عنوان زیرپروژه:

امکان‌سنجی سیستم طبقه‌بندی متون برای زبان فارسی

فهرست مطالب

عنوان **شماره صفحه**

عنوان **شماره صفحه**

عنوان **شماره صفحه**

1. مقدمه

در زمینه‌ی پردازش اطلاعات، سیستم‌های بسیاری ایجاد شده است. این سیستم‌ها در پنج گروه عمده، دسته‌بندی می‌شوند [1]:

1. سیستم‌های اطلاعات مدیریتی[۱]
2. سیستم‌های مدیریت پایگاه داده‌ها[۲]
3. سیستم‌های تصمیم‌یار[۳]
4. سیستم‌های پرسش و پاسخ[٤]
5. سیستم‌های بازیابی اطلاعات[٥]

البته باید ذکر کرد که با نگرشی دیگر موارد 3 و 4 می‌توانند خود بخشی از "بازیابی اطلاعات" باشند. باید ذکر کرد که در سراسر این نوشتار منظور از پردازش اطلاعات، بازیابی اطلاعات می باشد.

در این فصل مقدمات لازم برای دسته‌بندی متون در دنیای امروز و چند مثال از زمینه‌های کاربردی آن آورده شده است. مسأله دسته‌بندی متون در مقایسه با دسته‌بندی سلسله‌مراتبی متون و دیگر مسائل کلی نیز مورد بحث قرار گرفته است.

1-1. سیستم‌های طبقه‌بندی متون و اهمیت آن

در 10 سال اخیر مدیریت مبتنی بر محتوای متون (تحت عنوان کلی بازیابی اطلاعات شناخته می‌شوند) به علت رشد سریع و در دسترس قرار گرفتن متون به شکل دیجیتالی، از اهمیتی

---

[۱] Management Information Systems

[۲] Database Systems

[۳] Decision Support Systems

[٤] Question-Answering System

[٥] Information Retrival

دوچندان برخوردار شده است. دسته‌بندی متون (کلاس‌بندی متون) به عمل برچسب‌گذاری موضوعی متون زبان طبیعی بر مبنای یک مجموعه از پیش تعیین شده، یکی از این موارد است. هم اکنون دسته‌بندی متون در بسیاری از زمینه‌ها از شاخص‌گذاری متون بر مبنای یک لغت‌نامه کنترل شده[1] تا فیلتر کردن متون، تولید خودکار فراداده، ابهام‌زدایی از کلمه[2]، تولید کاتالوگ‌های سلسله‌مراتبی از منابع وبی[3]، و به طور کلی در هر کاربردی که نیاز به سازماندهی مستندات یا توزیع انتخابی و تطبیقی خاصی از مستندات مد نظر باشد، کاربرد دارد.

اگرچه این مسأله از دهه 1960 میلادی به این سو مورد مطالعه بسیاری قرار گرفته است[2] ولی از دهه 90 به بعد به لطف پیشرفت‌های نرم‌افزاری و سخت‌افزاری، آن را به یک مبحث جدی بدل نموده است. در حقیقت، سیستم‌های دسته‌بندی متون از مهندسی دانش (مجموعه‌ای از قوانین گردآوری شده توسط فرد خبره) به سمت مدل‌های آماری (به خصوص در زمینه‌های تحقیقاتی) در حال حرکت بوده است. [3] در تکنیک‌ها یادگیری ماشین، با استفاده از یک پروسه استنتاجی کلی، به طور خودکار دسته‌بندها با استفاده از یادگیری از یک مجموعه‌ی مستندات از پیش دسته‌بندی شده، مشخصات دسته‌ی مورد نظر را می‌سازند. از آن‌جا که برای ساختن دسته‌بندها نیازی به دانش مهندسی یا افراد خبره آن زمینه ندارد، مزایای این رهیافت، یک دقت قابل مقایسه در مقابل دقت به دست آمده از فرد خبره، و کاهش قابل توجه در هزینه انسانی می‌باشد. دسته‌بندی متون به صورت دستی علاوه بر زمان‌بری و هزینه‌ی زیاد، معایب ذیل را نیز با خود به همراه دارد: [4]

۱. برای زمینه‌های تخصصی خاص نیاز به دانش افراد خبره دارد (مانند بانک‌های پزشکی، بانک‌های حقوقی)

۲. از آن‌جا که برچسب‌گذاری دستی مبتنی بر دانش و تجربه فرد می‌باشد، بسیار خطا پذیر است.

۳. تصمیم دو فرد خبره در برچسب‌گذاری می‌تواند متفاوت و حتی ناسازگار باشد (سیستم ناسازگاری درونی دارد) [5]

---

[1] Controlled Dictionary

[2] Word Sense Disambiguation

[3] Population Of Hierarchical Catalogues Of Web Resources

امروزه بنابر آنچه گفته شد، دسته‌بندی متون در تقاطع یادگیری ماشین و بازیابی اطلاعات مطرح می‌باشد. هم‌چنین تعدادی از مشخصات این مسأله با مسائلی چون استخراج اطلاعات و دانش از متون، و داده‌کاوی متون[1] مشترک می‌باشد [6][7]. با این حال در ارتباط با مرز و تعریف دقیق آن‌ها کماکان مورد بحث می‌باشد. "داده‌کاوی متون" در وظایفی که با تحلیل مقدار زیادی از متون و یافتن کاربرد الگوها، سعی می‌کند تا به استخراج احتمالی اطلاعات (تنها با استفاده از اطلاعات احتمالی) بپردازد. امروزه، این مسأله به طور روز افزون مورد استفاده قرار گرفته شده است. بر طبق این دیدگاه، دسته‌بندی متون یکی از نمودهای داده‌کاوی متون می‌باشد. مفاهیم دسته‌بندی متون هم اکنون از یک ادبیات عمیقی برخوردار گشته است، اما این مفاهیم عمدتاً پراکنده بوده است. این مسأله در دو مقاله به تفضیل مورد بحث قرار گرفته است [8][9]. با توجه به بررسی‌های انجام شده تا زمان این گزارش، کتاب مشخصی در مورد دسته‌بندی متون به طور خاص مشاهده نشده و تنها در فصل 16 کتاب [10] و فصل 2 و 3 کتاب [11] به این مسأله پرداخته‌اند. باید متذکر شد، گه‌گاه "دسته‌بندی خودکار متون" در مقاله‌هایی مورد استفاده قرار می‌گیرد که با آنچه در این‌جا مورد نظر است کاملاً متفاوت می‌باشد. از یک طرف 1- تعین خودکار متون به دسته‌های از پیش تعریف شده، که در این‌جامد نظر است تا 2- تشخیص خودکار مجموعه‌ای از دسته‌ها (برای مثال [12]) یا 3- تشخیص خودکار مجموعه‌ای از دسته‌ها و گروه‌بندی متون بر مبنای آن (برای مثال [13]) که معمولاً خوشه‌بندی[2] متون نامیده می‌شود، یا 4- هر کاری که برای قرار دادن متون در گروه‌های خاصی باشد [10] قابل گسترش است.

2-1. کاربردهای دسته‌بندی متون

از کاربردهای دسته‌بندی متون می‌توان به : سیستم‌های اتوماتیک پاسخ به سوالات [14]، فیلتر کردن اطلاعات، تشخیص موضوعیت داده‌ها، نامه‌های الکترونیکی بی‌ارزش، تشخیص عنوان و دیگر زمینه‌های مرتبط اشاره نمود[15].

[1] Text mining

[2] Clustering

1-3. فعالیتها در زمینه‌ی زبان فارسی

باتوجه به بررسی‌های انجام شده توسط نگارنده، اگرچه تاکنون تحقیقی برای خوشه‌بندی متون فارسی [16]، گرامر محاسباتی [17]، رفع ابهام از کلمات فارسی [18] صورت گرفته است. ولی از منظر دسته‌بندی متون، اگرچه تحقیقاتی برای دیگر زبان‌ها (بالاخص زبان انگلیسی) انجام شده است. اما دسته‌بندی متون برای زبان فارسی تا تکمیل این نوشتار مشاهده نشده است.

1-4. تعریف‌ها

1-4-1. گنج واژه

گنج‌واژه[1] تنظیم و مرتب کردن واژه‌ها و عبارت‌های زبان نه بر حسب الفبا، بلکه بر حسب مفاهیمی که بیان می‌کنند. فرهنگ مفاهیم با فرهنگ لغات متفاوت است.[2]

1-4-2. پیکره‌ی زبانی

پیکره‌ی زبانی[3] مجموعه‌ای از نوشتار در یک زبان که ویژگی‌های زبان را به توان با استفاده از آن بازنمایی نمود.

1-4-3. واژگان

[1] Thesaurus

[2] چگنی، ابراهیم "فرهنگ توصیفی آموزش زبان و زبان‌شناسی کاربردی" انتشارات رهنما، بهار 1384، ص. 563.

[3] Corpus

واژگان[1] مجموعه‌ای از لغات و کلیه مشتقات آن که در بعضی اوقات قوانین تولید واژه را نیز شامل می‌شود.

1-4-4. مجموعه‌ی آموزشی

مجموعه اطلاعاتی که برای آموزش الگوریتم‌های با ناظر[2] استفاده می‌گردند.

1-4-5. مجموعه‌ی آزمایشی

مجموعه اطلاعاتی که برای آزمایش الگوریتم‌های با ناظر و یا بدون ناظر استفاده می‌گردند.

1-4-6. کلمات بی‌ارزش

کلماتی که هیچ گونه ارزش معنایی و یا مفهومی از نقطه نظر دسته‌بندی ندارند.

1-4-7. عبارت

دو یا چند کلمه که ترکیب مشخصی از آن‌ها مفهوم خاصی را منتقل می‌کند. برای مثال "سازمان" + "ملل" و از طرف دیگر "سازمان ملل".

1-4-8. کلمه کلیدی

کلمات خاصی که یک متن می‌تواند بر مبنای آن اندیس‌گذاری گردد. این کلمات یا عبارات به نوعی متن را دسته‌بندی می‌کنند.

---

[1] Lexicon

[2] Supervised Learning

2. دسته‌بندی متون

2-1. تعریف دسته‌بندی متون

در صورتی که مجموعه‌ای از متون $D = \{(d_1, y_1), \ldots, (d_i, y_i), \ldots, (d_n, y_n)\}$ داشته باشیم به طوری که n تعداد متون و $d_i = [w_{i,1}, \ldots, w_{i,k}, \ldots, w_{i,|di|}]$ متن iام این مجموعه باشد، $w_{i,k}$ کلمه kام متن iام باشد و y_i به دسته‌ای که متن به آن متعلق است(یعنی $y_i \in C$ به به طوری که $C=\{c_1, c_2, \ldots, c_{|C|}\}$ مجموعه دسته‌های از پیش تعریف شده در سیستم باشد) اشاره کند. هدف در دسته‌بندی متون، استنتاج یک تابع رابطه‌ای f است به نحوی که $y_i = f(d_i)$ باشد. یا به صورت کامل‌تر دسته‌بندی متون تعیین یک مقدار بولی[1] برای هر جفت $<d_j, c_i> \in D*C$، در جایی که D مجموعه‌ای از متون و C مجموعه دسته‌های از پیش تعیین شده می‌باشد. مقدار T تعیین می‌کند که متن d_j به دسته c_i متعلق است و مقدار F نیز عدم تعلق متن d_j به c_i را نشان می‌دهد. هدف در این‌جا به دست آوردن تخمین تابع $\Phi: D*C \rightarrow \{T, F\}$ می‌باشد. از این‌جا به بعد فرض می‌شود:

- دسته‌ها تنها برچسب‌های سمبولیک هستند و هیچ دانش اضافی(به لحاظ اجرایی یا تعریفی) با خود به همراه ندارند.
- *دانش از بیرون*[2] (برای مثال اطلاعاتی که به منظور دسته‌بندی از منبع خارجی) موجود نباشد. بنابراین دسته‌بندی می‌باید تنها بر اساس *دانش از درون*[3] (دانشی که از خود متن به دست می‌آید) انجام گیرد. به عبارت دیگر اطلاعات دیگر همچون نویسنده، تاریخ انتشار در دسترس نباشد.

متدهای دسته‌بندی متون که ما در مورد آن‌ها بحث خواهیم کرد کاملاً کلی بوده و برای زمینه خاصی نمی‌باشد. در حقیقت این پیش فرض‌ها اگرچه هزینه‌های دسته‌بندی را افزایش می‌دهد ولی

---

[1] Boolean

[2] Exogenous Knowledge

[3] Endogenous Knowledge

برای قانونی‌بودن عملیات دسته‌بندی متون اجباری است [19][20]. دسته‌بندی بر مبنـای دانـش از درون یعنی یک متن تنها بر اساس اطلاعات معنایی‌اش دسته‌بندی می‌شود.

2-2. دسته‌بندی تک برچسبی در مقابل چند برچسبی

ممکن است بسته به کاربرد، شروط متفاوتی برای مسأله دسته‌بندی متون در نظر گرفته شود. برای نمونه، ممکن است هر کدام از d_jها برای یک عدد صحیح k دسته (دقیقاً k یا بیش‌تر از k یا کم‌تر از k) از مجموعه دسته‌های C متعلق باشد. مسأله دسته‌بندی متون از حالتی که هر متن تنها به یـک دسته متعلق است، اصطلاحاً تک‌برچسبی (دسته‌ها با یکدیگر هم‌پوشانی ندارنـد) تـا وقتـی کـه بـه تعدادی از دسته‌ها (از 0 تا |C|) متلق باشند که اصطلاحاً چندبرچسبی (دسته‌ها هم‌پوشـانی دارنـد) گفته می‌شود، تعریف می‌گردند. یک روش خاص برای دسته‌بندی تک‌برچسبی، دسته‌بندی دودویی است که $d_j \in D$ باید به دسته c_i یا $\bar{c}_i$ متعلق باشد.

از نقطه نظر تئوریکی، از آن‌جا که هر الگوریتمی که برای دسته‌بندی دودیی اسـت مـی‌توانـد بـرای دسته‌بندی چند برچسبی به کار برده شود، حالت دودویی از حالت چند برچسبی عمومی‌تـر اسـت. تنها کافی است تا مسأله دسته‌بندی چندبرچسـبی $\{c_1, c_2, \ldots, c_{|C|}\}$ بـه |C| مسـأله تـک‌برچسـبی مستقل از دسته‌بندهای دودویی $\{c_i, \bar{c}_i\}$ کـه $i = 1, 2, \ldots, |C|$ تبـدیل شـود. اگرچـه ایـن مسـأله نیازمند این است که برای هر 'c و "c، دسته‌ها مستقل از یکـدیگر باشـند. بعبـارت دیگـر $\Phi(d_j, c')$ ارتباطی با $\Phi(d_j, c'')$ نداشته باشد و بلعکس. از طرفی باید توجه داشت که نمی‌توان نتیجـه گرفـت که الگوریتمی که برای دسته‌بندی چند برچسبی می‌باشد را نمی‌توان برای دسته‌بندی تک‌برچسبی یا دودویی استفاده کرد. در حقیقت، برای یک متن داده شده d_j برای دسته‌بندی، 1- دسته‌بند باید بتواند متن را به (k>1) دسته، دسته‌بندی کند که شاید چگونه دسته‌های مناسب را انتخاب می‌کند، مشخص نباشد. 2- دسته‌بند باید بتواند متن را به هیچکدام از دسته اعمال ننماید که شاید چگونـه دسته‌های غیر مناسب را پیدا می‌کند، مشخص نباشد.

باید ذکر نمود که در ادامه این نوشتار تا وقتی که به طور صریح ذکر نگردیـد، منظـور دسـته‌بنـدی دودویی می‌باشد. دلایل زیادی برای این امر می‌باشد از جمله:

- حالت دودویی به خودی خود بسیار مهم می‌باشد. زیراکه کاربردهای مهم دسته‌بندی متون، شامل فیلتر کردن جزو این دسته می‌باشد (برای مثال آیا این متن، یک خبر در ارتباط با موسیقی است یا نه؟!). در دسته‌بندی متون تعداد ویژگی‌هایی که نشان می‌دهند این خبر در ارتباط با موسیقی است از ویژگیهایی که نشان می‌دهند در ارتباط با موسیقی نیست بیش‌تر است (تعداد خصوصیاتی که نشان می‌دهند متن مربوط به موسیقی هست از تعداد خصوصیاتی که نشان می‌دهند متن مربوط به موسیقی نیست، بیش‌تر است).
- حل مسأله حالت دودیی همانند حل مسأله چند برچسبی است، که بازنمایی آن برای کاربردهای دسته‌بندی متون بسیار حائز اهمیت است. (شامل شاخص‌گذاری خودکار برای سیستم‌های دودویی)
- اغلب دسته‌بندی‌های متون را می‌توان در بستر دودویی ارزیابی کرد.
- اغلب تکنیک‌های دسته‌بندی دودویی تنها حالت خاص متد موجود برای دسته‌بندی تک برچسبی می‌باشند که توضیح آن‌ها ساده‌تر است.

در نهایت، در این‌جا منظور اینست که مسأله دسته‌بندی تحت $C = \{c_1, c_2, \ldots, c_{|C|}\}$ شامل $|C|$ مسأله مستقل از دسته‌بندی c_i دیده می‌شود، که $i = 1, 2, \ldots, |C|$ است. یک دسته‌بند برای دسته c_i، $\Phi_i: D \rightarrow \{T, F\}$ می‌باشد.

3-2. دسته‌بندی متون مبتنی بر دسته در مقابل مبتنی بر متن

دو روش متفاوت برای استفاده از دسته‌بندها وجود دارد. برای $d_j \in D$، ما باید تمام $c_i \in C$ را که باید انتخاب شوند را بیابیم (دسته‌بندی مبتنی بر متن[1])؛ از طرف دیگر برای $c_i \in C$ باید $d_j \in D$ را بیابیم که باید جزء آن باشند (دسته‌بندی مبتنی بر دسته[2]). این اختلاف بیش‌تر به لحاظ فلسفی است تا

[1] Document-Pivoted Categorization(DPC)

[2] Category-Pivoted Categorization(CPC)

مفهومی، اما از آن‌جا که ممکن است مجموعه‌های C و D از ابتدا موجود نباشند، این مسأله اهمیت پیدا می‌کند. هم‌چنین بسته به متدی که برای ساختن دسته‌بندها انتخاب می‌شود، ممکن است یکی از این دو امکان پذیر نباشد. DPC زمانی مناسب است که متن‌ها در طول زمان عرضه شوند. (برای مثال در فیلتر کردن پست‌های الکترونیکی) در مقابل CPC زمانی مناسب است که:

1- یک دسته جدید $c_{|C|+1}$ به مجموعه دسته‌های موجود $C = \{c_1, c_2, \ldots, c_{|C|}\}$ که قبلاً توسط تعدادی از متون تحت C دسته‌بندی شده بودند، اضافه شود.

2- این متون نیاز به بازنگری برای دسته‌بندی تحت $c_{|C|+1}$ دارند (برای مثال [21]). از آن‌جا که شرایط DPC از CPC متداول‌تر است، اولی بیش از دومی مورد استفاده قرار می‌گیرد. هم‌چنین بعضی از تکنیک‌های مشخص تنها برای یکی از این دو قابل اعمال می‌باشد. (برای مثال روش حد آستانه متناسب[1] تنها برای CPC قابل اعمال است) این مسأله یک استثنا است تا یک قانون.

2-4. دسته‌بندی قطعی در مقابل دسته‌بندی رتبه ای

از آن‌جا که یک سیستم دسته‌بندی متون نیازمند تصمیم T یا F برای هر جفت $\langle d_j, c_i \rangle$ می‌باشد، در حالی که در بعضی از شرایط تصمیم‌گیری می‌باید به صورت رتبه‌بندی صورت پذیرد.

