بندی نامه های الکترونیکی وجود دارند، عموماً شامل یک پایگاه قوانین می باشند که برای سازماندهی و طبقه بندی نامه ها، ایجاد شده و نامه های غیر درخواستی[1] را بر اساس وجود یک لغت کلیدی در عنوان یا متن نامه مشخص می کنند. با این وجود، اکثر کاربران تمایلی به تنظیم این نرم افزارها ندارند. همچنین، ایجاد یک پایگاه قوانین کامل و بدون نقص، بصورت دستی برای کاربرانی که پیوسته در حال ایجاد، حذف و تغییر نامه های الکترونیکی و پوشه های مربوط به آنها می باشند، بسیار مشکل است. حتی اگر، پوشه های مربوط به نامه ها، همواره یکسان باشند، طبیعت و ساختار نامه هایی که درون آن پوشه ها دسته بندی شده اند به مرور زمان تغییر میکند. بنابراین، پایگاه قوانین باید پیوسته توسط کاربر تنظیم شود که فرآیندی وقت گیر بوده و عاری از خطا نمی باشد. بنابراین، نیاز به یک سیستمی که بتواند بطور اتوماتیک یاد بگیرد که چگونه نامه های الکترونیکی را درون یک مجموعه پوشه (دسته) از پیش تعیین شده، دسته بندی کند، احساس می شود.

بیشتر پژوهشهای انجام شده بر روی الگوریتمهای یادگیری ماشین جهت دسته بندی نامه ها در پوشه های مربوطه متمرکز بوده است. بدین صورت که ابتدا از کاربر خواسته می شود که برای هر پوشه، یک تعداد از نامه ها را که متعلق به آن پوشه می باشند را تعیین کند؛ از این نامه ها بعنوان نمونه های آموزشی مثبت برای الگوریتم طبقه بندی کننده استفاده می شود. در برخی موارد، ممکن است که از کاربر خواسته شود که برای هر پوشه، نامه هایی را نیز مشخص کند که متعلق به آن پوشه نمی باشند (نمونه های آموزشی منفی). نظر به اینکه از چه الگوریتمی جهت یادگیری استفاده می شود، زمان محاسباتی لازم برای فرآیند آموزش تغییر می کند. پس از آموزش طبقه بندی کننده، یک نامه جدید به طبقه بندی کننده داده می شود تا پوشه مناسب را برای آن تعیین کند.

سیستمهای متعددی برای دسته بندی اتوماتیک نامه های الکترونیکی بر اساس روشهای یادگیری ماشین مورد استفاده در طبقه بندی متون توسعه یافته اند که می توان به سیستم های مبتنی بر قوانین تصمیم گیری [۳و۶]، طبقه بندی کننده های بیزین [۴]، ماشین بردار پشتیبان [۵]، شبکه های عصبی و روشهای مبتنی بر نمونه[2] [۹] اشاره کرد که برخی از آنها نتایج بسیار خوبی داشته اند.

کوهن [۱۰]، از یک عامل یادگیر گزاره ای برای ایجاد قوانینی جهت تشخیص واژگان کلیدی استفاده کرد تا نامه های الکترونیکی را درون پوشه های مربوطه بایگانی کند. ساهامی و همکارانش[۸]، برای فیلتر کردن نامه های اسپم از شبکه های بیزین استفاده کردند. سیستم iFile رنی [۷] از روش بیز ساده برای دسته بندی نامه ها در پوشه های مربوطه بهره میبرد. دو روش بیز ساده (NB) و نزدیکترین همسایه (k-NN) کارآیی چشمگیری در فیلتر کردن نامه های الکترونیکی بر اساس واژه های کلیدی و بر روی دادگان LingSpam داشته اند[۲].

در این مقاله سیستمی طراحی و پیاده سازی شده است که به کاربر کمک می کند تا نامه های الکترونیکی خود را که از طریق Microsoft Outlook دریافت می کند را با توجه به پوشه هایی که توسط خود کاربر مشخص شده، طبقه بندی کند. این سیستم شامل روتینهایی برای استخراج ویژگی، حذف حروف عطف و حروف اضافه و انتخاب پوشه مناسب برای نامه با استفاده از طبقه بندی کننده نزدیکترین همسایه می باشد.