برای مثال برای $d_j \in D$، یک سیستم با توانایی استفاده از رتبه‌دهی دسته‌های $C = \{c_1, c_2, \ldots, c_{|C|}\}$ بر مبنای تخمین تناسب آن‌ها با d_j، بدون هیچ گونه تصمیم قطعی بر روی هیچکدام آن‌ها عمل کند را دارد. از آن‌جا فرد خبره می‌تواند انتخاب دسته (یا دسته‌ها) را به بالای لیست به جای کل لیست محدود کند، یک چنین لیست رتبه‌داده‌شده‌ای کمک بزرگی برای تصمیم گیری نهایی او برای دسته‌بندی می‌باشد. از طرف دیگر برای $c_i \in C$، سیستم توانایی رتبه‌دهی ساده متون در D را بر مبنای تناسب c_i دارد؛ یعنی برای دسته‌بندی تحت c_i، کافی است فرد خبره متونِ با رتبه‌ی بالا را به جای کل متون مجموعه آزمایش کند . بعضی اوقات اصطلاحاً به این دو، دسته‌بندی متون با رتبه‌دهی دسته[2] و دسته‌بندی متون با رتبه‌دهی متون[1] گفته می‌شود [22].

[1] Proportional Thresholding Method

[2] Category-Ranking Text Categorization

در کاربردهای خاص که اثر یک سیستم کاملاً خودکار از یک سیستم فرد خبره به طور محسوسی پایین‌تر است، سیستم‌های دسته‌بندی تقابلی[23] و نیمه اتوماتیک بسیار مفید می‌باشند. این مسأله زمانی بیش‌تر نمایان می‌گردد که 1. کیفیت مجموعه‌ی آموزشی[۲] پایین است یا 2. وقتی که متون آموزشی بخوبی متون دیده نشده را که بعداً به سیستم وارد می‌شود بازنمایی نکنند.

در ادامه هر جا نوع الگوریتم ذکر نگردید، منظور الگوریتم‌های دسته‌بندی قطعی[۳] است. باید ذکر کرد که بیش‌تر الگوریتم‌هایی که بحث می‌شوند بالقوه می‌توانند برای دسته‌بندی رتبه‌ای[۴] متون نیز به کار روند.

---

[۱] Document-Ranking Text Categorization

[۲] Training Set

[۳] Hard Categorization

[۴] Ranking Categorization

3. کاربردهای دسته‌بندی متون

مسأله‌ی دسته‌بندی متون به کار آقای مارون بر روی دسته‌بندی متون به صورت احتمالی باز می‌گردد [24]. از آن پس برای کاربردهای متنوعی (که به طور خلاصه مهم‌ترین‌هایشان مرور خواهد شد) استفاده شده است. باید توجه داشت، از آن‌جا که بعضی از این گروه‌ها با هم هم‌پوشانی دارند، مرزهای بین این گروه‌ها دقیق نیست و بعضی، بعضی دیگر را پوشش می‌دهند. دیگر کاربردها همچون: دسته‌بندی گفتاری[۱] که ترکیبی از دسته‌بندی متون و تشخیص گفتار[۲] است [25][26]، دسته‌بندی متون چند رسآن‌های[۳] از طریق عنوآن‌های متنی [27]، تشخیص نویسنده برای متون ادبیاتی نامشخص یا مورد بحث [28] ، تشخیص زبان برای متونی که زبان آن‌ها نامشخص است [29] تشخیص خودکار جنس متن[۴] [30] ، و رتبه‌بندی خودکار کیفیت نوشتار[۵] [31] در این‌جا مد نظر نیست.

3-1. شاخص‌بندی برای سیستم‌های بازیابی اطلاعات بولی

بیش‌تر کاربردهای متداول در این زمینه‌ی تحقیقاتی برای شاخص‌بندی خودکار متون برای سیستم‌های بازیابی اطلاعات با تکیه بر یک لغت‌نامه کنترل شده بوده که مثال برجسته آن سیستم‌های بولی هستند [12][24][32][33][34]. در دسته اخیر، برای هر متن یک یا چند کلید واژه یا عبارت کلیدی که محتوای آن را توصیف می‌کند، تعیین می‌گردد. این کلید واژه‌ها و عبارات کلیدی متعلق به مجموعه محدودی[۶] است که اغلب شامل یک گنج واژه سلسله‌مراتبی موضوعی[۱]

---

[۱] Speech Categorization

[۲] Speech Recognition

[۳] Multimedia Document Categorization

[۴] Automated Identification Of Text Genre

[۵] Automated Essay Grading

[۶] به این مجموعه لغت‌نامه کنترل شده نیز گفته می شود.

می‌باشند (برای نمونه گنج‌واژه‌ی ناسا[۲] برای مرتب‌سازی مسائل فضایی، یا گنج واژه مش[۳] برای پزشکی). معمولاً این معین‌سازی توسط افراد شاخص‌بند خبره صورت می‌گیرد که فعالیتی بسیار پرهزینه می‌باشد. اگر ورودی‌ها در لغت‌نامه‌های کنترل شده به صورت دسته‌ها دیده شوند، آن‌گاه شاخص‌بندی متون را می‌توان یکی از مثال‌های دسته‌بندی متون دانست که توسط یکی از روش‌های توضیح داده شده در این نوشتار قابل حل می‌باشد. همان‌طور که بعداً توضیح داده خواهد شد، باید توجه داشت که در این کاربرد نیاز به $k_1 \leq x \leq k_2$ کلید واژه که برای هر متن تعیین شده باشند، می‌باشد (برای k_1 و k_2 داده شده). احتمالاً، از آن‌جا که ممکن است متون جدید در هنگام ورود دسته‌بندی شوند، دسته‌بندی متون مبتنی بر متن یکی از بهترین گزینه‌ها خواهد بود. به طور مشخص، دسته‌بندهای مختلفی برای شاخص‌بندی متون معرفی شده است [35][36][37] .

شاخص‌بندی خودکار با لغت‌نامه کنترل‌شده در ارتباط نزدیکی با تولید خودکار فراداده[۴] می‌باشد. معمولاً در کتابخانه‌های دیجیتال از منظرهای مختلف به برچسب‌گذاری متون به وسیله‌ی فراداده‌هایی که آن‌ها را توصیف می‌کنند، تمایل دارند (برای مثال تاریخ ایجاد، نوع متن یا قالب، موجود بودن و غیره). بعضی از این فراداده‌ها موضوعی هستند که نقش‌شان شرح معنایی متون با استفاده از کلید واژه‌ها یا عبارات کلیدی است. تولید این فراداده‌ها ممکن است به عنوان یک مسأله در شاخص‌بندی متون با لغت‌نامه کنترل شده دیده شود که می‌توان با استفاده از تکنیک‌های دسته‌بندی متون آن‌ها را رفع نمود.

3-2. سازمان‌دهی متون

شاخص‌بندی با یک لغت‌نامه کنترل شده یک مثالی از مسأله‌ی کلی‌تر سازماندهی مبتنی بر متن است. در حالت کلی، بحث‌های زیادی در ارتباط با سازماندهی متون و بایگانی برای مقاصد شخصی، سازمانی، یا ساختارهای حقوقی مبتنی بر متن از طریق تکنیک‌های دسته‌بندی متون انجام شده

---

[۱] Thematic Hierarchical Thesaurus

[۲] NASA Thesaurus

[۳] MESH Thesaurus

[۴] Automated Metadata Generation

است. برای مثال، در ورودی تحریریه یک روزنامه، تبلیغات و آگهی‌ها برای انتشار باید تحت گروه‌های خدمات عمومی، فروش، خرید، درخواست کار دسته‌بندی شده باشند. عموماً روزنامه‌ها با حجم زیادی از آگهی‌های دسته‌بندی شده سر و کار دارند. این مهم می‌تواند توسط سیستم‌های خودکاری که دسته‌های مناسب را برای هر کدام از آگهی‌ها انتخاب می‌کند، انجام یابد. دیگر کاربردهای ممکن می‌تواند سازمان‌دهی اختراعات [21] باشد که جستجوی آن‌ها را آسان‌تر می‌کند، هم‌چنین در این‌جا به بایگانی خودکار مقاله روزنامه‌ها تحت عناوین مختلف یا گروه‌بندی مقاله‌های کنفرانس برای نشست‌ها نیز اشاره کرد.

3-3.فیلتر کردن متون

فیلتر کردن متن یکی از فعالیت‌های دسته‌بندی است که یک رشته از متن‌های ورودی که توسط یک تولیدکننده اطلاعات برای یک مصرف‌کننده اطلاعات به صورت غیر همزمان توزیع می‌شود، می‌باشد [38]. یک نمونه تلکس‌های خبری[1] هستند که تولیدکننده خبر که یک خبرگزاری است برای یک مصرف‌کننده خبر که یک روزنامه است، می‌فرستد [39]. در این حالت، سیستم فیلتر باید از دریافت خبرهایی که برای گیرنده جذاب نیستند، جلوگیری کند (برای مثال در یک روزنامه‌ی ورزشی تمام اخبار ورزشی هستند). فیلتر کردن می‌تواند به شکل یک دسته‌بندی متون تک‌برچسبی دیده شود که دسته‌بندی متون وارد شده به دو دسته‌ی مجزا (مرتبط و غیر مرتبط) تقسیم می‌شوند. بعلاوه، یک سیستم فیلتر، ممکن است متن‌ها را به دسته‌های موضوعی مصرف‌کننده نیز دسته‌بندی کند؛ در مثال بالا، کلیه‌ی مقاله‌ها درباره ورزش باید باشند و بر طبق ورزشی که مرتبط با آن هستند، دسته‌بندی شوند به نحوی که روزنامه‌نگارانی که متخصص ورزشی هستند تنها به همان دسته دسترسی یابند. به طور مشابه ممکن است یک فیلتر پست الکترونیکی برای حذف هرزنامه‌ها آموزش داده شود و دیگر نامه‌های وارده به دسته‌های جذاب برای کاربر دسته‌بندی شوند [40][41].

یک سیستم فیلتر ممکن است در سمت تولیدکننده باشد، به نحوی که متون را بر مبنای علاقمندی‌های مصرف کنندگان توزیع کند و یا ممکن است در سمت مصرف‌کننده باشد و از

[1] Newsfeed

دریافت مطالب بی‌ربط به مصرف‌کننده جلوگیری کند. در شکل اول، سیستم برای هر مصرف‌کننده یک نمایه[1] منحصر بفرد تولید می‌کند و آنرا مرتباً به‌روز می‌نماید [42]. در حالی که در حالت دوم (که متداول‌تر است و مورد نظر ما در این بخش نیز می‌باشد) تنها یک نمایه، مورد نیاز است. نمایه ممکن است توسط کاربر مقداردهی اولیه شده باشد و با استفاده از اطلاعات بازخوردی از کاربر (به طور صریح یا غیر صریح)، توسط سیستم ارتباط یا عدم ارتباط پیغام‌های رسیده در نمایه بروز گردد. در TREC community [43]، این مسأله، فیلتر تطبیق‌پذیر نامیده شده است. در حالتی که نمایه مشخصی وجود نداشته باشد، فیلتر دسته‌ای[2] نامیده می‌شود. فیلتر دسته‌ای هم‌ارز با دسته‌بندی تک‌برچسبی تحت $|C| = 2$ می‌باشد. با توجه به اینکه در این‌جا بررسی دسته‌بندی متون در حالت کاملاً کلی می‌باشد، برخی از نویسندگان [44][45][46][47] از عبارت فیلتر کردن به جای دسته‌بندی استفاده نموده‌اند.

فیلتر کردن متون به دهه 60 میلادی (یعنی زمانیکه سیستم‌های مختلف اتوماسیون و حالت‌های کار با چند مصرف‌کننده مورد بحث بود.) بر می‌گردد که پخش اطلاعات نامیده می‌شدند [48] . انفجار اطلاعات دیجیتال، اهمیت این گونه سیستم‌ها را افزایش داده است. تا جایی که امروزه زمینه‌هایی همچون روزنامه‌های شخصی وبی[3] ، بلوکه کردن هرزنامه‌ها[4]، انتخاب خبرهای یوزنت[5] در دامنه کاربردهای دسته‌بندی متون هستند. فیلترینگ اطلاعات با استفاده از تکنیک‌های یادگیری ماشین در مقالات متعددی مورد بحث قرار گرفته است [49][50][51][52][53].

3-4. رفع ابهام از کلمه

رفع ابهام از کلمه[6]، به یافتن درست کلمات هم‌نویس در یک متن می‌باشد. برای مثال "مرد" می‌تواند در معنای اسمی به انسان ذکور بالغ گفته شود و یا در معنای فعلی بن ماضی سوم شخص

---

[1] Profile

[2] Batch Filtering

[3] Personalized Web Newspaper

[4] Junk E-Mail Blocking

[5] Usenet

[6] Word Sense Disambiguation (WSD)

ساده از مصدر "مردن" باشد. بنابراین یکی از وظایف رفع ابهام انتخاب یکی از این دو حالت برای این کاربرد در جمله می‌باشد. رفع ابهام برای بسیاری از کاربردها همچون پردازش زبان‌های طبیعی، و شاخص‌بندی مستندات با استفاده از ترجیح نقش کلمه بر خود کلمه در حوزه بازیابی اطلاعات مورد توجه می‌باشد. هم‌چنین رفع ابهام ممکن است به صورت یکی از وظایف دسته‌بندی متون دیده شود [54][55]. در این‌جا، تکرار کلمه در زمینه همانند متون و نقش کلمه به عنوان دسته‌ها دیده می‌شود. کاملاً آشکار است، این مسأله یک حالت دسته‌بندی متون تک برچسبی است. رفع ابهام تنها یک مثال از مسائل رفع ابهام زبان طبیعی که خود یکی از مهم‌ترین مسائل زبان‌شناسی محاسباتی است، می‌باشد. مثال‌های دیگر که درگیر با این مسأله هستند شامل غلط‌یابی حساس به متن، تشخیص عبارات اضافی، برچسب‌گذاری مقوله‌ی واژگانی[1]، و انتخاب کلمه‌ی مناسب در ماشین‌های ترجمه می‌باشند. برای مقدمه‌ای بر این مسأله به [56] مراجعه شود.

3-5.دسته‌بندی سلسله‌مراتبی صفحات وب

اخیراً به دلیل جذابیت زیاد و امکان کاربرد آن، این مسأله برای دسته‌بندی خودکار صفحات وب نیز مورد توجه بسیاری قرار گرفته است. هنگامی که صفحات وب بدین صورت کاتالوگ‌بندی می‌شوند، امکان و کیفیت پرس‌وجو برای متورهای جستجوی وب بهتر شده است. هم‌چنین، آن‌ها آسان‌تر به دسته مورد نظر دسترسی پیدا می‌نمایند.

از آن‌جا که دسته‌بندی دستی صفحات وب عملاً نشدنی است. لذا مزایای دسته‌بندی صفحات وب به طور خودکار کاملاً نمایان می‌باشد. برخلاف کاربرد قبلی، هر دسته بایستی با استفاده از یک مجموعه متون $k_1 \le x \le k_2$ پر شود. در این‌جا باید روش‌های دسته‌بندی مبتنی بر دسته انتخاب گردد تا اضافه کردن دسته‌ای جدید و یا حذف دسته‌های بی‌مصرف امکان‌پذیر باشد.

با توجه به آنچه قبلاً در ارتباط با کاربردهای دسته‌بندی متون بحث گردید، دسته‌بندی خودکار صفحات وب دو ویژگی خاص خود را دارد.

[1] Part of Speech Tagging

۱. ذات مستندات ابر متن‌ها[۱]:

پیوندها یکی از منابع با ارزش اطلاعات در ارتباط با صفحاتی که به آن‌ها پیوند داده شده می‌باشند. تکنیک‌هایی که با این نگرش برای زمینه دسته‌بندی متون استفاده شده اند در نمونه کارهای [57][58][59][60][61] ارائه گردیده است. مجموعه‌ی این دسته از کارها توسط [62] مقایسه گردید.

۲. ساختار سلسله‌مراتبی مجموعه دسته‌ها:[۲]

هنگامی که ریز نمودن مسأله‌ی دسته‌بندی به دسته‌های کوچکتر مورد نظر است، این مسأله مورد استفاده قرار گیرد. تصمیم برای زیر دسته‌ها در هر کدام از گره‌های داخلی گرفته می‌شود. نمونه این کارها توسط [63][64][65][66][67][68] ارائه شده است.

[۱] Hypertextual Nature Of The Documents

[۲] Hierarchical Structure Of The Category Set

4. رهیافت یادگیری ماشین برای دسته‌بندی متون

در دهه 80 معمول‌ترین رهیافت (دست کم در کارهای عملی) برای تولید دسته‌بندی خودکار مستندات متنی، به صورت روش‌های دستی (به معنی تکنیک‌های مهندسی دانش، به صورتی که یک فرد خبره تصمیم گیری‌هایی در ارتباط با دسته‌بندی متون انجام می‌دهد) صورت می‌پذیرفت.

در روش دستی، همانند سیستم‌های خبره که شامل مجموعه‌ای از قوانین منطقی که به طور دستی تعریف می‌شوند، برای هر دسته قوانینی منحصر بفرد از نوع زیر تعریف می‌گردد:

if <*DNF formula*> **then** <*category*>

یک فرمول DNF[1] یک ترکیب فصلی از ترکیب‌های عطفی-شرطی می‌باشد. اگر متن داده شده این مجموعه شروط را ارضا نمود، آنرا تحت دسته category دسته‌بندی می‌کنند. معروف‌ترین مثال این رهیافت سیستم کانسچر[2] که به وسیله‌ی گروه کارنگی برای خبرگزاری رویترز ساخته شده است، می‌باشد.[39] نمونه‌ای از قوانین که در کانسچر استفاده شد در شکل 1 آورده شده است.

if ((*wheat* & *farm*) **or**
(wheat & commodity) **or**
(bushels & export) **or**
(*wheat* & *tonnes*) **or**
(*wheat* & *winter* & ¬ *soft*)) **then** WHEAT **else** ¬WHEAT

شکل 1 کلاس‌بند مبتنی بر قوانین برای دسته گندم ؛ کلمات کلیدی به صورت ایتالیک نشان داده شده‌اند و دسته‌ها به صورت حروف بزرگ نوشته شده‌اند

[1] Disjunctive Normal Form

[2] CONSTRUE

ضعف این روش، معضل استخراج قوانین کامل و مناسب از متون یک دسته بود. قوانین می‌بایست به صورت دستی توسط یک مهندس دانش با کمک یک خبره‌ی آن زمینه، تعریف گردند (در این‌جا، منظور از خبره، کسی است که دسته‌ی مناسب را برای متون انتخاب می‌کند). حال اگر مجموعه‌ای از این دسته‌ها بخواهند به‌روز شوند هر دو نفر مهندس دانش و فرد خبره دوباره باید برای این منظور با هم کار کنند و اگر دسته‌هایی با موضوعات کاملاً متفاوت با این موضوع نیاز بود، می‌بایست فرد خبره دیگری نیز متعاقباً بکار گرفته شود و این روند تکرار گردد.

اما از طرف دیگر، این روش می‌تواند به طور ابتکاری نتایج خوبی نیز بدهد. هیز و همکارانش [39] نتیجه‌ای در حدود 90 درصد بر روی یکی از زیر مجموعه‌های اطلاعات آزمایشی رویترز را گزارش کردند. اگرچه، هیچ دسته‌بند دیگری بر روی مجموعه داده‌های مشابه همچون کانسچر تست نشد و مشخص نیست که این انتخاب تصادفی بوده و یا این نتایج بر روی سرتاسر داده‌های رویترز نیز صادق بوده است. ولی تقریباً این نتایج از بهترین دسته‌بندهای تا اواخر دهه 90 بهتر بود. به طور کلی، آنگونه که یانگ [22] استدلال نموده است، نتایج بالا اجازه نمی‌دهد که به طور مؤثری در این زمینه استدلال نمود.

از اوایل دهه‌ی 90، رهیافت‌های یادگیری ماشین برای دسته‌بندی متون به طور عام مورد توجه قرار گرفت. به طوری که دست کم در مراکز تحقیقاتی به عنوان یک رهیافت جدی بر دیگر رهیافت‌ها اولویت یافت (برای اطلاعات بیش‌تر و جامع‌تر به [69] مراجعه شود). در این رهیافت، یک پرسه استنتاجی عمومی (معمولاً یادگیرنده نامیده می‌شود) به طور خودکار دسته‌بندها را برای یک دسته‌ی c_i می‌سازد این کار با مشاهده مشخصات یک مجموعه از متون که به طور دستی تحت ci یا $\overline{ci}$ توسط یک فرد خبره دسته‌بندی شده‌اند، انجام می‌شود. پروسه‌ی استنتاجی با جمع‌آوری اطلاعات از این متون، قادر خواهد بود که به دسته‌بندی یک متن جدید که قبلاً دیده نشده تحت دسته‌ی c_i بپردازد. از آن‌جا که پروسه یادگیری با نظارت نمونه‌هایی از هر یک از دسته‌ها برای یادگیری همراه است (در اصطلاح یادگیری ماشین، به مسأله‌ی دسته‌بندی در این روش یک فعالیت یادگیری باناظر گفته می‌شود[1]).

[1] محدوده مدیریت متون بر مبنای محتوای آن‌ها یک مثالی از نمونه‌های یادگیری بدون ناظر می‌باشد که عموماً به آن حوزه خوشه‌بندی متون گفته می‌شود. (مراجعه شود به بخش 1)

مزیت رهیافت یادگیری ماشین نسبت به مهندسی دانش مشهود می‌باشد. تلاش و نیروی مهندسی در این‌جا مصروف ساخت می‌شود نه دسته‌بندی و دسته‌بندها به طور خودکار تولید می‌گردند. به عبارت دیگر با استفاده از تعدادی نمونه‌ی محدود دسته‌بندها به طور خودکار تولید می‌گردند و این مسأله هنگامی که دسته‌ها می‌خواهند به‌روز شوند و یا به طور کلی تغییر کنند نیز صادق است.

در رهیافت یادگیری ماشین، متون پیش دسته‌بندی شده نقش منبع اصلی را بازی می‌کنند. در بیش‌تر حالت‌های معمول این متون در دسترس هستند. به عبارت دیگر آن‌ها به عنوان نمادین برای نمایش پروسه‌ی دسته‌بندی استفاده می‌شوند. حالت دیگر زمانی است که هیچ متن پیش‌دسته‌بندی‌شده به طور دستی موجود نباشد. حتی در این حالت رهیافت‌های یادگیری ماشین برای دسته‌بندی متون بسیار ساده‌تر از استخراج کلمه کلیدی می‌باشد. در حقیقت، دسته‌بندی دستی تعدادی محدود از متون بسیار ساده‌تر از تولید مجموعه‌ای از قوانین است که مشخصات آن دسته را شرح می‌دهند (برای مثال آوردن یک متن بسیار ساده‌تر از شرح مفاهیم کلمات آن متن است).

امروزه، دسته‌بندهایی که با تکنیک‌های یادگیری ماشین عرضه شده‌اند به سطوح بالایی از دقت رسیده‌اند و دسته‌بندهای خودکار با کیفیت (نه فقط به لحاظ صرفه اقتصادی) تولید شده‌اند.

4-1. مجموعه‌ی آموزشی، مجموعه‌ی آزمایشی و مجموعه اعتبار سنجی

رهیافت‌های یادگیری ماشین متکی بر موجودیت یک پیکره‌ی زبانی اولیه $\Omega=\{d_1,\ldots d_{|\Omega|}\} \subset D$ از متون پیش‌دسته‌بندی‌شده تحت $C=\{c_1,\ldots,c_{|C|}\}$ می‌باشند. مقدار تابع انتقالی $\Phi : D * C \rightarrow \{T, F\}$ برای هر جفت $<d_j, c_i>$ شناخته شده است. اگر $\Phi(d_j, c_i)= T$ باشد، به متن d_j یک مثال مثبت[1] برای c_i گفته می‌شود. برعکس، اگر $\Phi(d_j, c_i)= F$ باشد، به متن d_j یک مثال منفی[2] برای c_i گفته می‌شود.

[1] Positive Example

[2] Negetive Example

در مسائل تحقیقاتی و هم‌چنین در بیش‌تر مسائل عملی، هربار که یک دسته‌بند ساخته شد دقت آن را ارزیابی می‌نمایند. برای این منظور قبل از ساخت یک دسته‌بند، پیکره‌ی زبانی اولیه به دو دسته نه الزاماً هم‌اندازه تقسیم می‌شود:

- مجموعه‌ی آموزشی (و اعتبار سنجی): مجموعه‌ای از متون هستند که مقدار تابع انتقال برای آن‌ها مشخص می‌باشد. این مجموعه متون برای ساختن دسته‌بند و تقریب زدن تابع انتقال به کار می‌روند.
- مجموعه‌ی آزمایشی[۱]: مجموعه‌ای از متون که مقدار تابع انتقال برای آن‌ها مشخص است ولی از آن برای آزمایش و ارزیابی میزان دقت تابع انتقال تقریبی استفاده می‌کنند.

نوع دسته متون مجموعه‌ی آزمایشی نباید به صورت مشخص مشهود باشد؛ زیرا که در این صورت این مجموعه دیگر هیچ ارزش علمی نخواهد داشت (صفحه 129 از [69])، در نهایت در موارد عملی، پس از ارزیابی الگوریتم، سیستم بر روی سرتاسر مجموعه‌ی داده‌ها (اعم از آموزشی یا آزمایشی) آموزش داده می‌شود. این مسأله به دلیل توسعه مجموعه‌ی داده‌ها دقت سیستم را افزایش می‌دهد. در حقیقت میزان دقت ارزیابی شده در حالت اول دقت بدبینانه سیستم خواهد بود [70].

به این مسأله رهیافت آموزش-آزمایش نیز گفته می‌شود. یک حالت دیگر رهیافت به‌هم‌زدن kتایی[۲] می‌باشد (رجوع شود به صفحه 146 از [69])، در این حالت k تا دسته‌بند متفاوت با استفاده از تقسیم پیکره‌ی زبانی اولیه به k تا مجموعه‌ی آموزشی و آزمایشی مستقل از یکدیگر ساخته و ارزیابی می‌شود. در نهایت دقت الگوریتم با استفاده از میانگین دسته‌بندها محاسبه می‌گردد.

در هر دو این روش‌ها، پارامترهای درونی دسته‌بندها می‌باید مقداردهی مناسب شود. برای این منظور اغلب با آزمایش اولیه در مجموعه‌ی آموزشی صورت می‌گیرد. معمولاً مجموعه‌ی آموزشی به دو دسته‌ی آموزش اولیه و تست اولیه یا اعتبار سنجی[۳] تقسیم می‌شود. زمانی که سیستم به بالاترین دقت مورد انتظار رسید، دسته‌بند نهایی می‌گردد. اگر دسته اعتبار سنجی مستقل از مجموعه‌ی آموزشی اولیه بود، به آن دسته بیرون گذاشته شده[۴] نیز گفته می‌شود. کاملاً مشهود

---

[۱] Test Set

[۲] K-Fold Cross-Validation.

[۳] Validation Set.

[۴] Hold Out Set.

است که این مسأله برای حالت به‌هم‌زن kتایی نیز صادق است. باید ذکر کرد، مجموعه‌ی آزمایشی قطعاً باید از مجموعه اعتبار سنجی و مجموعه‌ی آموزشی مستقل باشد.

برای یک پیکره‌ی زبانی داده شده Ω، ممکن است که میزان تعمیم $g_\Omega(c_i)$ برای دسته c_i تعریف شود که درصد متونی که متعلق به دسته c_i هستند:

$$g_\Omega(c_i) = \frac{|\{d_j \in \Omega \mid \breve{\Phi}(d_j, c_i) = T\}|}{|\Omega|}$$

تعمیم مجموعه‌ی آموزشی دسته‌ی $g_{Tr}(c_i)$ و تعمیم مجموعه‌ی اعتبارسنجی $g_{Va}(c_i)$ و هم‌چنین تعمیم برای مجموعه‌ی آزمایشی $g_{Te}(c_i)$ می‌باشد. این توابع نیز به طریقه مشابه تعریف می‌گردد.

4-2. تکنیک‌های بازیابی اطلاعات و دسته‌بندی متون

دسته‌بندی متون به طور عمیقی متکی بر اصول حاکم بر بازیابی اطلاعات می‌باشد. دسته‌بندی متون یک عمل مدیریت متون مبتنی بر مفهوم می‌باشد. به طوری که مشخصه‌های مشترک بسیاری با اعمال بازیابی اطلاعات، همچون جستجوی متن دارد.

تکنیک‌های بازیابی اطلاعات در دو فاز از دسته‌بندهای متون مورد استفاده قرار می‌گیرد:

۱. شاخص‌بندی به شکل بازیابی اطلاعات همواره در خلال فاز عملیاتی بر روی متون پیکره‌ی زبانی اولیه و بر روی آن‌هایی که می‌خواهند دسته‌بندی بشوند یا شده‌اند، انجام می‌شود.

۲. تکنیک‌های از نوع بازیابی اطلاعات (همچون انطباق متن مورد تقاضا، تبدیل قالب پرس و جو) در ساخت استنتاجی دسته‌بندها مورد استفاده قرار می‌گیرد.

در اغلب رهیافت‌های مختلف معمولاً با تغییر قسمت (2) سعی در بهبود و ارتقای کیفیت دسته‌بندی می‌نمایند و بسیار به ندرت سعی در تغییر قسمت (1) می‌نمایند [70].

5. شاخص‌بندی متن و کاهش ابعاد

5-1. شاخص‌بندی متن

یک متن نمی‌تواند به صورت مستقیم توسط یک دسته‌بند یا یک الگوریتم دسته‌بندساز تفسیر شود. بلکه با استفاده از یک پروسه‌ی شاخص‌بندی که متن d_j را به یک نمایه (که محتویات آن را بیان می‌کند)، نگاشت داده می‌شود. این مهم کمک می‌کند تا یک‌نواختی و یک شکلی لازم برای متون مجموعه‌ی آموزشی، آزمایشی و اعتبار سنجی فراهم آورده شود. انتخاب نمایه برای متن، بسته به مسائل مختلفی دارد، همچون: 1- واحدهای معنایی (مسأله‌ی واژگان معنایی[1]) و 2- قوانین معنایی طبیعی برای ترکیب این واحدها (مسأله‌ی ترکیب معنایی[2]). معمولاً در دسته‌بندی متون مشکل دوم در نظر گرفته نمی‌شود (مشابه آنچه در بازیابی اطلاعات وجود دارد) و یک متن d_j، با یک برداری از وزن عبارت‌هایش نشان داده می‌شود. به عبارت دیگر $d_j = <w_{1j}, w_{2j}, \ldots, w_{|T|j}>$ به طوری که T مجموعه عبارت‌هایست که دست‌کم یک‌بار در سرتاسر مجموعه‌ی آموزشی آمده باشند[3] (در بعضی اوقات به آن ویژگی نیز گفته می‌شود) و $0 \leq w_{kj} \leq 1$ باشد. معمولاً تفاوت رهیافت‌ها در این زمینه به یکی از دلایل ذیل می‌باشد:

1. تفاوت در تعریف چیزی که "عبارت" نامیده می‌شود.

2. تفاوت در طریقه محاسبه وزن ترم‌ها.

یک انتخاب متداول برای مسأله اول، تعین هر کلمه به عنوان عبارت می‌باشد. اغلب این مسأله، رهیافت مجموعه کلمات یا کیسه کلمات[4] برای نمایش متون (مستقل از اینکه وزن‌ها دودیی باشند یا نباشند) گفته می‌شود.

---

[1] Lexical Semantics

[2] Compositional Semanics

[3] یک نگرش دیگر برای این مسأله رهیافت یادگیری مبتنی بر مدل های مخفی مارکوف می‌باشد [76][77].

[4] Word Bag

در تعدادی از مطالعات همچون [71][72][73] مسأله اینکه نمایه‌های پیچیده‌تر تاثیر آن‌چنان بسزایی در دقت مسأله با توجه به نتایج مشابه در بازیابی اطلاعات نخواهند داشت، بررسی شده است.[74] به صورت محدود، بعضی از محققان از عبارت‌ها به جای کلمات برای شاخص‌بندی متون استفاده نموده‌اند [37][47][75]. اما صرف‌نظر از تعریف عبارت، نتایج تجربی به طور یکسان با این مسأله برخورد نکرده‌اند.

- به طور صرفی و نحوی؛ تعریف عبارت با استفاده از قوانین گرامری زبان[73]
- به طور آماری؛ در نظر گرفتن یک ترکیب، نه به لحاظ قواعد گرامری بلکه بنا به همنشینی آن با مجموعه یا دنباله‌ای از کلمات که الگوی خاصی را دنبال می‌نمایند.

لیویز [73] استدلال کرد که استفاده از عبارت‌ها بیش‌ترین ارزش معنایی و کم‌ترین ارزش آماری را دارد. از طرف دیگر استفاده از کلمات کم‌ترین ارزش معنایی و بیش‌ترین ارزش آماری را دارد: یک شاخص‌بندی مبتنی بر عبارت زبانی، الفاظ و ترم‌های زیادی دارد و به همین نسبت مترادفات یا ترم‌های نزدیک به آن ترم به لحاظ معنایی بیش‌تری نیز وجود دارد. از طرف دیگر سازگاری کم‌تری در تخصیص و ارجاع کلمات به متون نیز وجود دارد (زیرا که کلمات مترادف به متن مشابه ارجاع نمی‌شوند یا به عبارت دیگر در یک متن از یکی از حالات مترادف استفاده می‌شود نه همه‌ی آن‌ها) و هم‌چنین فرکانس تکرار متن کم‌تری نیز برای کلمات را موجب می‌شود. برای شرح بیش‌تر به صفحه 40 از [73] مراجعه شود. به نظر می‌رسد که ترکیب این دو رهیافت یکی از روش‌های خوب برای مسأله دسته‌بندی متون می‌باشد: تزراس[1] و هارتمن[2] بهبود نسبی خوبی را با استفاده از عبارت‌های اسمی که از طریق ترکیب شرایط آماری و قواعد صرفی-نحوی بود، به دست آوردند [37]. به طوری که یک متد ضعیف نحوی-صرفی به همراه یک فیلتر آماری را (تنها عبارت‌هایی که دست‌کم در 3 متن از مثال مثبت از دسته c_i آمده‌اند، استفاده شده است) با هم ترکیب کردند.

همان‌طور که قبلاً نیز گفته شد، عموماً وزن‌ها در محدوده بین صفر و یک تعریف می‌شوند. ولی در حالت‌های خاص، از وزن‌دهی دودیی نیز استفاده می‌شود که در این شرایط (1 به معنای حضور و 0 به معنای عدم حضور کلمه در متن است و یا در حالت کلی‌تر تکرار بیش از m مرتبه یک کلمه در متن با عدد 1 و کم‌تر از آن با 0 نشان داده می‌شود. که در این‌جا m می‌تواند یک عدد ثابت یا بر

[1] Tzeras

[2] Hartmann

مبنای فرمولی محاسبه شود. یک مثال ساده برای آن می‌تواند ($m_{di} = 3*Length(d_i)$ باشد.) انتخاب وزن‌دهی دودویی یا غیر دودویی بستگی به الگوریتم یادگیری دسته‌بند دارد. در حالت شاخص‌بندی غیر دودویی برای تعیین وزن w_{kj} برای ترم t_k در متن d_j با استفاده از تکنیک‌های شاخص‌بندی در بازیابی اطلاعات استفاده می‌شود. معمولاً برای این منظور، تابع استاندارد tfidf استفاده می‌شود.

$$tfidf(t_k, d_j) = \#(t_k, d_j) \cdot \log \frac{|Tr|}{\#_{Tr}(t_k)},$$

به طوری که $\#(t_k, d_j)$ تعداد دفعاتی که ترم t_k در متن d_j رخ می‌دهد، و $\#Tr(t_k)$ تعداد متن‌های موجود در مجموعه‌ی آموزشی که ترم t_k در آن‌ها وجود دارد، می‌باشد. این تابع تضمین می‌کند که 0/1 ترم‌هایی که اغلب در یک متن رخ می‌دهد نماینده محتویات آن هستند و 0/2 کلماتی که در اکثر متون دیده می‌شوند ارزش بسیار کم‌تری از کلماتی که تنها در یک متن می‌آیند، دارند. شکل‌های مختلفی از tfidf موجود می‌باشد که تفاوت آن‌ها با یکدیگر در لگاریتم گیری، نرمال سازی یا دیگر توابع تصحیح می‌تواند باشد [74][78]. باید توجه داشت که در این فرمول (همانند بیش‌تر فرمول‌های شاخص‌بندی) وزن اهمیت یک ترم در یک متن در تعداد رخداد آن ترم است نه محل وقوع آن؛ به عبارت دیگر در این‌جا صرف‌نظر از موقعیت مکانی ترم اعم از ابتدا، وسط یا انتهای متن، تنها تعداد رخداد متن بر وزن آن تاثیر می‌گذارد. یعنی معنای یک متن به مجموعه‌ای از معنای واژگانی ترم‌هایی که در آن رخ داده‌اند، تقلیل می‌یابد. به منظور اینکه محدوده‌ی وزن‌ها در بازه [0, 1] باشد و متن‌ها با بردارهای هم اندازه از وزن‌ها نشان داده شوند، معمولاً وزن‌های tfidf باید نرمال شوند.

$$w_{kj} = \frac{tfidf(t_k, d_j)}{\sqrt{\sum_{s=1}^{|T|}(tfidf(t_s, d_j))^2}}.$$

گرچه tfidf نرمال‌شده یکی از عمومی‌ترین‌هاست، ولی دیگر توابع شاخص‌بندی همچون تکنیک‌های احتمال[60] یا تکنیک‌های شاخص‌بندی متون ساخت‌یافته [23] نیز مورد استفاده قرار می‌گیرند. به

خصوص وقتی که مجموعه‌ی آموزشی در ابتدا کاملاً مشخص نبود، دیگر توابع متفاوت با tfidf همچون تخمین tfidf نیز می‌تواند مورد استفاده قرار گیرد. به بخش 4.3 از [79] مراجعه شود.

معمولاً قبل از شاخص‌بندی، کلمات بی‌اهمیت همچون حرف تعریف[1]، حروف اضافه[2]، حروف ربطی و عطفی[3] حذف می‌شوند [80][81][82]. از سویی دیگر مناسب بودن استفاده از ریشه‌یابی‌ها (برای مثال گروه‌های اسمی که ریشه موفولوژیکی مشابه دارند) در دسته‌بندی متون نیز مورد بحث است. گرچه باید اذعان داشت که به طور مشابه با خوشه‌بندی بدون ناظرِ ترم‌ها، در بعضی اوقات ریشه‌یابی دقت الگوریتم را کاهش می‌دهد [83]. گرایش اخیر به اتخاذ روشی برای ریشه‌یابی است که به نحوی این کار انجام شود که هم فضای دامنه ترم‌ها و هم وابستگی آماری بین ترم‌ها کاهش یابد.