ساختار این مقاله اینگونه تنظیم شده است که در بخش دوم، چگونگی دسته بندی متون که دسته بندی نامه های الکترونیکی، یکی از کاربردهای آن می باشد، بررسی شده و در بخش سوم، الگوریتم پیشنهادی جهت مدیریت نامه های دریافت شده از طریق نرم افزار Microsoft Outlook توضیح داده شده و در نهایت، در بخش چهارم نتایج حاصل از پیاده سازی انجام شده، ارائه گردیده است.

۲- دسته بندی متون[3]

از ۱۰ سال پیش تاکنون بدلیل افزایش اسناد و متون دیجیتالی و نیاز به سازماندهی آنها، طبقه بندی یا دسته بندی اتوماتیک متون در دسته های از پیش تعیین شده مورد توجه زیاد پژوهشگران واقع شده است. در پژوهشهای انجام شده در این زمینه، بهترین و سودمندترین روش مورد استفاده برای سازماندهی این اسناد دیجیتالی، روشهای مبتنی بر الگوریتمهای یادگیری ماشین می باشد؛ یعنی یک فرآیند استنتاج کلی که بطور اتوماتیک با یادگیری از یک مجموعه اسناد از پیش طبقه بندی شده و خصوصیات دسته های موجود، یک طبقه بندی کننده ایجاد می کند. مراحل اصلی این فرآیند در این بخش آورده شده است.

۲-۱ استخراج ویژگی

جهت استخراج ویژگی در طبقه بندی متون از روشهای متعددی استفاده می شود که یکی از آنها ساختار tf-idf[4] میباشد که در ادامه شرح داده میشود. این ساختار، یک روش آماری است که میزان اهمیت یک واژه را برای یک سند متنی ارزیابی می کند. میزان اهمیت یک واژه به نسبت تعداد دفعاتی که در یک سند متنی بکار رفته افزایش می یابد، امّا این درجه اهمیت با توجه به اینکه در کل اسناد موجود، عموماً چند مرتبه از این واژه استفاده شده، تنظیم می شود. از ساختار

tf-idf معمولاً در موتورهای جستجو برای یافتن مناسبترین اسناد موجود به درخواست و تقاضای کاربر استفاده می شود.

در این ساختار، tf نشان دهنده تعداد دفعاتی است که از یک واژه در یک سند استفاده شده است و میزان اهمیت واژه t_i را در یک سند مشخص، تعیین می کند و بصورت زیر قابل محاسبه می باشد:

$$tf = \frac{n_i}{\sum_k n_k} \qquad (۱)$$

که ni تعداد دفعات رخداد واژه شماره t_i و مخرج کسر نشان دهنده مجموع رخداد تمامی واژه های انتخاب شده از سند می باشد. همچنین، idf مشخص کننده اهمیت کلی و سراسری یک واژه می باشد که بصورت زیر محاسبه می گردد.

$$idf = \log \frac{|D|}{|(t_i \subset d_i)|} \qquad (۲)$$

که در آن $|D|$ مشخص کننده تعداد همه اسناد موجود و $|(t_i \subset d_i)|$ تعداد اسنادی است که در آنها از واژه t_i استفاده شده است. بنابراین، tf-idf = tf . idf

۲-۲ طبقه بندی کننده نزدیکترین همسایه

جهت طبقه بندی متون (و نامه های الکترونیکی) روشهای مختلفی مورد استفاده قرار گرفته است که یکی از متداولترین این روشها، روش نزدیکترین همسایه می باشد. علت انتخاب این طبقه بندی کننده، سادگی در پیاده سازی، سرعت و راحتی در آموزش آن و همچنین انجام دسته بندی بصورت محلی و عدم نیاز به نمونه های منفی است.

طبقه بندی کننده k-NN، جزء خانواده الگوریتمهای یادگیری مبتنی بر نمونه بوده و دسته بندی را تا ورود یک نمونه جدید به تأخیر می اندازد. جهت طبقه بندی یک نامه جدید و مشخص کردن پوشه مربوط به آن، این نامه با تمام نامه های موجود در دسته ها مقایسه شده و با توجه به مقدارk، تعداد kتا از نامه های شبیه به نامه آزمایشی از هر دسته انتخاب می شود. آنگاه برای هر دسته، میانگین میزان شباهت محاسبه شده و نامه آزمایشی به دسته ای تخصیص داده می شود که بیشترین میانگین شباهت را داشته باشد. میزان کارآیی این طبقه بندی کننده، وابستگی زیادی به انتخاب مناسب مقدار k دارد.