بسته به کاربرد، کل یا بخشی از یک متن می‌تواند به عنوان شاخص‌بندی مد نظر قرار گیرد. تا هنگامی که انتخاب اول قوانین باشند، استثناء نیز وجود خواهد داشت. برای مثال، در کاربرد دسته‌بندی حق امتیاز [21] تنها عنوان، چکیده، 20 خط اول خلاصه، و بخشی که ایده‌ی نو را توضیح می‌داد، مد نظر قرار می‌گرفت. این رهیافت تنها با در نظر گرفتن این مطلب که مستندات حقِ امتیاز ساخت‌یافته هستند، قابل اجرا می‌باشد. به طور مشابه هنگامی که عنوان یک متن مشخص باشد، می‌توان اهمیت بیش‌تری را به کلماتی داد که در آن هستند [84][85]. هنگامی که یک متن قالب خاصی ندارد و به صورت متن تنهاست، تشخیص قسمتی که بیش‌ترین ارتباط را با کل موضوع دارد، یک مسأله بغرنج و غیر مشخص می‌شود.

5-2. کاهش ابعاد

ابعاد بزرگ فضای ترم‌ها در دسته‌بندی متون (یعنی مقدار بزرگ |T|) معمولاً دردسر ساز می‌باشد. در حقیقت، با بزرگ شدن فضای ترم‌ها، تعداد ویژگی‌ها نیز افزایش می‌یابد که از طرفی باعث پیچیدگی بیش‌تر (صرف هزینه‌ی زمانی و فضای حافظه‌ی بیش‌تر) و از طرفی عدم وابستگی بین اطلاعات کم‌تر می‌گردد (داده‌هایی که به هم وابسته هستند، عموماً ارزش دسته‌بندی ندارند. بیش‌تر

[1] Artcle

[2] Prepositions

[3] Conjunctions

دسته‌بندها در این موارد به مشکل بیش یادگیری برمی‌خورند). تعداد ترم‌ها در شاخص‌بندی متون به صورت ساده، بسته به تعداد متون در یک پیکره‌ی زبانی افزایش می‌یابد. برای مثال در حدود 20000 متن موجود در مجموعه داده رویترز 21578[1] در حدود 15000 ترم متفاوت دارد.

بنابراین عموماً قبل از دسته‌بندی، داده‌ها از قسمت کاهش ابعاد می‌گذرند. به منظور رهایی از این مشکلات و مسائل، در بحث کاهش ابعاد سعی می‌کنند که با حذف ترم‌های بی‌ارزش به ابعاد فضای بردار از $|T|$ به $|T| >> |T'|$ برسند. تکنیک‌های کاهش ابعاد در دو حوزه عمومی و محلی قابل بحث می‌باشد.

۱. در حوزه‌ی خصوصی

برای هر دسته C_i یک مجموعه ترم T'_i یافت گردد به شرطی که $|T| >> |T'_i|$

۲. در حوزه‌ی عمومی

یک مجموعه ترم T' یافت شود به طوری که برای تمام دسته‌ها $|T| >> |T'_i|$ باشد.

به عبارت دیگر در کاهش ابعاد عمومی سعی می‌شود تا با تحلیل پیکره‌ی زبانی و کلیه متون موجود در مجموعه‌ی آموزشی ترم‌هایی را که ارزش پایینی در کاربرد مد نظر دارند، تعیین گردند و این دسته از ترم‌ها به صورت یک لیست ثابت، معین می‌گردند. ترم‌های متن ورودی به صورت خودکار توسط این لیست فیلتر می‌گردند. در حوزه‌ی محلی، همین کار برای هریک از دسته‌ها به طور مجزا انجام می‌شود.

بنابر آنچه گفته شد، مسأله‌ی کاهش ابعاد خود یکی از زمینه‌های جالب توجه در تحقیقات مربوط به بازیابی اطلاعات و مخصوصا دسته‌بندی متون می‌باشد. عمده تحقیقات تاکنون بر دو محور اصلی استوار بوده‌اند:

- با انتخاب ترم‌ها

یک مجموعه ترم بر مبنای تئوری اطلاعات یا ویژگی‌های آماری از متن‌ها انتخاب می‌شود.

- با استخراج ترم‌ها

[1] Reatuers 21578 Data Set.

ترم‌ها در فضای ترم جدید 'T از طریق تابع تبدیل خاصی به دست می‌آید ($T' = \delta(T)$) به طوری که ممکن است ترم‌های 'T کاملا با ترم‌های اصلی متفاوت باشد.

5-2-1. کاهش ابعاد با استفاده از انتخاب ترم‌ها

اولین رهیافت برای کاهش ابعاد با استفاده از انتخاب ترم‌ها، رهیافت فیلترکردن نامیده می‌شود. با استفاده از ابزارهایی که تئوری آمار یا اطلاعات فراهم نموده است ترم‌های بی‌ربط از ترم‌های استخراج شده فیلتر می‌شوند. در نهایت دسته‌بندها مستقل از تابع فیلترساز استفاده شده، با استفاده از فضای ترم کاهش یافته تولید می‌شوند.

یکی دیگر از رهیافت‌ها که تکنیک لفاف[1] [86] نیر نامیده می‌شود این است که انتخاب ترم‌ها بر مبنای الگوریتم دسته‌بندی استفاده شده، مشخص می‌گردد. با شروع از یک فضای ترم اولیه، یک فضای ترم جدید با اضافه نمودن یا کاهش ترم‌ها تولید می‌شود. در نهایت دسته‌بند با استفاده از فضای ترم جدید آموزش یافته و بر روی مجموعه اعتبار سنجی آزمایش می‌شود. فضای ترمی که بهترین جواب را تولید نماید به عنوان مجموعه ترم نهایی برای الگوریتم دسته‌بند انتخاب می‌شود. اگرچه فضای ترم مناسب برای دسته‌بندها مزیت‌های غیر قابل انکاری دارد ولی هزینه و پیچیدگی‌های محاسباتی این روش یکی از بزرگترین نقاط ضعف آن است. لذا در این گزارش این روش نادیده گرفته خواهد شد.

فرکانس متن: یکی از توابع ساده کاهش ابعاد مبتنی بر فرکانس متن یک ترم t_k می‌باشد. برطبق قانون زیپف-مندلبورت، ترم‌هایی که فرکانس متن بسیار پاین یا بسیار بالایی دارند می‌توانند نادیده گرفته شوند. نتایج تجربی نشان داده‌اندکه به حذف فاکتور 10 ترم‌ها بدون از دست دادن اطلاعات باارزش می‌توان دست زد.

[1] Wrapper Thechnique

جدول 1 مهم‌ترین توابع انتخاب ترم‌ها [70] با در نظر گرفتن دسته Ci. برای در نظر گرفتن یک معیار عمومی این توابع باید با یکدیگر ترکیب شوند. ترم‌هایی که بیش‌ترین نتیجه را برمی‌گردانند انتخاب می‌شوند.

Function	$f(\mathsf{C_i}, t_k)$	Mathematical form
DIA association factor	$z(t_k, \mathsf{C_i})$	$Pr[(\mathsf{C_i}\|t_k)]$
Information gain	$IG(t_k, \mathsf{C_i})$	$\sum_{\mathsf{C_k}\in\{\mathsf{C_i},\bar{\mathsf{C_i}}\}}\sum_{t\in\{t_k,\bar{t_k}\}} Pr[t,\mathsf{C_k}] * \log\frac{Pr[t,\mathsf{C_k}]}{Pr[t]*Pr[\mathsf{C_k}]}$
Mutual information	$MI(t_k, \mathsf{C_i})$	$\log\frac{Pr[t_k,\mathsf{C_i}]}{Pr[t_k]*Pr[\mathsf{C_i}]}$
Chi-Square	$\chi^2(t_k, \mathsf{C_i})$	$\|T\|\frac{[Pr[t_k,\mathsf{C_i}]*Pr[\bar{t_k},\bar{\mathsf{C_i}}]-Pr[t_k,\bar{c_i}]*Pr[\bar{t_k},\mathsf{C_i}]]^2}{Pr[t_k]*Pr[\bar{t_k}]*Pr[\mathsf{C_i}]*Pr[\bar{c_i}]}$
NGL coefficient	$NGL(t_k, \mathsf{C_i})$	$\sqrt{\|T\|}\frac{[Pr[t_k,\mathsf{C_i}]*Pr[\bar{t_k},\bar{\mathsf{C_i}}]-Pr[t_k,\bar{c_i}]*Pr[\bar{t_k},\mathsf{C_i}]]}{\sqrt{Pr[t_k]*Pr[\bar{t_k}]*Pr[\mathsf{C_i}]*Pr[\bar{c_i}]}}$
Relevancy score	$RS(t_k, \mathsf{C_i})$	$\log\frac{Pr[t_k\|\mathsf{C_i}]+d}{Pr[t_k\|\bar{c_i}]+d}$
Odds ratio	$OR(t_k, \mathsf{C_i})$	$\frac{Pr[t_k\|\mathsf{C_i}]*(1-Pr[t_k\|\mathsf{C_i}])}{(1-Pr[t_k\|\mathsf{C_i}])*Pr[t_k\|\bar{\mathsf{C_i}}]}$
GSS coefficient	$GSS(t_k, \mathsf{C_i})$	$Pr[t_k, \mathsf{C_i}] * Pr[\bar{t_k}, \bar{c_i}] - Pr[t_k, \bar{c_i}] * Pr[\bar{t_k}, \mathsf{C_i}]$

تابع‌های انتخاب مبتنی بر تئوری اطلاعات و تئوری‌های آماری: متدهای پیشرفته مشتق شده از تئوری‌های آماری و اطلاعاتی در بسیاری از موارد مختلف برای کاهش ابعاد حتی تا حدود فاکتور 100 استفاده شده‌اند. در جدول 1 متداول ترین تابع‌های انتخاب ترم را که با مثال در [70] شرح داده شده‌اند، آورده شده است.

یک تابع $f(t_k, c_i)$، ترم t_k را برای دسته c_i را که در مجموعه‌های نمونه‌های مثبت و منفی پخش شده‌اند را انتخاب می‌کند (کاملاً واضح است، ترمی که تنها در مجموعه نمونه‌های مثبت یا منفی رخ دهند، با ارزشترین ترم‌ها می‌باشند). برای استنتاج یک شرط کلی مبتنی بر یک تابع انتخاب ترم، این توابع باید بر روی مجموعه دسته‌های داده شده C ترکیب شوند. ترکیب‌های معمول برای به دست آوردن $f(t_k)$:

- **جمع**: مجموع تابع انتخاب ترم بر روی تمام دسته‌ها محاسبه می‌شود:

$$f_{sum}(t_k) = \sum_{i=1}^{C} f(t_k, \mathsf{C_i})$$

- **جمع وزن‌دهی شده**: مجموع تابع انتخاب ترم بر روی تمام دسته‌ها به طور وزن‌دهی شده با وزن احتمال دسته:

$$f_{wsum}(t_k) = \sum_{i=1}^{C} Pr[\mathrm{C_i}] f(t_k, \mathrm{C_i})$$

- **بیشینه**[1]: بیشینه‌ی تابع انتخاب ترم بر روی تمام دسته‌ها انتخاب می‌شود:

$$f_{max}(t_k) = \max_{i}^{C} f(t_k, \mathrm{C_i})$$

ترم‌هایی که بیش‌ترین نتایج را با در نظر گرفتن تابع انتخاب برمی‌گردانند، به عنوان فضای ترم جدید در نظر گرفته می‌شوند و مابقی ترم‌ها حذف می‌شوند. نتایج تجربی نشان داده است که $\{OR_{sum}, NGL_{sum}, GSS_{max}\} > \{X^2_{max}, IG_{sum}\} > \{X^2_{wsum}\} >> \{MI_{max}, MI_{sum}\}$ به طوری که منظور از > یعنی "بهتر عمل می‌کند از".

5-2-2. کاهش ابعاد با استفاده از استخراج ترم

متدهای استخراج ترم، یک فضای ترم جدید 'T با تولید ترم‌های ترکیبی جدید از مجموعه اصلی را می‌سازند. این دسته از متدها سعی می‌کنند تا با انجام یک کاهش ابعاد با استفاده از جایگزینی کلمات با مفاهیم‌شان یک فضای ترم کوچک‌تر را تولید کنند.

دو متد که به طور مشخص در تحقیقات مورد بررسی قرار گرفته اند:

- خوشه‌بندی ترم‌ها[2]
- شاخص‌بندی معنایی مخفی[3]

خوشه‌بندی متون: گروه‌بندی ترم‌ها با استفاده از بالاترین درجه‌ی معنایی مرتبط جفت-جفت ترم‌ها با یکدیگر انجام می‌شود، به طوری که این گروه‌ها در فضای ترم‌ها به جای ترم‌های تکی نشان داده می‌شوند. بنابراین، یک معیار شباهت بین کلمات باید تعریف شود و تکنیک‌های خوشه‌بندی

[1] Maximum

[2] Term Clustering

[3] Latent Semantic Indexing (LSI)

(برای مثال k-mean) بر روی آن‌ها اعمال شود. برای مروری بر خوشه‌بندی متون [70] و [87] مطالعه شود.

شاخص‌بندی معنایی مخفی: LSI بردار ترم متن را به ابعاد کوچک‌تری از فضای ترم‌ها فشرده سازی می‌نماید. میزان کاهش ابعاد فضای ترم با ترکیب ترم‌ها در فضای ترم اصلی به طور خطی حرکت می‌کند. تبدیل با استفاده از یک مقدار تکی تجزیه (SVD)، از ماتریس ترم متن از فضای اصلی ترم انجام می‌شود. برای ماتریس ترم-با-متن D_{m*n} که $m = |T|$ تعداد ترم‌هاست و $n = |D|$ تعداد متون می‌باشد. SVD به صورت زیر انجام می‌شود:

$$D = U\ G\ B$$

که U_{m*r} و B_{r*n} ستون‌های ارتونرمال را دارد و G_{r*r} یک ماتریس قطری با مقادیر تکی از ماتریس اصلی D می‌باشد. به طوری که $r \leq \min(m, n)$ رتبه‌ی ماتریس ترم-با-متن D می‌باشد.

تبدیل فضا، یعنی $r - k$ از کم‌ترین مقدارهای تکی از G حذف گردند (مقدارشان برابر صفر قرار می‌گیرد). که نتایج در یک ماتریس ترم-با-متن جدید

$$\hat{D} = \hat{U}_{m*k}\ \hat{G}_{k*k}\ \hat{B}^{T}_{k*n}$$

که تقریبی از D است. ماتریس $\hat{G}$ با حذف مقادیر تکی کوچک از G به دست می‌آید. $\hat{U}$ و $\hat{D}$ از حذف سطر و ستون‌های متناظر به دست می‌آید. بعد از به دست آمدن این نتایج از SVD مبتنی بر داده‌های آموزشی، متن جدید با نگاشت ذیل به دست می‌آید که به ابعاد فضای کوچک‌تری تقلیل می‌یابد ([88] و [89] مطالعه شود).

$$\mathbf{d'^{\hat{a}}} = \hat{G}^{-1}\ \hat{U}^{T}\ \mathbf{d^{\hat{a}}}$$

به طور اساسی، LSI سعی می‌کند که ساختار مخفی در الگوی کلمات استفاده شده در سرتاسر متن را با استفاده از متدهای آماری استخراج نماید. نتایج انجام شده در [47] نشان داده شده است که برای یک دسته‌بندی خوب، ترم‌هایی که به عنوان بهترین ترم‌ها برای یک دسته به وسیله‌ی روش انتخاب ترم X^2 انتخاب نشده‌اند، به وسیله‌ی LSI ترکیب می‌شوند. بعلاوه، هم‌چنین نشان داده شده است که برای یک تفکیک کننده خطی[1] و رگرسیون منطقی[2]، LSI از X^2 بسیار مؤثرتر است، اما

[1] Linear Discriminent

[2] Logisitic Regression

دقتی برابر با دسته‌بندی شبکه عصبی دارد. بعلاوه، [90] نشان داده است که استفاده از LSI برای تولید نمایه دسته‌ای مشخص، بازده‌ای بهتر نسبت به تولید نمایه عمومی LSI دارد.

از مهم‌ترین روش‌های استخراج ویژگی‌ها می‌توان به موارد ذیل اشاره نمود:

- اطلاعات دوسره[۱] [91][92]: در این روش سعی می‌شود تا با استفاده از اطلاعات دوسره بین کلمات در متون و یا کلمات و دسته‌ها، با انتخاب ویژگی‌های بارزش‌تر و وزن‌دهی مؤثر آن‌ها، دقت الگوریتم را افزایش دهد. اگرچه این روش ابعاد بردار بازنمایی متون را کاهش می‌دهد، ولی این روش به دلیل محاسباتی در مجموع پر هزینه است [4].
- TFIDF[۲] [4][92]: این وزن‌دهی بر مبنای تعداد تکرار کل کلمه و معکوس تعداد متونی که این کلمه را در بر دارند، محاسبه می‌گردد. به عبارت دیگر، هر چه تعداد تکرار کلمه در متن بیش‌تر و تعداد متن‌هایی که این کلمه را در بر دارند کم‌تر باشد، وزن آن ویژگی بیش‌تر است. واضح است، کلمه‌ای که به طور یک‌نواخت در تمام متن‌ها آمده است، ارزش خاصی ندارد.
- اندیس‌گذاری معنایی مخفی[۳] [15][93]: یکی دیگر از شماهای رمز گذاری متن‌هاست که بر مبنای ساختار معنایی یک پیکره‌ی زبان[۴] با تعیین بیش‌ترین فاکتورهای آماری باارزش در فضای وزن‌دهی کلمه، به استخراج ویژگی‌ها می‌پردازد. مزیت این روش علاوه بر کاهش ابعاد، در کاوش در حساب‌رسی روابط بین گروه‌های کلمات که در خود متن‌ها رخ می‌دهد، می‌باشد.

تاکنون دسته‌بندهای مختلفی برای انجام این کار همچون تئوری بیز[۵] و شکل‌های مختلف آن [94][95][2] ماشین بردارهای حامی[۶] [15] مدل n-gram [95] آموزش استنتاجی(قیاسی)[۱] [96][97] برای این مسأله پیشنهاد شده است.

---

۱ Mutual Information

۲ Term Frequency Inverse Document Frequency

۳ Latent Semantic Indexing (LSI)

۴ Corpus

۵ Naïve Bayes Theory

۶ Support Vector Machine

3-5.روش احتمالی بیز

روش تئوری بیز یکی از روش‌های متداول در دسته‌بندی متون می‌باشد. در این روش متن به صورت مجموعه‌ای از کلمات مستقل از یکدیگر و مستقل از محل قرار گرفتن در متن دیده می‌شود. لذا تعریف تابع احتمال هر متن از حاصل‌ضرب احتمال کلمات آن و احتمال رخداد متنی با آن طول به دست می‌آید (فرمول 1). احتمال هر دسته نیز، از تعداد متن‌های متعلق به آن دسته نسبت به تعداد کل متن‌ها به دست می‌آید (فرمول 2).

$$p(d_i) = p(|d_i|) * \prod_{k=1}^{|d_i|} p(w_k^i) \quad (1) \qquad p(c_j) = \frac{\#(d_i \in c_j)}{\#(d_i)} \quad (2)$$

که در این‌جا d_i متن iام مجموعه‌ی آموزشی و w_k^i کلمه kام در متن iام بوده و تابع (.)# تعداد آیتم‌ها می‌باشد. حال با توجه به تئوری بیز، احتمال تعلق هر متن به هر دسته طبق (3) محاسبه می‌گردد.

$$p(c_j|d_i) = \frac{p(c_j) * p(d_i|c_i)}{\sum_{c_j \in C} p(c_j) * p(d_i|c_i)} \quad (3)$$

حال اگر احتمال رخداد کلیه متن‌ها با طول متفاوت یکسان باشد. باتوجه به اینکه مخرج کسر در (3) برای کلیه‌ی دسته‌ها یکسان است، (3) به (4) تبدیل می‌گردد.:

$$c_j = \arg\max_j \left(p(c_j) * \prod_{k=1}^{|d_i|} p(w_k^i|c_j) \right) \quad (4)$$

در این‌جا تابع argmax اندیس دسته‌ای است که بیش‌ترین مقدار نسبی را برای کلیه jهای معتبر را بر می‌گرداند.