۲-۳ معیار شباهت

عموماً، الگوریتمهای یادگیری مورد استفاده در طبقه بندی نامه های الکترونیکی، مبتنی بر استفاده از مقادیر بردارهای tf-idf می باشد. از مقادیر این بردارها برای ارائه نامه های الکترونیکی و همچنین محتویات یک پوشه استفاده می گردد [۱،۱۱]. سپس می توان بردار نامه ها و بردار پوشه ها را با بکارگیری ضرب نقطه ای[۵] (معیار کسینوس)، مقایسه کرد. یک نامه الکترونیکی در پوشه ای قرار می گیرد که بردار مربوط به آن بیشترین شباهت را به بردار پوشه داشته باشد. اگر از چنین سیستمی جهت مقایسه استفاده شود، هر نامه می تواند فقط متعلق به یک پوشه باشد. در این مقاله نیز فرض شده که هر نامه نمی تواند متعلق به چند و یا هیچ پوشه ای باشد.

برای مقایسه یک نامه جدیدی که به سیستم وارد شده با نامه های موجود، جهت تعیین مناسبترین پوشه برای آن از معیار شباهت Dice [۱۳] استفاده شده است:

$$Sim(Query, Doc) = \frac{2\sum_{i=1}^{\#Terms} tfIdf(Query(i)) \bullet tfIdf(Doc(i))}{\sum_{i=1}^{\#QueryTerms} tfIdf(Query(i))^2 + \sum_{i=1}^{\#DocTerms} tfIdf(Doc(i))^2} \qquad (۳)$$

مقدار این معیار شباهت عددی بین ۰ و ۱ می باشد که ۱ مشخص کننده شباهت کامل و ۰ مشخص کننده عدم شباهت می باشد.

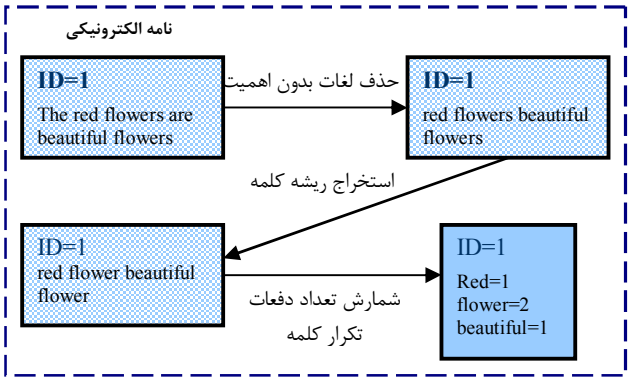
۳. الگوریتم پیشنهادی

۳-۱ پیش پردازش

اولین قدم برای پیاده سازی سیستم طبقه بندی نامه های الکترونیکی، خواندن نامه، تجزیه آن به لغات و اعمال پیش پردازشهایی بر روی آن می باشد. عملیات پیش پردازش عبارتند از:

- حذف لغاتی که فراوانی آنها در متن زیاد می باشد مانند حروف عطف (and, or, with,...)، حروف اضافه (at, as, in,...)، فعلهای ربطی(do, was, will,…)، ضمایر(you, we,…)، قیدهای زمان و مکان(most, almost, after,…)، حروف اشاره(that, this,…) و سایر کلماتی که در مشخص شدن محتوا و مضمون نامه تاثیری ندارند.
- دامنه استفاده از افعال در زمانهای مختلف (ماضی، مضارع، آینده) در یک متن بسیار زیاد می باشد. لذا بن فعل استخراج می شود و اگر اسمی بصورت جمع بکار برده شده باشد نیز، حالت مفرد آن درنظر گرفته می شود، در واقع از ریشه کلمات بعنوان واحدهای اطلاعاتی استفاده می شود.
- با حذف حروف عطف، حروف اضافه (stop word ها) و همچنین استفاده از بن فعل، همچنان متن مورد پردازش دارای واژه های زیادی می باشد، در نتیجه یک راه دیگر برای کاهش بیشتر ابعاد بردار ویژگی می تواند حذف واژه هایی باشد که مثلاً کمتر از سه حرف داشته باشند.