---

[1] Inductive Learning

5-4. روش مدل n-gram

مدل n-gram در ابتدا برای مسائل پردازش گفتار معرفی شد. ولی هم اکنون نگارش‌های گوناگونی از این مدل برای مسائل پردازش زبان‌های طبیعی نیز به طور وسیع مورد استفاده قرار گرفته است [95]. در این‌جا هدف مدل‌سازی زبان، تخمین احتمال رخداد طبیعی دنباله‌ای از کلمات $s = w_1 w_2 \dots w_N$، یا به عبارت ساده‌تر احتمال دنباله کلماتی که واقعاً رخ می‌دهند (و احتمال پایین در دنباله کلمه‌ای که هرگز رخ نمی‌دهد) می‌باشد. اگر یک دنباله کلمه‌ای $w_1 w_2 \dots w_N$ به عنوان پیکره‌ی زبانی آزمایشی استفاده شود، کیفیت مدل زبان را می‌توان با استفاده از (5) و معیار آنتروپی را با استفاده از (6) ارزیابی نمود، به نحوی که هدف کاهش آن‌ها می‌باشد.

$$Perplexity = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{p\left(w_i \middle| w_1 w_2 \cdots w_{i-1}\right)}} \quad (5)$$

$$Entropy = \log_2 Perplexity \quad (6)$$

حال $P(w_i|w_1 w_2 \dots w_{i-1})$ طبق (7) محاسبه می‌گردد:

$$p\left(w_i \middle| w_1 w_2 \cdots w_{i-1}\right) = \frac{\#\left(w_1 w_2 \cdots w_i\right)}{\#\left(w_1 w_2 \cdots w_{i-1}\right)} \quad (7)$$

در حالت کلی این روش یک مدل توسعه یافته از نگرش تئوری بیز می‌باشد. در این روش برخلاف تئوری بیز، محل قرار گیری کلمات نیز در احتمال رخداد متن مهم می‌باشد. با در نظر گرفتن این مسأله (1) به (8) تبدیل می‌گردد.

$$p(d_i) = p(|d_i|) * \prod_{k=1}^{|d_i|} p\left(w_k^i \middle| w_1^i w_2^i \cdots w_{k-1}^i\right) \quad (8)$$

حال اگر در این‌جا نیز احتمال رخداد طول متن یکسان بوده و دنباله‌ی کلمات بـه n کلمـه محـدود شود، فرمول (7) به (9) ساده می‌گردد.

$$p(d_i) = \prod_{k=1}^{|d_i|} p\left(w_k^i \middle| w_{k-n+1}^i \cdots w_{k-1}^i\right) \quad (9)$$

همان‌طور که در بخش قبل گفته شد و با در نظر گرفتن تئوری بیز (4) به (10) تبدیل می‌گردد.

$$c_j = \arg\max_j \left(p(c_j) * \prod_{k=1}^{|d_i|} p_{c_j}\left(w_k^i \middle| w_{k-n+1}^i \cdots w_{k-1}^i\right) \right) \quad (10)$$

که در این‌جا $p_{c_j}\left(w_k^i \middle| w_{k-n+1}^i \cdots w_{k-1}^i\right)$، احتمال کلمه w^i_k است به شرطی که دنباله $w^i_{k-n+1} \ldots w^i_{k-1}$ و دسته c_j رخداده باشد.

5-5. دسته‌بندهای خطی

دسته‌بندهای خطی به سبب سادگی ذاتی، پشتوانه‌ی بسیار خوب تئوریکی دارند. یکـی از مشـکلات این دسته از دسته‌بندها محدود بودن فرض‌های مسأله است. یک دسته‌بند خطی از ترکیـب خطـی تمام ترم‌ها از یک فضای ترم‌ها یا ویژگی‌هاست. در حالت کلی یک دسته‌بند خطی $h(\vec{d_j}) \rightarrow \{1, -1\}$ به صورت زیر می‌باشد:

$$h(\vec{d_i}) = sign(\vec{w} \cdot \vec{d_i} - \theta) = sign\left(\sum_{k=1}^{|T|} w_k * d_{k,i} - \theta\right)$$

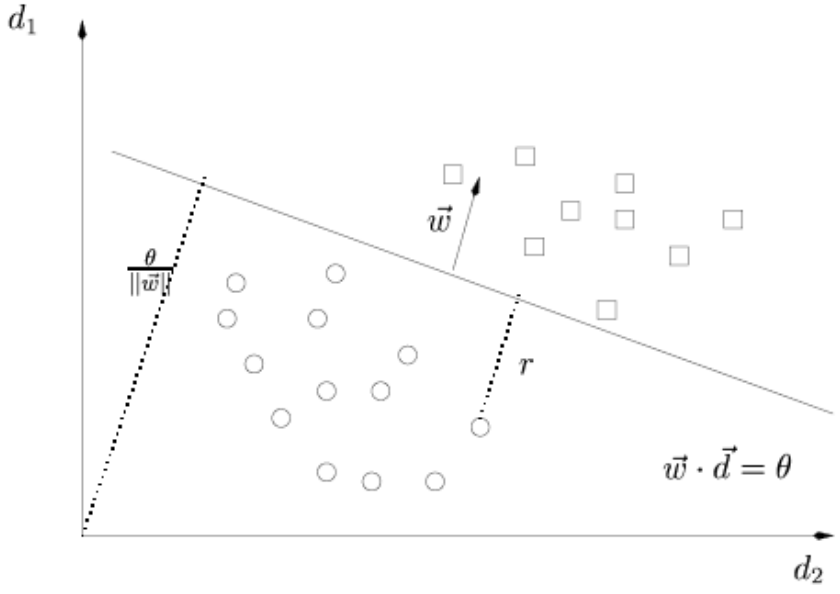
در جایی که w_k، وزن ترم k، و $d_{k,i}$ مقدار ترم k در متن i باشد. بنابراین هر دسته c_j با یک بردار وزن نمایش $\vec{w}_j$ داده می‌شود که اگر ضرب داخلی $\vec{w}_j \cdot \vec{d}_i$ از حد آستانه خاصی θ_j فراتر رفت یک متن $\vec{d}_i$ را به یک دسته تخصیص می‌دهد و بالعکس.

شکل 2 یک دسته‌بند خطی برای حالت فضای دو بعدی را نشان می‌دهد. معادله $\vec{w}_j \cdot \vec{d}_i = \theta_j$ رویه (سطح) تصمیم را (در حالت ابرصفحه[1]) در فضای |T| تعریف می‌کند. بردار وزن $\vec{w}$ پروجکشن نرمال[2] از ابر صفحه‌های مجزا در جایی که فاصله‌ی ابرصفحه‌ها از مقدار اصلی برابر است با:

$$\frac{\theta}{\|\vec{w}\|}$$

فاصله‌ی نمونه‌های آموزشی تا ابرصفحه‌ها برابر است با:

$$r = \frac{\vec{w} \cdot \vec{d} - \theta}{\|\vec{w}\|}$$



شکل2 دسته‌بند خطی که داده‌های آموزشی را در حالت دودوئی از یکدیگر جدا می‌کند. چهارگوش‌ها و دایره‌ها نمونه‌های آموزشی مثبت و منفی هستند.

[1] Hyperplane

[2] Normal Projection

آموزش دسته‌بندها به طرق مختلفی می‌تواند صورت پذیرد. یکی از معروف‌ترین الگوریتم‌های این دسته الگوریتم پرسپترون[۱] است که در حقیقت یک الگوریتم کاهش گرایان می‌باشد. مشابه با الگوریتم پرسپترون، وینو[۲] یک الگوریتم کاهش گرادیان چندگانه[۳] است. هر دوتای این الگوریتم‌ها قادر به یادگیری دسته‌بندی خطی در حالت‌های تفکیک‌پذیر خطی هستند. یک آلترناتیو برای الگوریتم پرسپترون، ماشین‌های بردار حامی[۴] که در قسمت زیر توضیح داده شده است، می‌باشند. SVMها قادر به جدا سازی بهینه ابرصفحه در شرایط تفکیک‌پذیر خطی هستند. البته نوع خاصی از آن‌ها برای مسائل غیر تفکیک‌پذیر خطی برای رسیدن به کم‌ترین میزان خطا نیز ارائه شده است. برای مثال، از دیگر الگوریتم‌های مهم یادگیری ماشین رویه‌های کمینه کردن مربع خطا[۵] مثل ویدرو هوف[۶] و رویه‌های برنامه ریزی خطی[۷] می‌باشند. مقدمه‌ای بر آن‌ها در [98] آمده است.

5-5-1. ماشین‌های بردار حامی

SVMها دسته‌بندهای خطی هستند. آن‌ها سعی می‌کنند ابرصفحه‌ای را بیابند به طوری که بیش‌ترین حاشیه‌ی مجاز بین ابر صفحه و نمونه‌های مثبت و منفی را دربرداشته باشد. شکل 3، ایده‌ی دسته‌بندهای بیش‌ترین حاشیه را نشان می‌دهد. برای شرح بیش‌تر بر تئوری‌های این روش به [99] و [100] مراجعه شود. نمونه‌ای از این گونه دسته‌بندی برای متون در [101] و [102] آورده شده است.

---

[۱] Perceptron Algorithm

[۲] Winnow

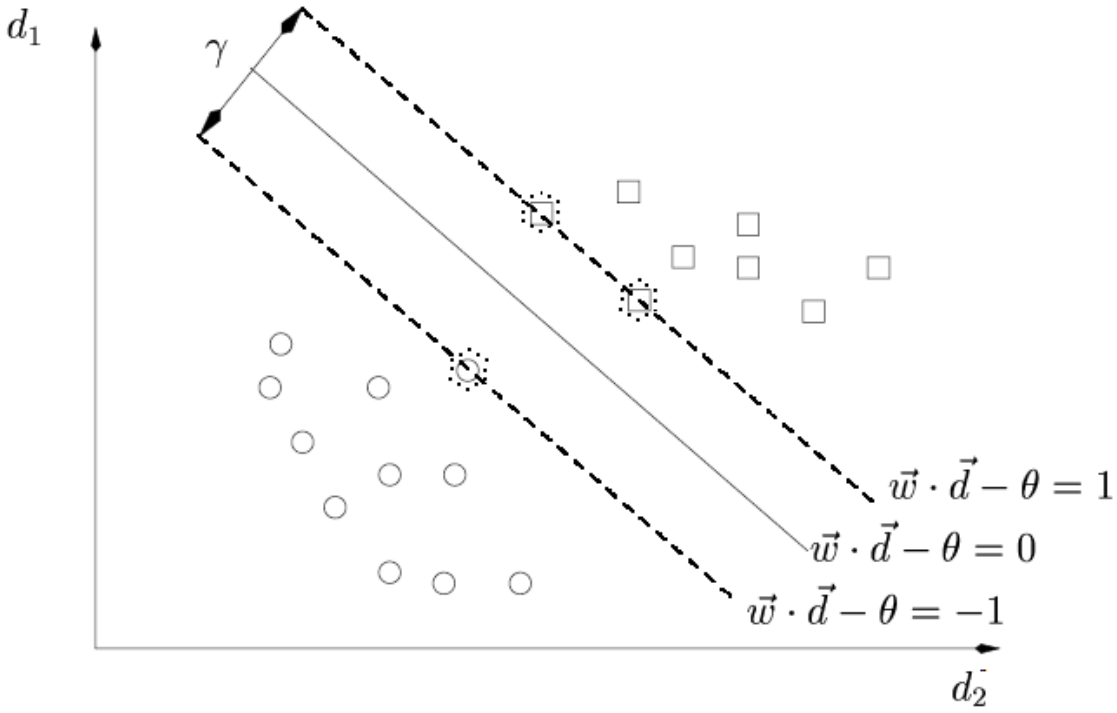
[۳] Multiplicative Gradient Decent Algorithm

[۴] Support Vector Machine (SVM)

[۵] Minimum Squered Error Producer

[۶] Widrow Hoff

[۷] Linnear Programming Producer



شکل 3 شکل فوق دسته‌بند خطی‌ای که بیش‌ترین حاشیه را برای نمونه‌های آموزشی که در حالت دودئی از یکدیگر جدا شده‌اند، نشان می‌دهد. خط‌چین‌ها، بردارهای حامی هستند.

باید خاطر نشان کرد، اگرچه روش حالت‌های تفکیک‌ناپذیر خطی نیز بسط داده شده است و هم‌چنین ماشین بردار حامی غیر خطی با گسترش ابعاد فضا نیز پیشنهاد شده است[1] ولی معمولاً کماکان آموزش این روش از مرتبه‌ی درجه‌ی دوم و بالاتر است که هزینه‌ی محاسباتی، زمانی و حافظه‌ای در این روش‌ها یک معضل تمام عیار گریده است [103].

5-6. نتیجه‌گیری

همان‌طور که در این بخش به تفضیل بررسی شد، مسأله دسته‌بندی به علت مشکلات مربوط به ذات زبان طبیعی و هم‌چنین بزرگی فضای ویژگی‌ها برای موارد دسته‌بندی مشکل می‌باشد. به

[1] این مسأله خود به زمانبر شدن و افزایش پیش نیازهای اجرا کمک می‌کند

عبارت دیگر می‌باید از الگوریتم‌های کاهش ابعاد که منجر به کم‌ترین از دست رفتن اطلاعات به عنوان پیش‌پردازش بهره جست.

علاوه بر استفاده از الگوریتم‌های کاهش ابعاد، به صورت تجربی نشان داده شده است که استفاده از الگوریتم‌های مدل زبانی، تئوری بیز و دیگر الگوریتم‌های احتمالی به دلیل عدم نیاز به پرسه مرحله‌ای برای آموزش (معمولاً در این روش‌ها با استفاده از یک یا چند فرمول احتمال ویژگی‌ها محاسبه می‌شود و برای به دست آوردن احتمال تعلق برای هر دسته از ضرب احتمال ویژگی‌هایش به طور ساده بهره گرفته می‌شود.)

6. دسته‌بندی خودکار برای متون زبان فارسی

6-1. مقدمه‌ای بر زبان‌های طبیعی

بنابر کتاب [104]، "در زبان‌شناسی مقایسه ای، زبان‌ها را، از نظر ساخت‌واژه، در برش اول به دو گروه کلی "تک واژکی[۱]" و "چند واژکی[۲]" می‌توان تقسیم کرد.

- تک واژکی زبانی است که در آن هر واژه فقط از یک واژک تشکیل شده که تغییرناپذیر است. در این حالت آرایش جمله، روابط نحوی را مشخص می‌کند. زبان‌های چینی و ویتنامی از این دسته هستند.
- چند واژکی‌ها، ساختمان واژه از یک یا چند واژک تشکیل شده است. این گروه از زبان‌ها خود، به سه دسته‌ی "پیوندی[۳]"، "ترکیبی[۴]"، و "بساوندی[۵]" تقسیم شده است.

۱. در زبان‌های پیوندی مرز بین واژک‌ها در واژه مشخص است و تطابق یک به یک بین واژک‌ها و مفاهیم آن‌ها وجود دارد، مانند زبان ترکی.

۲. در زبان‌های ترکیبی مرز بین واژک‌ها مشخص نیست و تطابق یک به یک بین واژک‌ها وجود ندارد، مانند زبان عربی و لاتین.

۳. در زبان‌های بساوندی مرز بین واژه و جمله مشخص نیست. به عبارت دیگر، در این گونه زبان‌ها بسیاری از مفاهیم که معمولا در زبان‌های دیگر از طریق جمله بیان می‌شوند، چه قاموسی و چه نحوی، از طریق فراهم آوردن تعداد زیادی واژک به صورت یک واژه بیان می‌شوند. این دسته خود به دو زیر دسته "تک پایه ای"

---

[۱] Mono Morphemic

[۲] Poly Morphemic

[۳] Agglutinative

[۴] Synthetic / Fusional

[۵] Polysynthetic

(مانند "یوپیک" در سیبری) و "چند پایه‌ای" (مانند "چوک‌چی" در شمال شرق سیبری) تقسیم می‌گردد.

بعضی از زبان‌ها مانند فارسی و انگلیسی، گرچه بیش‌تر دارای ویژگی‌های زبان‌های پیوندی هستند، اما ویژگی‌هایی از زبان‌های ترکیبی نیز در آن‌ها مشاهده می‌شود. برای مثال کلمه‌ی "مردانگی‌هایی" (mard - āne – gi – hā- – ?i) در فارسی نمایان‌گر خصوصیت پیوندی بودن این زبان است. در حالی که "شناسه‌های فعلی[۱]" مفاهیم شخص و شمار، هر دو را در بر می‌گیرند که تفکیک‌ناپذیرند. "

6-2. تعریف‌ها

6-2-1. صرف (ساخت‌واژه)

"صرف" بخشی از دستور زبان است که ساخت‌واژه[۲] را مورد تحلیل قرار می دهد. ساخت‌واژه دو لایه تصریفی[۳] و اشتقاقی[۴] وجود دارد. در صرف تصریفی به بخشی از واژه پرداخته می شود که مستقیماً با نقش نحوی آن سر و کار دارد. در حالی که در صرف اشتقاقی به آن بخش از ساخت‌واژه توجه می شود که به خود واژه، فارغ از نقش نحوی آن، تعلق دارد. در این‌جا منظور از صرف، صرف تصریفی می باشد.

6-2-2. واژه

[۱] Personal Ending|

[۲] Morphology

[۳] Inflectional

[۴] Derivational

واژه را می‌توان از چهار نظر 1. آوایی 2. ساخت صرفی 3. معنایی یا 4. املایی تعریف کرد. در تعریف آوایی، واژه را به لحاظ ساخت آوایی (هجاهای تشکیل دهنده، تکیه، درنگ) مد نظر قرار می‌دهند. در تعریف معنایی، یک واژه، یک واحد معنایی است که بر یک یا چند مفهوم منفرد دلالت می کند. به لحاظ املایی، یک واژه دارای وحدت املایی است. (البته این مسأله برای زبان فارسی در حوزه کامپیوتر و نشریات اتفاق نیفتاده است و یکی از مشکلاتی است که در این فصل بررسی خواهد شد) تعریف واژه به لحاظ ساخت صرفی که عمدتاً مد نظر ماست: واژه از یک یا چند واژک تشکیل شده و در سلسله‌مراتب دستوری زبان در ساختمان "گروه[1]" به کار می رود.

6-2-3. واژک

واژک کوچکترین واحد معنی‌دار یا نقش‌دار زبان است که در سلسله‌مراتب واحدهای دستوری زبان در ساختمان واژه به کار می رود. واژک مفهوم انتزاعی دارد، تظاهر عینی آن را "واژ[2]" و گونه‌های آن را "واژگونه[3]" نامیده‌اند. وقوع واژگونه‌ها مشروط به بافت است؛ یعنی در بافت‌های متفاوت، صورت های متفاوتی از آن ها ظاهر می شود.

واژک ها به دو دسته‌ی "واژک آزاد[4]" و "واژک مقید[5]" تقسیم می‌گردد. واژک آزاد می‌تواند به تنهایی و به طور مستقل به کار رود که در این صورت یک واژه بسیط محسوب می شود (مانند کار، مهر) واژک مقید به طور مستقل به کار نمی‌رود و به واژک‌های دیگر می‌چسبد (مانند "مند" در "کارمند"، "بان" در "مهربان").

6-2-4. وند و پایه

---

[1] Phrase

[2] Morph

[3] Allomorph

[4] Free Morpheme

[5] Bound Morpheme

کلمه از ترکیب "وند[1]" (واژک مقید) و "پایه[2]" (معمولاً واژک آزاد است، ولی بعضی از پایه‌ها که به طور مستقل به کار نمی‌روند که به آن‌ها "پایه مقید" گفته می شود) تشکیل می‌شود.

وندها خود به دو دسته "وند تصریفی" و "وند اشتقاقی" تقسیم می‌شود. وندهای تصریفی نقش نحوی دارند، یعنی کلمه را برای ایفای نقش معینی در ساخت‌های نحوی پردازش می‌دهند و کاربردشان قیاسی است. به عبارت دیگر می‌توانند با همه‌ی اعضای یک مقوله از کلمات به کار روند و کم‌تر استثناء می‌پذیرند. از طرف دیگر وندهای اشتقاقی نقش واژه‌سازی دارند (واژه‌ی جدید را می‌سازند) و کاربردشان سماعی است. به عبارت دیگر با همه‌ی اعضای یک مقوله از کلمات به کار نمی‌روند. تعداد وندهای اشتقاقی از تعداد وندهای تصریفی بیش‌تر است و از نظر محل قرار گرفتن، در مقایسه با وند های تصریفی، به پایه کلمه نزدیکترند و غالباً مقوله کلمه را تغییر می‌دهند. [3]

6-2-5. واژه‌بست

واژه‌بست[4] یکی از انواع کلمه به شمار می رود که کاربرد مستقل ندارد. مانند وندها به کلمه قبل یا بعد از خود می‌چسبد، ولی برخلاف وندها جزء ساخت کلمه محسوب نمی‌شود.

گروهی که به کلمه قبل خود می پیوندند "پی‌بست" و گروهی که به کلمه بعد خود می‌چسبند "پیش بست" نامیده می‌شوند. واژه‌بست‌های فارسی، همگی پی‌بست می‌باشند.

6-2-6. ادات

[1] Affix

[2] Base

[3] بعضی از کلمات به مرور زمان علاوه بر معنی اصلی خود، معنی ثانوی پیدا می کنند و با معنی جدید در کلمات بسیط به کار می روند، به این دسته از کلمات "شبه وند" می گوییم. (مانند "شاه" در شاهراه") شبه‌وندها ممکن است به تدریج معنی اصلی خود را از دست بدهند؛ به عبارت دیگر به طور آزاد به کار نروند که در این صورت تبدیل به وند می‌شوند.

[4] Clitic

"ادات" کلماتی دستوری هستند که به طور مستقل به کار نمی‌روند. این نوع کلمات را، به خلاف سایر کلمات، دقیقاً نمی‌توان در یکی از طبقات "انواع کلمه" قرار داد، مانند پیش‌وندهای فعلی در فارسی، to (نشانه مصدر)، not (نشانه نفی فعل)، و up در call up در زبان انگلیسی.

6-3. زبان فارسی

زبان فارسی یک از زبان‌های دسته هند-اروپایی است که به طور اصلی در ایران، افغانستان و بخش‌هایی از تاجیکستان خوانده و نوشته می‌شود. همانند انگلیسی، زبان فارسی یک مورفولوژی وندافزایی دارد. به عبارت دیگر، از پس‌وندها و پیش‌وندها برای تغییر معنایی کلمه استفاده می‌شوند (قابل ذکر است که در زبان فارسی میان‌وند وجود ندارد).

زبان فارسی از الفبای عربی (بعلاوه حروف "گ"، "چ"، "پ"، و "ژ") استفاده کرده و به صورت راست به چپ نوشته می‌شود. کاراکترها در یک کلمه اغلب به یکدیگر می‌چسبند و اغلب کاراکترها شکل‌های مختلفی بسته به موقعیت‌شان دارند. نوشتار فارسی اجازه می‌دهد تا واژک[1]های معینی به صورت وندهای آزاد و یا به صورت واژک‌های پیشین ظاهر شوند. برای نمونه "می" می‌تواند به صورت چسبیده و یا جدا نشان داده شود ("می شود" یا "میشود") [105]. از طرف دیگر در رسم‌الخط فارسی بعضی از کلمات به طور ذاتی جدا از یکدیگر نوشته می‌شوند. (برای مثال "بین المللی"). این مسأله باعث می‌شود تا یافتن مرز کلمات دشوار گردد.

علاوه بر مورد قبلی، یکی دیگر از مشکلات در زبان فارسی عدم نوشتن واکه‌ها در نوشتار می‌باشد. که این مسأله خود مشکل هم‌نویسه‌ها و ابهام در تشخیص ویژگیها را افزایش می‌دهد برای مثال "مرد" با تلفظ "mard" و کلمه "مرد" با تلفظ "mord" همانند یکدیگر نوشته می‌شوند.

قابل ذکر است که افعال مرکب در زبان فارسی معمولاً با استفاده از هم‌کردهایی همچون "خوردن"، "کردن"، "دادن" و "زدن" تولید می‌شود که معنای آن‌ها با یکدیگر به طور کامل دگرگون می‌کند. برای مثال "زمین خوردن" یا "کتک خوردن" که در این‌جا هم مشکل تشخیص مرز کلمه و

[1] Morpheme

استخراج ویژگی مطرح می‌باشد و هم‌کرد "خوردن" در معنای غیر اصلی خود به کار رفته است و مشکل ابهام را نیز متعاقباً با خود به همراه دارد.

همانند انگلیسی اسم‌ها در زبان فارسی می‌توانند جمع شوند. اما از سویی دیگر افعال در زبان فارسی بسیار پیچیده تر هستند. فعل‌ها در زبان فارسی می‌توانند به لحاظ زمانی، شخص، شمار (جمع یا مفرد بودن)، و منفی شدن صرف شوند. بنابراین، یک فعل داده شده ممکن است ترکیبی از آن‌ها را با یکدیگر را دربرداشته باشد. برای مثال به "دیدمش" به معنای "من او را دیدم" می‌باشد. معمولا فعل‌هایی که از بن ماضی استفاده می‌کنند برای زمان حال یا آینده می‌باشند و فعل‌هایی که بن مضارع در آن‌ها بکار رفته است برای زمان حال می‌باشند.

جدول 2 مصدرهای باقاعده و بن فعل

حالت با قاعده			
بن مضارع	**وند**	**بن ماضی**	**مصدر**
آور، خور	د	آورد، خورد	آوردن، خوردن
	ت		
افت	اد	افتاد	افتادن
کش، بر، پر	-ِید	کشید، برید، پرید	کشیدن، بریدن، پریدن

در زبان فارسی، بخشی از کلمات معمولاً از بن فعل ساخته می‌شوند. لذا از نقطه نظر دستوری، اولین قدم در استخراج ریشه، یافتن مقوله‌ی واژگانی[1] کلمه است. برای مثال می‌توان به ریشه‌ی کلمه "شنونده" را با حذف پس‌وند "نده" رسید. در حالت کلی، یافتن حالت دستوری کلمه به دلیل وجود حالت‌های بی قاعده دشوار است. مصدرهای باقاعده در زبان فارسی به "-َن" ختم می‌شود و

[1] Part Of Speech (POS)

بن ماضی فعل در حالت باقاعده با حذف "ن" از مصدر قابل بازیابی است. همین‌طور در حالت‌های باقاعده با حذف "د"، "ت"، "اد"، و "-ِید" از بن ماضی، به بن مضارع می‌رسیم.

بن مضارع + ("د"، "ت"، "اد"، "-ِید") = بن ماضی

بن ماضی + "-َن" = مصدر

حالت‌های باقاعده و الگوهای بازیابی بن ماضی تحت عنوان "قیاسی" شناخته می‌شود. حالت‌های بی‌قاعده‌ای که (مانند "گفتن"، "زدن"، "دیدن"، "رفتن"، "شنیدن") معمولاً هیچ الگوی خاصی برای رسیدن به ریشه آن‌ها وجود ندارد. مصدرهای بی‌قاعده معمولاً بر مبنای شنیدن می‌باشند و بدین سبب به آن‌ها "سماعی" می‌گویند. مثال‌هایی از مصدرهای باقاعده و بن‌های ماضی و مضارع‌شان در جدول 2 آمده است.

6-3-1. پیش‌وندهای تصریفی در زبان فارسی

در فارسی، به طور معمول با اضافه نمودن پیش‌وند "ن" برای منفی کردن فعل‌ها استفاده می‌شود. برای مثال "نرو" حالت منفی فعل امر از ریشه "رو" و مصدر "رفتن" و "نشنید" حالت منفی از گذشته ساده از ریشه "شنو" و مصدر "شنیدن" هستند. مشابه این پیش‌وند "م" می باشد ولی با این تفاوت که بعد از این پیش‌وند حرف "م" نمی‌تواند بیاید.

پیش‌وند "ب" برای ساختن فعل امری مثبت یا مضارع التزامی بکار می‌رود. برای مثال می‌توان "ب" در فعل امر "برو"، "بزن"، "بخور" و بگو" اشره کرد که پیش‌وند "ب" به بن مضارع "رو"، "زن"، "خور" و "گوی" اضافه شده است. متقابلاً برای مضارع التزامی نیز می‌توان به "بروم"، "بزنم"، "بخورم" و "بگویم" اشاره نمود.

یکی دیگر از پیش‌وندهایی که در زبان فارسی استفاده می‌شود، پیش‌وند "می" می‌باشد. این پیش‌وند قبل از ریشه فعل می‌آید و هم در ماضی و هم در مضارع کاربرد دارد. در ماضی برای ماضی استمراری و برای زمان حال برای مضارع اخباری بکار می‌رود. در حالت منفی "می" بین "ن" و ریشه فعل قرار می‌گیرد، لذا برای این حالت خاص می‌توان فرض نمود که پیش‌وند "نمی" برای حالت منفی بکار می‌رود. (جدول 3)

جدول 3 لیست پیش‌وندهای تصریفی عمده در زبان فارسی. در این‌جا بن فعل = (بن ماضی / بن مضارع)

نوع ترکیب	فرمول
فعل امر مثبت / مضارع التزامی	"ب" + بن فعل
فعل امر منفی / ماضی ساده منفی	"م" + بن فعل
فعل امر منفی / ماضی ساده منفی	"ن" + بن فعل
ماضی استمراری / مضارع اخباری	"می" + بن فعل
ماضی استمراری منفی / مضارع اخباری منفی	"ن" + "می" + بن فعل

در زبان فارسی پیش‌وندهای اشتقاقی نسبتاً زیادی وجود دارد. پیش‌وندهای اشتقاقی معمولا به دو دسته "پیش‌وندهای فعلی" و "پیش‌وندهای غیر فعلی" تقسیم می‌شوند. به نمونه‌ای از این پیش‌وندهای فعلی می‌توان به "بر" در "برکشید"، "در" در "درآمیخت" و "فرو" در "فروخورد" یا "فروکرد" اشاره نمود. هم‌چنین برای پیش‌وندهای غیر فعلی می‌توان به "هم" در "همکلاسی"، "ب" در "بهنجار"، و "بی" در "بی‌غم" یا "بی‌درد" اشاره نمود. اما دسته پیش‌وندهای اشتقاقی قطعاً معنای کلمه را تغییر می‌دهند و حتی در مورد غیر فعلی‌ها می‌توانند طبق دستوری کلمه را عوض کنند. برای مثال کلمه "هنجار" اسم می‌باشد و با اضافه نمودن پیش‌وند "ب" تبدیل به صفت می‌گردد.

6-3-2. واژه‌بست‌ها و پس‌وندهای تصریفی در زبان فارسی

در این قسمت واژه‌بست ها و پس‌وندهای تصریفی در زبان فارسی به اجمال مورد بررسی قرار می‌گیرد. پس‌وند "تر" و "ترین" برای "صفت"ها قابل استفاده هستند (برای مثال "بزرگتر" و "بزرگترین"، "نزدیکتر" و "نزدیکترین"، "کوتاه‌تر" و "کوتاه‌ترین" اشاره نمود). پس‌وند "تر" برخلاف پس‌وند "ترین" برای قید نیز قابل استفاده می‌باشد (برای مثال "تر" در "تندتر": آن مرد نمی‌تواند تندتر کار کند).

فرض کنید، بتوان به بن فعل رسید، آن‌گاه می‌توان با پیروی از قوانین زبان فارسی یک فعل را در حالت‌های مختلف برای نمونه مضارع اخباری صرف نمود. برای مثال برای فعل "رفتن"، می‌توان فعل امر مثبت "برو" و با اضافه‌کردن پس‌وندهای تصریف حال آورده شده در جدول 4 در حالت مضارع التزامی صرف نمود.

جدول 4 پس‌وندهای تصریف مضارع التزامی

فعل	پس‌وند	زمان	فعل	پس‌وند	زمان
بروم	م	اول شخص مفرد	برویم	یم	اول شخص جمع
بروی	ی	دوم شخص مفرد	بروید	ید	دوم شخص جمع
برود	د	سوم شخص مفرد	بروند	ند	سوم شخص جمع

قوانین تصریف حال برای تولید دیگر تصریف‌ها نیز استفاده می‌شود. به طریق مشابه برای تصریف فعل در زمان گذشته می‌توان با حذف "ن" از انتهای مصدر و اضافه نمودن پس‌وندهای فوق به استثناء "سوم شخص مفرد" می‌باشد. در حالت گذشته سوم شخص مفرد پس‌وند ندارد. برای مثالی از مصدر "رفتن" به جدول 5 مراجعه شود.

جدول 5 پس‌وندهای تصریف گذشته

فعل	پس‌وند	زمان	فعل	پس‌وند	زمان
رفتم	م	اول شخص مفرد	رفتیم	یم	اول شخص جمع
رفتی	ی	دوم شخص مفرد	رفتید	ید	دوم شخص جمع
رفت	-	سوم شخص مفرد	رفتند	ند	سوم شخص جمع

برای جمع در در حالت با قاعده در زبان فارسی از پس‌وند "ها" و "ان" استفاده می‌شود و برای کلمات عربی از پس‌وند "ات"، "ون" و "ین" استفاده می‌گردد. اگرچه این علامت‌های جمع برای کلمات زبان عربی بوده ولی به طور محدود برای تعدادی از کلمات خاص نیز استفاده می‌شوند. این کلمات در زبان فارسی اغلب به معنای غیر از معنای جمع حالت مفرد خود دارند. برای نمونه می‌توان به "تبلیغات" و "تاسیسات" اشاره نمود که در فارسی در معنای جمع "تبلیغ" و "تاسیس" بکار نمی‌روند. یکی دیگر از پس‌وندهای جمع "جات" می‌باشد. باید ذکر کرد که در زبان فارسی این پس‌ند به مفهوم انواع یک چیز استفاده می‌شود. برای مثال منظور "سبزیجات" یا کارخانه‌جات"، مقدار زیادی از سیز یا تعداد زیادی کارخانه نیست، بلکه در این‌جا منظور انواع سبزی و انواع کارخانه می‌باشد. (جدول 6)

جدول 6 علامت‌های جمع در زبان فارسی

اسم در حالت جمع	پس‌وند
پسرها، کتاب‌ها، خانه‌ها	ها
جوانان، دانش‌آموزان، درختان	ان
مشکلات	ات
معلمین، محصلین	ین
روحانیون	ون
میوه‌جات، کارخانه‌جات، سبزی‌جات	جات

دسته‌ای دیگر از پس‌وندها شامل "م"، "ت"، "ش"، "مان"، "تان"، و "شان" می‌باشند. این پس‌وندها بین فعل و اسم مشترک می‌باشند. (دیدمت، خوردمش، کتابم، کلاس‌مان) این وندها برای اسم‌ها نقش اضافی را بازی می‌کنند و ترکیب مضاف-مضاف الیه‌ای می‌سازند و برای فعل‌ها نقش ضمائر مفعولی را ایفا می‌کنند. برای مثال در "کتابش"، "کتاب" اسم می‌باشد و "ش" مضاف الیه آن است. در "دیدمش"، "دیدم" فعل ماضی ساده بوده و "ش" نقش ضمیر مفعولی را بازی می‌کند (جدول 7).

جدول 7 پس‌وندهای "م"، "ت"، "ش"، "مان"، "تان"، و "شان" در نقش اضافی و مفعولی

پس‌وندها در نقش اضافی					
اسم	**پس‌وند**	**نتیجه**	**اسم**	**پس‌وند**	**نتیجه**
برادر	م	برادرم	برادر	مان	برادرمان
برادر	ت	برادرت	برادر	تان	برادرتان
برادر	ش	برادرش	برادر	شان	برادرشان
پس‌وندها در نقش مفعولی					
فعل	**پس‌وند**	**نتیجه**	**فعل**	**پس‌وند**	**نتیجه**
ببرند	م	ببرندم	ببرند	مان	ببرندمان
ببرند	ت	ببرندت	ببرند	تان	ببرندتان
ببرند	ش	ببرندش	ببرند	شان	ببرندشان

در اسم‌ها علامت‌های جمع نسبت به ضمائر اولویت دارند. برای مثال می‌توان به "کتاب‌هایشان" ("کتاب" : اسم + "علامت جمع :ها + "ی" صامت میانجی + ضمیر اضافی : "شان") و هم‌چنین "کارخانه‌جاتشان" ("کارخانه" : اسم + علامت جمع : "جات" + ضمیر اضافی : "شان") اشاره نمود.

فعل + ضمائر متصل (نقش مفعولی)

اسم + ضمائر متصل (نقش اضافی)

اسم + علامت جمع + ["ی" صامت میانجی] + ضمائر متصل

یکی دیگر از وندهای تصریفی علامت "ی" نکره می‌باشد. برای مثال وند "ی" در "مردی": مردی را دیدم. در این مثال وند "ی" یک علامت نکره می‌باشد. باید توجه داشت که "ی" نکره هیچگاه با ضمائر که به ذات معرفه هستند نمی‌آید. قابل ذکر است "ی" در کتابهایشان، "ی" صامت میانجی

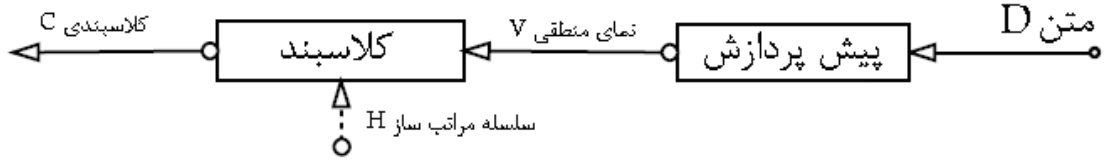
می‌باشد که برای جلوگیری التقاط دو صوت بکار می‌رود. به عبارت دیگر بین دو مصوت بلند از "ی" صامت میانجی استفاده می‌کنند.

در حالت کلی برای اسامی، معمولا وندهای اشتقاقی نسبت به وند تصریفی به ریشه اسم نزدیکتر هستند. استثناء موجود در این‌جا وند جمع کننده "ات" می‌باشد. برای مثال می‌توان به "دهاتی" و تبلیغاتی" اشاره نمود که "ی" اشتقاقی پس از پس‌وند جمع کننده "ات" ظاهر شده است.