سپس هر نامه به مجموعه واژگانی تبدیل می شود که برای تمامی این واژه ها، تعداد دفعاتی را که در آن نامه تکرار شده اند، محاسبه می شود. برای هر نامه تعداد N واژه را که به طور تصادفی انتخاب شده اند را جهت ارائه آن نامه، ذخیره می کنیم؛ که در اینجا N=۵۰ در نظر گرفته شده است. فرآیند بیان شده در شکل۱ نشان داده شده است.



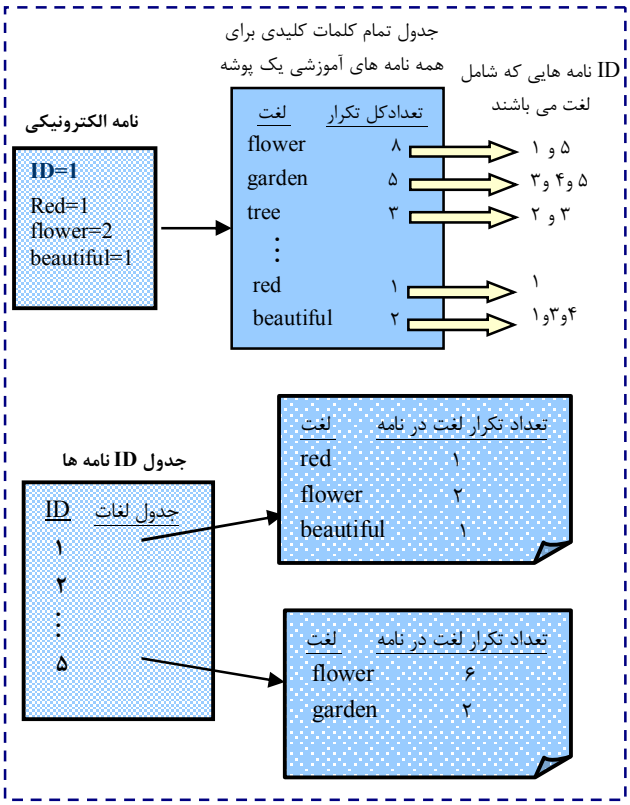
شکل(۱): حذف لغات بدون اهمیت، استخراج ریشه کلمه و شمارش تعداد دفعات تکرار لغات

۳-۲ ایجاد لیست معکوس[۶]

دومین گام، فهرست بندی هر یک از نامه ها به یک لیست معکوس می باشد که ساختار این لیست معکوس در شکل۲ نشان داده شده است. چگونگی ایجاد این لیست معکوس بدین ترتیب است که تعداد تکرار هر واژه در یک نامه شمرده شده و در یک جدول کلمات کلیدی سراسری قرار داده می شود. این جدول با هر واژه کلیدی، قابل دسترسی می باشد. برای هر واژه کلیدی، اشاره گری وجود دارد که شناسه تمام نامه هایی که شامل آن واژه کلیدی هستند را نشان می دهد. این اشاره گر از آن جهت مفید است که یک دسترسی آسان به تمام نامه هایی که شامل آن واژه خاص می باشند را فراهم می کند. در کنار جدول کلمات کلیدی سراسری، همچنین جدول شناسه نامه ها نیز وجود دارد. این جدول شامل تمام واژه های استخراج شده از نامه به همراه تعداد دفعات تکرار آن واژه ها در آن نامه می باشد. از این مقادیر و تعداد دفعات تکرار سراسری واژه ها برای ساختن بردار tf-idf استفاده می شود.

۴- نتایج پیاده سازی

در این مقاله از outlook برای تهیه نمونه های (نامه های) آموزشی و آزمایشی استفاده شده است. بدین ترتیب که در پوشه Inbox تعداد شش پوشه متفاوت (آگهی ها، نامه های سرگرم کننده، اخبار علمی، نامه های شخصی، اخبار سیاسی و اخبار ورزشی) ایجاد کرده و از سایتهای مختلف، نامه هایی را در زمینه های بیان شده، دریافت می کنیم. نامه هایی را که می خواهیم بعنوان داده آموزشی باشند را در داخل پوشه های مربوط به خود قرار داده و نامه هایی را که می خواهیم بعنوان داده آزمایشی از آنها استفاده شود را در پوشه اصلی inbox می گذاریم. هر کدام از پوشه ها دارای ۱۰ تا ۲۵ نامه آموزشی هستند و تعداد نامه های آزمایشی ۲۰۰ نامه است. تعداد لغات هر نامه بین ۲۰۰ تا ۵۰۰ لغت می باشد.