اسم + وند اشتقاقی + وند تصریفی

7. سیستم دسته‌بندی خودکار متون فارسی

مسأله‌ی دسته‌بندی آماری متون با سه مشکل اساسی مواجه است. مشکل اول عبارت است از: مقیاس کاربردهای دسته‌بندی متون برای بسیاری از الگوریتم‌های یادگیری ماشین استاندارد ایجاد مشکل می‌کند. متونی که با استفاده از بردارهایی که با مقادیر عددی (با در نظر گرفتن یک عدد برای هر کدام از کلمه ها) نمایش داده می‌شوند، می‌توانند به ابعادی در حدود 5^10- 6^10 یا بیش‌تر برسند. ابعاد بالا علاوه بر مشکل زمان زیاد پردازش، می‌تواند خطر بیش یادگیری[1] را با خود به همراه داشته باشد [2]. مشکل دوم به مدل‌سازی و یادگیری فهم انسانی از مسأله دسته‌بندی متون باز می‌گردد. برای مثال، معمولاً به طور عام افراد نسبت به کلمات احساسات متفاوتی دارند که تخمین مناسبی برای دسته‌بندی می‌باشد [2]. مشکل سوم، کلمات به تنهایی در کاربردهای مختلف می‌تواند معانی متفاوتی از خود بروز دهد و یا در عبارات وابستگی زیادی به کلمات قبلی و بعدی داشته باشند. اصولاً، مسأله‌ی دسته‌بندی متون از ماژول عمده پیش‌پردازش و کلاس‌بند تشکیل می‌شود که بسته به کاربرد می‌تواند از ابزارهای درخت‌ساز و یا کلاس‌بندهای سلسله‌مراتبی برای ایجاد نتیجه به صورت سلسله‌مراتبی بهره گیرد (شکل 4).



شکل4 ساختار ماژولار مسأله دسته‌بندی متون

ماژول پیش‌پردازش، یک روش رمز کردن متن می‌باشد که به طور خودکار نمای منطقی از متن می‌سازد [15]. اصول پایه در رمز کردن شامل: 1- جوابگو بودن در برابر تفسیر ارائه شده از طرف الگوریتم‌های استنتاج کلاس‌بند 2- در بر داشتن معنای متن به طور فشرده، تا امکان‌پذیری و انعطاف را با خود به همراه داشته باشد [106].

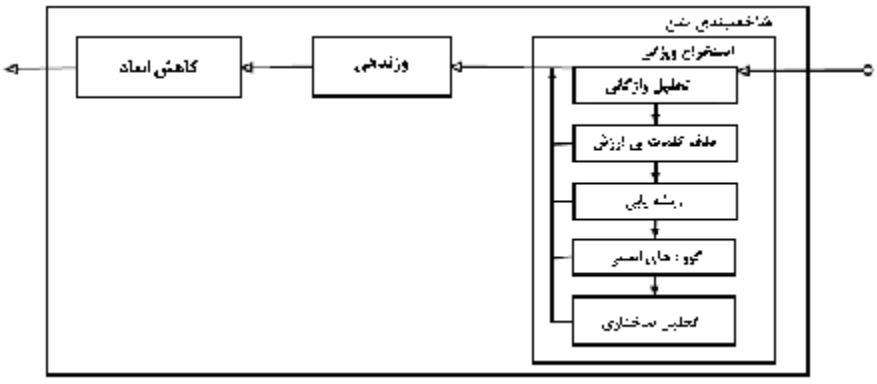
در مابقی این نوشتار سعی شده است تا یک سیستم ماژولار دسته‌بند متون با اجزا تشکیل‌دهنده آن پیشنهاد شود. با توجه به اینکه این سیستم اجزای نسبتاً زیادی دارد، در این پروژه اجزای بسیار

[1] Overfitting

ضروری آن پیاده‌سازی شده است و برای دیگر اجزای آن به مثابه یک ماژول بسیار ساده دیده شده است. این مسأله امکان می دهد تا در آینده بتوان با ارتقای آن ماژول‌ها و کیفیت سیستم را توسعه داد و هم‌چنین جامعیت مورد نظر برای سیستم نیز حفظ گردد.

7-1. پیش‌پردازش

در حقیقت، پیش‌پردازش وظیفه نگاشت متن داده شده به یک نمای منطقی را بر عهده دارد. به عبارت دیگر استخراج ویژگی و وزن‌دهی و کاهش ابعاد در این قسمت انجام می‌گیرد. بسته به کاربرد استخراج ویژگی می‌تواند بسیار ساده و یا بسیار مفصل باشد. تحلیل واژگانی شامل عملیات مربوط به یکسان سازی متن، قواعد مربوط به نشانه‌گذاری‌ها و مرزبندی بین کلمات می‌باشد. بعد از این مرحله عموما دسته‌ای از کلمات بی ارزش که متناوبا تکرار می‌شوند و بار معنایی خاصی ندارند: مانند حرف ربط ("و"، "که"، "تا"، "وقتیکه"، "اگر"، "اما"، "اینکه")، حرف اضافه ("به"، "با"، "از"، "در")، فعل ربطی ("است"، "بود"، "شد") و حرف تعریف ("یک" در "یک دانشجوی نمونه کسی است که ...") از متن داده شده حذف می‌شوند. سپس با استفاده از الگوریتم‌های ریشه‌یابی، به منظور بهینه‌سازی ویژگی‌های استخراج شده، کلمات ریشه‌یابی می‌شوند. در نهایت با استفاده از گروه‌های اسمی کلمات دسته‌بندی می‌گردند. تحلیل‌های ساختاری بیش‌تر به اطلاعات سطح بالاتر همچون پاراگراف‌بندی بر می‌گردد. کلمات و اطلاعات استخراج شده برای وزن دهی به قسمت بعدی ارسال می‌شود. یکی از ساده ترین راه‌های برای کاهش ابعاد ویژگی‌ها در قسمت بعد، می‌تواند حذف ویژگی‌هایی باشد که وزن آن‌ها از حد معینی کم‌تر است (شکل 5).



شکل 5 زیر ماژول‌های قسمت پیش‌پردازش و ارتباط آن‌ها با یکدیگر

7-1-1. تحلیل واژگانی

متون الکترونیکی فارسی موجود به صورت خام دارای مشکلات ذیل می باشند:

۱. عدم وجود رسم‌الخط یکسان: متأسفانه متون فارسی موجود از یک رسم‌الخط واحد پیروی نمی‌کنند. یکی از مهم‌ترین این دلایل متداول بودن بعضی از حالت های چسبیدن وندهایی همانند "ها" به اسم ها در زبان فارسی می باشد. برای مثال "کتاب ها"، "کتاب‌ها"، "کتابها" .

۲. وجود کدهای متفاوت برای حروف فارسی: اگرچه سیستم یونی‌کد سعی در یکسانی سازی کدها برای کلیه زبان‌ها نموده است. ولی با این حال در مواردی برای پشتیبانی از کلیه حالت ها ناچار به استفاده از کدهای بعضا متفاوت برای یک حرف نموده است (در "ضمیمه‌ی الف" بخش یونی‌کد مربوط به زبان فارسی و عربی آورده شده است).

۳.

7-1-2. کلمات بی‌ارزش

کلمات بی ارزش[1] به کلماتی که هیچ گونه ارزش مفهومی به لحاظ طبقه‌بندی ندارند و بیش‌تر شامل حروف اضافه، ضمایر ملکی و افعال ربطی می‌گردد، می‌گویند. این دسته از کلمات باعث ایجاد ابهام در پروسه دسته‌بندی می‌گردد. به علاوه به‌کارگیری آن‌ها باعث صرف منابع پردازشی و حافظه گردیده و در مقابل هیچ کمکی به دقت الگوریتم نمی نماید (جدول 8).

[1] Stop Words

جدول 8 لیست کلمات بی ارزش

کلمات بی ارزش
امروز، گفتم، اکنون، خواهند، آر، آقا، آقای، آقایان، آمد، آمده، آن، آنان، آن‌جا، آنچه، آنکه، آن‌ها، آیا، اخیر، از، است، اسلامی، اش، افزود، اگر، اگرچه، الا، البته، الی، ام، اما، امروزه، اند، اندی، او، اولین، ای، ایران، ایشان، ایم، این، این‌جا، اینکه، این گونه، با، بااین، بااینکه، بار، باز، باشد، باشید، باشیم، بالاخره، باید، بجز، بدهید، بدون، بر، برای، براین، برخی، برلزوم، بسیار، بسیاری، به طور، بعد، بکنید، بگذاریم، بگوییم، بلکه، بماند، به، بود، بودند، بوده، بی، بیش، بین، پس، پی، پیش، تا، تر، تری، تمامی، تو، توسط، توی، جا، جز، جمهوری، چرا، چنان، چند، چنین، چه، چو، چون، چونکه، حال، حالی، حالی که، حتی، حدود، حقیقتا، خانم، خانمها، یگزاری، خواهد، خود، خودم، خودمان، خویش، داخل، داد، دادم، دادند، داده، دار، دارای، دارد، دارند، داریم، داشت، داشته، داند، دانند، در، درآن، دراین، درباره، دربر، دربعد، دربین، درپی، درجای، درحال، درحالی، درحالی که، دردو، درکل، درین، دور، دیگر، را، رسیده، رو، روی، زدند، ساعت، سر، سعی، سو، سوی، شامل، شد، شدن، شدند، شده، شما، شود، طی، علیرغم، علیه، غیر، فقط، کرد، کردم، کردن، کردند، کرده، کنار، کند، کنم، کنند، کنید، کنیم، که، گذاری، گرچه، گردند، گرفت، گرفته، گفت، گفتند، گفته، لزوم، ما، مانند، متوالی، مثلا، من، می، می‌شود، میان، میتواند، میخواهیم، میداند، میرسد، میشود، میکنم، میکنند، ندارد، ندارم، ندارند، نداشته، نشدند، نظر، نماید، نموده، نمی، نمیکند، نیز، نیست، نیستند، ها، های، هایی، هر، هریک، هست، هستم، هستند، هستید، هستیم، هم، همان، همه، همین، هنوز، هیچ، و، وجود، ولی، وی، یا، یافت، یعنی، یکدیگر، یکم، یکی، یم

7-1-3. ریشه‌یابی

از نمونه کاربردهای ریشه‌یاب‌ها، می‌توان به استفاده از الگوریتم‌های ریشه‌یاب برای یادگیری بدون ناظر گرامر [107] و یا حذف وندها برای و کاهش ابعاد در مسائل بازیابی اطلاعات می‌توان اشاره

کرد. امروزه تقریبا برای اکثر زبان‌ها از انگلیسی،فرانسوی تا چینی و ژاپنی و حتی عربی و فارسی نیز ریشه‌یاب‌هایی تهیه شده است [112][111][110][109][108].

از مشکلات ریشه‌یابی می‌توان به بیش‌ریشه‌یابی (تولید ریشه‌هایی که هیچ معنایی در زبان ندارند) و کم ریشه‌یابی (عدم امکان تولید ریشه برای حالت‌های استثناء) می‌توان اشاره نمود [118]. در حقیقت یافتن حداکثر پس‌وندهای چسبیده به کلمه برای تشخیص مقوله‌ی واژگانی کلمه بسیار مفید است[107].

فاکتورهای اصلی برای ارزیابی یک ریشه‌یاب در بازیابی اطلاعات شامل نوع الگوریتم ریشه‌یابی، معیارهای ارزیابی دقت و بازیابی موفقیت آمیز، وابستگی یا عدم وابستگی به زبان، هزینه زمانی و پردازشی، طول متن، طول کلمات و دیگر معیارهای ممکن می باشد [113]. همان‌طور که پایس ذکر کرد در حالت کلی ریشه‌یاب‌های پیچیده از معیار بازیابی بهتر و ریشه‌یاب‌های ساده از دقت بهتری برخوردارند[111][114] . باید ذکر کرد که استفاده از یک لغت‌نامه کمک بسیار زیادی به بهبود نتایج می نماید.

7-1-3-1. کارهای قبلی در زمینه ریشه‌یابی

روش‌های عمده در ریشه‌یابی شامل مواد ذیل می‌گردد:

- **روش‌های آماری (روش‌های اتوماتیک)**

در این روش از ویژگی های آماری کلمات بدون استفاده از منابع انسانی برای استخراج دانش مورفولوژی بهره می گیرد [107]. معمولا در این دسته از الگوریتم‌ها تلاش می شود تا قابلیت تعمیم الگوریتم و سرعت اجرای روش افزایش یابد [110].

۱- *تحلیل کم‌ترین فاصله توصیفی*؛ ابتدا یک مدل توصیفی برای مورفولوژی تعریف شود و سپس با مقایسه این مدل با داده‌های نمونه این فاصله توصیفی را کمینه می‌نماید [110][115].

۲- *روش ترکیب چند ریشه‌یاب ساده*؛ ابتدا تعدادی ریشه‌یاب ساده [107][116] با استفاده از قوانین ابتکاری آماری ساده از هم نشینی و هم-رخدادی کلمات تهیه می شود و سپس ریشه‌یاب نهایی بر اساس ترکیب آن‌ها با یکدیگر به دست می‌آید [111].

از مزایا این دسته از روش‌ها می‌توان به موارد زیر اشاره کرد:

- کم‌هزینه بودن پیاده‌سازی الگوریتم به دلیل عدم نیاز به هیچ دانش زبانی از پیش
- چندزبانه بودن الگوریتم

از مشکلات این دسته از روش‌ها می‌توان به مورد ذیل اشاره کرد:

- از مشکلات این دسته تهیه یک پیکره به نحوی که به تعداد کافی از حالت‌های یک کلمه و به دفعات تکرار شده باشد می باشد. برای مثال زبان‌های سامی در این روش معمولاً دقت خوبی ندارند. بر طبق [117] از پردازش توزیعی روزنامه‌ای، کلمات عربی در در مقایسه با انگلیسی در اندازه مشابه بسیار منحصر به فرد رخ می‌دهند که به دلیل ناکافی‌بودن برای روش‌های آماری، کیفیت الگوریتم را پایین می‌آورد.

- **روش‌های ریشه‌یابی زبانی**

این روش معمولا مبتی بر زبان بوده و در این روش لیست کلیه وندها و قوانین تصریفی زبان به صورت دستی در سیستم لحاظ می‌گردد. معروف ترین روش در این دسته الگوریتم های Porter می باشد.[119][118] که ابتدا برای زبان انگلیسی مطرح گردید ولی هم‌اکنون برای بیش‌تر زبان‌های اروپایی (فرانسوی، آلمانی، ایتالیایی) [110] و حتی فارسی نیز پیاده‌سازی شده است [112].

مزایا: [118]

- این دسته از روش‌ها از تعداد خطاهای نسبی بسیار کم‌تری در مقایسه با دیگر روش‌ها دارد.
- این روش‌ها از سادگی نسبی خوبی در پیاده‌سازی برخوردارند.

معایب:

- وابستگی به زبان

7-1-3-2. کارهای قبلی در زمینه‌ی زبان فارسی

تاکنون مهم‌ترین کارهای انجام شده و مشاهده شده توسط نگارنده، برای ریشه‌یابی کلمات در زبان فارسی به چهار کار محدود شده است:

۱. آقای ریاضتی در [14] بر مبنای یک سیستم مورفولوژی دو سطحی برای تحلیل‌های صرفی و اشتقاقی استفاده می نماید. مدل Peslex ارائه شده با الهام از مدل Englex که در [15] ارائه شده تهیه شده است.

۲. خانم مگردومیان در [6] سعی کرده است تا بر مبنای الگوریتم پورتر[۱] یک ماشین حالت متناهی تنها برای تحلیل صرفی واژگان ارائه نماید.

۳. آقای کاظم تقوا و همکارانش در [120] سعی کرده‌اند که به ارائه یک ریشه‌یاب بر مبنای الگوریتم پورتر بپردازند. این ریشه‌یاب به لحاظ کلی خوب ولی ناکافی بوده است.

7-1-3-3. ریشه‌یاب پیشنهادی

همان‌طور که قبلاً ذکر شد، منظور و هدف از ریشه‌یابی در دسته‌بندی متون، افزایش دقت سیستم دسته‌بندی متون می باشد و نه یک اقدام زبان‌شناختی صرف. بنابراین ملاحظات متعددی در این مسأله مطرح می‌گردد که اعم آن‌ها شامل موارد ذیل می باشد:

- افزایش دقت: همان‌طور که پیش‌تر اشاره شد، ریشه‌یابی یکی از مجموعه اعمالی است که با کم کردن فضای ابعاد، تنوع ویژگی ها را کاهش می دهد. در کل، این امر ممکن است در مواردی (مانند بیش ریشه‌یابی یا ریشه‌یابی اشتقاقی) منجر به کاهش دقت سیستم دسته‌بندی متون گردد. برای مثال، فرض کنید که "خاتمی" به "خاتم" ریشه‌یابی شود، در

[۱] Porter Algorithm

حالی که بدون ریشه‌یابی خود این ویژگی یک نشانه خوب برای متن‌های دسته‌ی سیاسی می‌توانست باشد.

- افزایش سرعت: باید در نظر گرفت که زمان مصرفی درکل سیستم به عامل‌های زیادی بستگی دارد. اگر چه ریشه‌یابی منجر به کاهش ابعاد و در نتیجه سرعت می‌گردد، ولی یک ریشه‌یاب سنگین خود می‌تواند منجر افزایش کل سیستم شود. به طوری که ریشه‌یابی ارزش خود را از دست بدهد.
- حداقل سربار محاسباتی و حافظه‌ای: ریشه‌یاب‌ها بسته به دربر داشتن واژگان و لیست ریشه ها و همین‌طور پروسه‌های تصحیح برای حالت های استثناء، خود می‌توانند سربار زیادی بر سیستم تحمیل کنند.

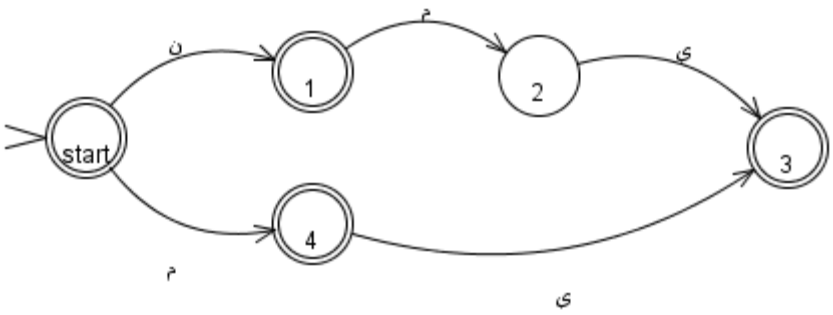
بنابر آنچه در بالا گفته شد، برای یک ریشه‌یابی مؤثر به عنوان یک پیش‌پردازش برای دسته‌بندی، می‌باید از یک ریشه‌یاب بسیار ساده برای صرف تصریفی به منظور کمک به افزایش نسبی دقت دسته‌بندی متون استفاده شود.[1] تاکید می شود که هدف از ریشه‌یابی نه به منظور یکی از شاخه‌های زبان‌شناسی می‌باشد بلکه برای کمک به سرعت و دقت نهایی سیستم دسته‌بندی متون بکار رود. به عبارت دیگر نیاز به پرسه‌های تصحیح برای حالت‌های خاص مانند تبدیل "ه" به "گ" در هنگام اضافه نمودن "ان" جمع به انتهای اسم می‌باشد (مانند: واژه + "ان" = واژگان).

برای این منظور، می‌باید از یک ریشه‌یاب حذف پس‌وند (پیش‌وند) استفاده گردد که به طور اتوماتیک طولانی‌ترین پس‌وند (پیش‌وند) ممکن را از انتهای کلمه (مستقل از درست بودن یا نادرست بودن مانده) حذف می‌سنماید.

ایده‌ی اصلی استفاده از اتومات های قطعی محدود[2] توسط پورتر ارائه شده است [121]. همان‌طور که گفته شد، اکثر کارهای انجام شده برای زبان فارسی نیز عموماً از همین روش بهره برده‌اند. لذا در این‌جا نیز، بر مبنای آنچه در بخش 6.3 آورده شد، دو اتومات قطعی محدود ساده و نسبتاً کامل صرف تصریفی، که یکی برای حذف پیش‌وندها و دیگری برای حذف پس‌وندهاست، ارائه شده است.