شکل(۲): ساختار لیست معکوس

یک بهینه سازی که می تواند جهت افزایش سرعت مقایسه واژگان بکار رود استفاده از یک جدول هش جهت تبدیل واژگان که بصورت رشته ای هستند به مقادیر ۳۲ بیتی می باشد [۱۲]. این عمل، موجب افزایش قابل ملاحظه ای در سرعت مقایسه و کم شدن حافظه مورد نیاز برای ذخیره واژگان می شود که در پیاده سازی انجام شده، مورد استفاده قرار گرفته است.

برای بدست آوردن میزان خطای طبقه بندی، درصد نامه هایی را به غلط دسته بندی شده اند را به تعداد کل نامه های آزمایشی بدست می آوریم که برای۳= k کمترین خطا (۰،۹۷ ٪) بدست آمده است. برای k های بزرگتر از ۵ نیز میزان خطا ثابت باقی مانده است. درصد خطا در جدول۱ آورده شده است.

جدول(۱): خطای طبقه بندی با kهای مختلف

مقدار k	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰
خطا	۴،۸۶	۲،۴۷	۰،۸۶	۴،۷۳	۵،۶۳	۵،۶۳	۵،۶۳	۵،۶۳	۵،۶۳	۵،۶۳

برای بررسی میزان پایداری سیستم ارائه شده، تعداد ۵۰ نامه نویزی را وارد سیستم کردیم. منظور از نامه نویزی، نامه های الکترونیکی هستند که تعدادی از واژه های آنها تغییر کرده یا پاراگرافهای آنها جابجا گردیده و یا پاراگرافی از نامه حذف و یا به آن اضافه گردیده است. عملکرد سیستم در برخورد با اینگونه نامه ها، عاری از خطا بود و با انتخاب k=۳ تمام این نامه ها را بدون خطا در پوشه های مربوط به خود دسته بندی نمود.

جهت بررسی عملکرد سیستم در مقابل تعداد نامه های آزمایشی زیاد، تعداد نامه های آزمیشی را از ۲۰۰ نامه به ۴۰۰ نامه افزایش داده و خطای دسته بندی را محاسبه نمودیم. کمترین اختلاف خطای بدست آمده برای ۳= k بدست آمد. خطای سیستم با دو برابر شدن تعداد نامه های آزمایشی، تنها ۰،۳۴ افزایش پیدا کرد و این نشان می دهد که سیستم پیشنهاد شده با نمونه های آموزشی کم، توانایی دسته بندی داده های زیاد با خطای دسته بندی جزئی را دارا می باشد.

در بکارگیری شبکه عصبی پرسپترون چند لایه و قانون یادگیری انتشار خطا به عقب برای دسته بندی پایگاه داده ای شامل ۴۲۳ نامه الکترونیکی و تعداد ۶ دسته مختلف، دقت ۶۸،۵۴٪ گزارش شده است[۱۴] که در مقایسه با روش پیشنهادی، خطای دسته بندی قابل ملاحظه ای را دارا می باشد.

۵- نتیجه گیری و کارهای آینده

در این مقاله سیستمی ارائه گردید که مبتنی بر ایجاد طبقه بندی کننده ها بر روی پوشه هایی که توسط کاربر ایجاد شده اند، می باشد. هر طبقه بندی کننده با اسکن کردن تمام نامه هایی که کاربر درون هر پوشه قرار داده (نامه های آموزشی) ایجاد می شود. از هر نامه ویژگیهایی استخراج شده و به طبقه بندی کننده اضافه می شود. این سیستم توانایی استخراج ویژگی از موضوع و بدنه اصلی نامه را دارا می باشد. این ویژگیها، همان واژه های نامه بوده که از بین آنها واژه های کم اهمیت و بعبارتی، واژه هایی که در دسته بندی نامه بدون تأثیر می باشند (مانند حروف عطف، حروف اضافه، قیود زمان و مکان و ...) و همچنین واژه هایی که دارای کمتر از سه حرف هستند، حذف می گردند. سپس به هر یک از این واژه ها بر اساس تعداد دفعاتی که در نامه بکار رفته اند، وزنی داده می شود.

طبقه بندی کننده بکار رفته در این سیستم، طبقه بندی کننده ساده نزدیکترین همسایه (k-NN) می باشد. علت انتخاب این طبقه بندی کننده، سادگی در پیاده سازی، سرعت و راحتی در آموزش آن و همچنین انجام دسته بندی بصورت محلی و عدم نیاز به نمونه های منفی است.

با توجه به تنوع موضوعات مورد علاقه هر کاربر، داده های آزمایشی زیادی وجود دارد. در تهیه داده های آزمایشی، سعی شده است که نامه های متنوعی برای سلیقه های متفاوت و از سایتهای اینترنتی متعددی جمع آوری شود. کمترین خطای بدست آمده جهت دسته بندی ۲۰۰ نامه الکترونیکی، ۰،۹۷ برای k=۳ بوده است.

از جمله کارهایی که می توان جهت بهبود روش، بکار گرفته شود، استفاده از طبقه بندی کننده های بهتر و کارآمدتر بگونه ای که توانایی دسته بندی نامه ها در چند پوشه مختلف را داشته باشند و همچنین کار کردن بر روی نامه هایی که فقط متنی نباشند را نام برد.

منابع

[1] Boone, G. Concept Features in Re: Agent, an Intelligent Email Agent. *Proceedings of the Second International Conference on Autonomous Agents*. Minneapolis/St. Paul, May 10-13, 1998.
[2] I. Androutsopoulos et al. "Learning to Filter Spam E-mail: A Comp. of a NB and Memory-Based Approach", *PKDD, 2000.*
[3] Cohen, W. Learning rules that classify e-mail. *Proceedings of the 1996 AAAI Spring Symposium on Machine Learning in Information Access*. 1996.
[4] Sahami, M., Dumais, S., Heckerman, D., Horvitz, E.A Bayesian approach to filtering junk e-mail. *AAAI'98 Workshop on Learning for Text Categorization*, Madison, WI, July 1998.
[5] Dumais, S., Platt, J., Heckerman, D., Sahami, M. Inductive learning algorithms and representations for text categorization. *Proceedings of ACM-CIKM98*, November, 1998.
[6] Apte, C., Damerau, F., & Weiss, S. Automated learning of decision rules for text categorization. *ACM Transactions on Information Systems*, 12, 233-251. 1994.
[7] J. Rennie, "An Application of Machine Learning to E-Mail Filtering", *KDD-2000 Text Mining Workshop*, 2000.
[8] M. Sahami, S. Dumais, et al. "A Bayesian Approach to Filtering Junk E-Mail", *AAAI Workshop Learning for TC*, 1998.
[9] Mock, K & Vemuri, V. Information Filtering via Hybrid Techniques. *Information Processing and Management*, Permagon Press, v33, n5, 1997, pp 633-644.
[10] W. Cohen, "Learning Rules that Classify E-Mail", *AAAI Symp. on Machine Learning in Inf. Access*, 1996, pp.18-25.
[11] Mock, K., Adams, R., Spangler, L. Venice: Content-Based Information Management for Electronic Mail. *Intel Software Developers Conference*, Portland, Oregon. 1997.
[12] Jenkins, B. Algorithm Alley: Hash Functions. Dr. Dobbs 22, 9, September 1997.
[13] Rasmussen, Edie (1992). "Clustering Algorithms". Information Retrieval: Data Structures & Algorithms. William B. Frakes and Ricardo Baeza-Yates (Eds.), Prentice Hall, 1992.
[14] James Clark, Irena Koprinska, and Josiah Poon, "A Neural Network Based Approach to Automated E-mail Classification", Proceedings of the IEEE/WIC International Conference on Web Intelligence (WI'03), 2003 IEEE

زیرنویس‌ها

[1] spam
[2] Instance-based learning
[3] Text Classification
[4] Term frequency-inverse document frequency
[5] dot product
[6] inverted list