[1] افزایش مطلق دقت الگوریتم نیازمند یک ریشه‌یاب سنگین و دقیق به همراه واژگان ریشه لغات و دیگر پرسه های تصحیح می باشد که این خود علاوه سربار زیاد حافظه ای و پردازشی در بسیاری از موارد، منجر به کاهش سرعت سیستم دسته‌بندی نیز می شود.

[2] Deterministic Finite Automata



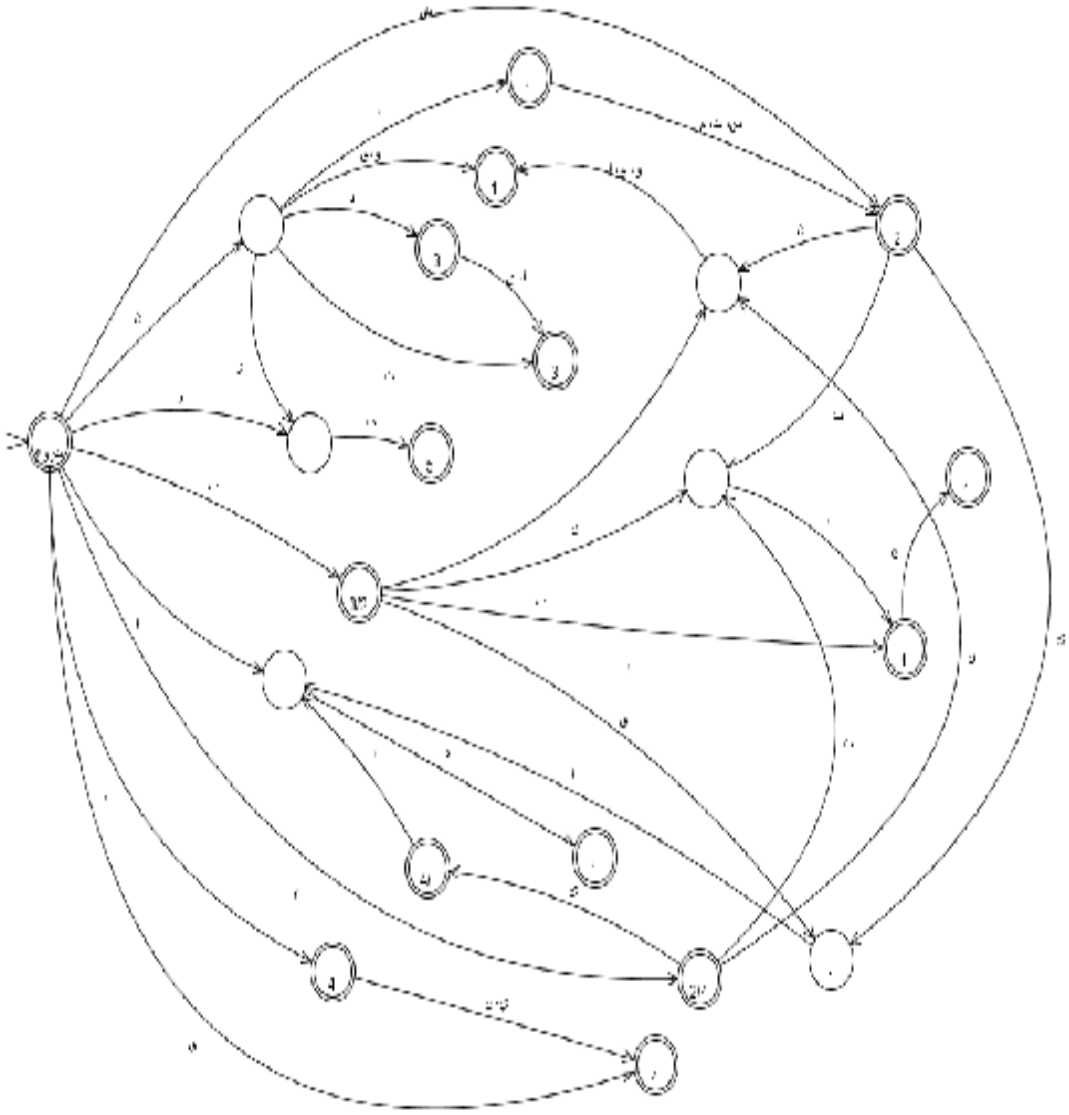
شکل 6 نمایه NFA (None-Deterministic Finite Automata) ارائه شده در این‌جا برای پیش‌وندهای ریشه‌یاب

نمایه اتوماتای ریشه‌یاب در شکل های 6 و 7 آورده شده است. در شکل 6 و 7 برای ساده‌تر و خوانا شدن تصویر "انتقال غیر" حذف شده است. منظور از انتقال غیر، یعنی در هر گره پایانی در صورتی که کاراکتر داده شده به غیر از یکی از کاراکترهای انتقال بود سیستم از اتومات خارج می‌شود. باید ذکر کرد که در شکل 7، اعداد نوشته شده در هر کدام از وضعیت[1]ها، شماره وضعیت نیست بلکه نوع پس‌وند را نشان می دهد (جدول 9).

جدول 9 کد پس‌وندهای استفاده شده در شکل 7

نوع پس‌وند	کد
"ی" صامت میانجی	-
علامت‌های جمع	1
نشانه مفعولی و اضافی	2
پس‌وندهای بن فعل برای مصدرهای باقاعده	3
پس‌وندهای تصریف زمان حال	4
"تر" و "ترین"	5

[1] State



شکل 7 نمایه NFA (None-Deterministic Finite Automata) ارائه شده در این‌جا برای پس‌وندهای ریشه‌یاب

7-1-4. گروه‌های اسمی

در این سیستم این قسمت یک ماژول تهی است و بیش‌تر به منظور جامعیت سیستم دیده شده است. این ماژول وظیفه گروه‌بندی عبارت های به دست آمده را برعهده دارد (البته در بعضی از سیستم ها استخراج عبارت ها نیز در این بخش صورت می پذیرد). این مسأله هم به کاهش بیش‌تر ابعاد مسأله کمک می کند و هم از خود گروه‌های اسمی در بهتر شدن دقت الگوریتم می‌توان بهره گرفت. در صورت نیاز یا برای ارتقای سیستم می‌توان این قسمت را تکمیل‌تر کرد.

یک پیشنهاد ساده برای این منظور می‌تواند مقایسه عبارت‌ها استخراج شده توسط یک الگوریتم استخراج عبارت‌ها با یک جدول ثابت باشد. پیشنهاد دیگر می‌تواند با استفاده از یک ریشه‌یاب صرف اشتقاقی یا دیگر الگوریتم‌های پیچیده‌تر صورت پذیرد.

7-1-5. تحلیل ساختاری

این ماژول وظیفه تحلیل ساختاری متن را برعهده دارد. یک کاربرد از این مسأله می‌تواند در نمونه‌های کاربردی برای پردازش صفحات ابر متن باشد. در این سیستم تمام متن‌ها مستقل از قالب می باشند.

7-2. وزن‌دهی

همان‌طور که در فصل 5 به تفضیل بحث شد، روش‌های زیادی برای وزن‌دهی پیشنهاد شده است که در این‌جا از متداول ترین آن ها به نام TFIDF استفاده شده است.

7-3. کاهش ابعاد

همان‌طور که پیش‌تر گفته شد، کاهش ابعاد تاثیر به سزایی در سرعت، دقت، و کاهش پیش نیازهای سخت‌افزاری برای سیستم دسته‌بندی متون فراهم می‌آورد. باید ذکر کرد که در این

سیستم علاوه بر استفاده از ریشه‌ی کلمات به جای خود کلمات و حذف کلمات بی‌ارزش، ماژول مستقلی برای کاهش ابعاد دیده شده است.

در این‌جا یک رویه‌ی ساده برای کاهش ابعاد استفاده شده است و در صورت نیاز می‌توان این رویه را ارتقا داد. لذا، به منظور کاهش ابعاد رویه زیر اجرا می‌گردد:

- فرکانس کلمات محاسبه می شود و کلمات بر مبنای فرکانس تکرار مرتب می‌شوند.
- کلماتی که فرکانس تکرارشان از 0.1 فرکانس میانگین کم‌تر بود، حذف می شود.

7-4. روش‌های دسته‌بندی

ماژول دسته‌بندی، وظیفه دسته‌بندی متون حاصله را بر عهده دارد. در این‌جا با استفاده از مدل تئوری بیز، مدل 2-gram، 3-gram، به عنوان مدل های اصلی استفاده شده است. هم‌چنین در یک حالت دیگر با استفاده از لغزاندن یک پنجره به طول 2 و اعمال یک مدل بیز بر روی آن نیز استفاده شده است. شرح مدل n-gram در بخش 5.4 به تفضیل آمده است.

8. نتایج تجربی

8-1. بانک اطلاعاتی

یکی از قسمت‌های مهم برای نرم‌افزار دسته‌بندی متون فارسی، طراحی و ایجاد بانک اطلاعاتی مناسب برای نگهداری اطلاعات مجموعه‌ی آموزشی، مجموعه‌ی آزمایشی، سطح منطقی متون و هم‌چنین اطلاعات استخراجی از کلاس‌بند برای طبقه‌بندی مناسب می‌باشد. نوع و ساختار بانک اطلاعاتی به جهت همواری و یا سلسله‌مراتبی، نوع الگوریتم‌هایی را که می‌توان توسط این مجموعه آزمایش شوند را مشخص می‌نماید.

ولی برای زبان فارسی کارهای پراکنده‌ای انجام گرفته است که بعضاً برای این کار کافی یا در دسترس نبوده اند [16] [17]. برای نمونه در [16] تعداد بسیار محدودی از پایان نامه ها، چکیده مقالات یا فصلی از یک کتاب یا پایان نامه گرداوری شده است که هم به لحاظ طول و دیگر مشخصات اختلافات بسیار زیاد داشتند و هم تنها به نوشته‌جات مربوط به کامپیوتر محدود بودند.در این پروژه یک بانک اطلاعاتی رابطه‌ای (هموار) جهت آموزش و آزمایش الگوریتم‌های دسته‌بندی هموار متون تهیه گردیده است.

برای آزمایش سیستم نیاز به تهیه یک مجموعه برای آموزش و آزمایش این سیستم بود. با توجه به اینکه تاکنون هیچ مجموعه‌ای برای دسته‌بندی متون فارسی تهیه نشده است. ابتدا سعی گردید تا مجموعه داده‌های استانداردی برای این منظور فراهم گردد. لذا در تهیه مجموعه داده ها، می‌بایست فاکتورهای یک‌نواختی (متن‌های انتخاب شده با توجه به دسته هایشان تفکیک پذیر باشند و هم‌پوشانی دسته‌ها و وجود متن‌های دو پهلو از حد معینی بیش‌تر نباشد) و پوشا بودن (اخبار مجموعه‌ی آزمایشی در ارتباط با مجموعه‌ی آموزشی باشد) رعایت گردد. مشکل اول در هنگامی رخ می‌دهد که دسته‌های انتخاب‌شده زیرمجموعه یکدیگر باشند. در این حالت می‌توان به جای استفاده از زیردسته‌ها از دسته‌های کلی استفاده کرد. مشکل دوم هنگامی رخ می‌دهد که اخبار وقایع و یا حوادث رخ‌داده برای یک دسته در مجموعه‌ی آموزشی نسبت به مجموعه‌ی آزمایشی در یک راستا نباشد. برای مثال برای دسته "بلایای طبیعی" در مجموعه‌ی آموزشی بیش‌تر خبرها در ارتباط با سیل باشد در حالی که در مجموعه‌ی آزمایشی در ارتباط با زلزله باشد. در این‌جا می‌توان به یکی از

دو روش متداول عمل نمود: 1- خبرهای مجموعه‌ی آموزشی و آزمایشی از بازه‌ای نسبتا طولانی (برای مثال در حدود 10 سال) انتخاب شود. 2- اخبار مجموعه‌ی آموزشی و آزمایشی از دو بازه نزدیک به هم انتخاب شود.

برای آزمایش سیستم از 10704 خبر دانلود شده از سایت ایسنا[1] (خبرگزاری دانشجویان ایران) از ماه مارچ تا ماه آگوست سال 2005 استفاده شده است. اخبار در سه دسته‌ی کلی اجتماعی، سیاسی و فرهنگی تقسیم شده که 6314 خبر از ماه‌های جون، جولای و آگوست برای آموزش و 4390 خبر از ماه‌های مارچ، آوریل و می‌سال 2005 برای آزمایش سیستم تعیین شده است (جدول 10). با توجه به آنچه گفته شد، برای رفع مشکل اول از سه دسته‌ی کلی "اجتماعی"، "سیاسی" و "فرهنگی" و برای رفع مشکل دوم از دو بازه زمانی نزدیک بهم استفاده گردید.

بانک آموزش و آزمایش متون فارسی بر اساس 10704 خبر از مجموع 54298 اخبار دانلود شده از وب سایت ایسنا[2] تهیه گردیده است.

8-1-1. کافی بودن

یکی از مسائل مهم در حوزه طبقه‌بندی متون، تعداد متون مجموعه‌ی آموزشی و آزمایشی می‌باشد. با توجه به تحقیقات و بررسی‌های انجام شده دست کم 2000 متن در هر دسته برای یادگیری باید در نظر گرفته شود.

8-1-2. یک‌نواختی

یک‌نواختی یعنی متون هر دسته با صراحت کافی از یک گروه تلقی گردد و هم‌پوشانی دسته‌ها از حد مشخصی بیش‌تر نباشد. به عبارت دیگر متون دو پهلو به طوری که نتوان به راحتی آن‌ها را در یک دسته قرار داد از حد معینی کم‌تر باشد. این گونه متون مانع از یادگیری 100% مجموعه‌ی آموزشی توسط الگوریتم می‌گردند. در حقیقت حد بالای آموزش الگوریتم‌های یادگیرنده با میزان

[1] http://www.isna.ir

[2] http://www.isna.ir

هم‌پوشانی دسته‌ها محدود می‌گردند. در این پروژه بین دسته‌های اجتماعی و فرهنگی به دلیل ماهیت موضوعی، نزدیکی و هم‌پوشانی طبیعی وجود دارد. لذا متون این دو دسته علی‌رغم ویرایش‌های انجام شده از میزان یک‌نواختی کم‌تر از 90 در صد برخوردار هستند ولی متون دسته‌ی سیاسی از یک‌نواختی تقریبی 90 درصد بر خوردارند.

طبق تعریف هم‌پوشانی توسط نسبت زیر محاسبه می‌گردد.

کل متون آن دسته / متونی که دقیقا متعلق به آن دسته هستند = میزان یک‌نواختی

با توجه به حجم متون به صورت تصادفی از هر 5 متن یک متن بررسی و ویرایش هم‌پوشانی شده است.

8-1-3. پوشا بودن

به منظور استاندارد بودن متون جمع‌آوری شده برای تشخیص صحت الگوریتم می‌توان به دو طریق عمل نمود.

۱- متون آموزشی و آزمایشی به طور نرمال و یک‌نواخت در یک بازه طولانی از حوادث انتخاب گردد.

۲- متون آموزشی و آزمایشی در دو بازه زمانی کوتاه ولی پشت سر هم انتخاب گردد.

در مواردی که حجم زیادی از متون در دسترس باشد روش اول و در غیر این صورت روش دوم انتخاب می‌گردد. بانک اطلاعاتی تهیه شده حاضر از دسته دوم می‌باشد.

8-2.مجموعه‌ی آموزشی

در این پروژه بخشی از بانک اطلاعاتی برای آموزش سیستم به کار می‌رود. برای این منظور 4390 متن از اخبار ماه‌های مارچ، آوریل و می‌سال 2005 برای آموزش سیستم در 3 دسته استفاده شده است (جدول 10). این اطلاعات از ساختار آورده شده در جدول 11 پیروی می‌کند.

8-3.مجموعه‌ی آزمایشی

اطلاعات آزمایشی به منظور تعین دقت الگوریتم به کار می‌رود. لذا در این‌جا از 6314 متن از و اخبار ماه‌های جون، جولای و آگوست سال 2005 برای آزمایش سیستم استفاده شده است. همان‌طور که در قسمت‌های قبل گفته شد، در این سیستم برای ارضای شروط سه گانه فوق از اخبار سال 2005 ایسنا استفاده گردیده است. (جدول 10 و 11)

جدول 10 ساختار مجموعه‌ی آموزشی و آزمایشی بانک اطلاعاتی ایسنا 10704

		آموزشی	آزمایشی
اجتماعی		2349	1482
	آسیب‌های اجتماعی	44	29
	اجتماعی - عمومی	14	4
	اجتماعی - آموزش و پرورش	108	101
	اجتماعی - جوانان	84	68
	اجتماعی - حوادث	4	-
	اجتماعی - خانواده	42	19
	اجتماعی – زنان	75	72

		آموزشی	آزمایشی
	اجتماعی - شهری	468	430
	اجتماعی - محیط زیست	249	144
	اجتماعی - معارف	2	1
	حج و زیارت	109	129
	حوادث	1000	749
	کار و اشتغال	150	96
سیاسی		2164	1402
	آسیای میانه ، روسیه	121	129
	افغانستان	75	15
	بین‌الملل	263	103
	پاکستان ، هند و چین	14	4
	ترکیه	22	9
	حاشیه خلیج فارس	4	1
	سیاسی خارجی	979	690
	سیاسی خارجی – ایران	264	157
	سیاسی خارجی - ایران در جهان	417	291
	عراق	5	3
فرهنگی		1801	1146

		آموزشی	آزمایشی
	فرهنگ و هنر – تئاتر	339	330
	فرهنگ و هنر - رادیو و تلویزیون	293	1
	فرهنگ و هنر – سینما	734	478
	فرهنگ و هنر – موسیقی	110	83
	فرهنگ و هنر - هنرهای تجسمی	325	254
جمع کل		6314	4390

8-4. گروه‌های اسمی

استفاده از گروه‌های اسمی به جای کلمات و عبارت در دسته‌بندی متون بسیار مهم می‌باشد. برای این منظور گروه‌های اسمی در جدولی دیگر به طور جداگانه نگهداری می‌گردد. نوع این گروه‌های اسمی نیز در جدول دیگری قرار دارد.

جدول 11 انواع عبارت‌های اسمی

کد	عنوان
1	کلمات بی ارزش
2	اسم مکان
3	عبارات مختصر شده
4	رزرو شده
5	اسم خاص

8-4-1. اسم مکان

دسته‌ای از کلمات که اسم مکان خاصی باشند همچون اسم شهرها و کشورها در این جدول نگهداری می‌گردد. این جدول برای ماژول تشخیص گروه‌های اسمی که در استخراج ترم‌ها (پیش‌پردازش الگوریتم‌های دسته‌بندی متون) استفاده می‌گردد، به کار می‌رود (جدول 12).

جدول 12 لیست اسم مکان

اسم مکان
سایگون، شیکاگو، وارسا، آبادان، آذربایجان، آذربایجان‌شرقی، آذربایجان‌غربی، آستارا، آلمان، آمریکا ، آمل، آنکارا، آنگولا، اتیوپی، اراک، اردبیل، اردستان، اردن، ارسنجان، ارمنستان، اروپا ، ارومیه، ازبکستان، اسپانیا ، اسدآباد ، اسراییل، اشتهارد ، اصفهان، افغانستان، امان، انزلی، انگلیس، اهرم، اهواز، اوکراین، ایالات‌متحده، ایتالیا ، ایران، ایروان، ایلام، باکو ، بانکوک ، بجنورد ، بروجرد ، بقاع، بلاروس، بلژیک، بلغارستان، بلوچستان، بم، بندرانزلی، بندرعباس، بنگلادش، بوشهر، بویراحمد ، بیجار ، بیرجند ، بیروت، پاکستان، پکن، تاجیکستان، تانزانیا، تایلند ، ، ترکیه، تفلیس، تکاب، تنگستان، تهران، جمکران، جورجیا، چین، خاورمیانه، خراسان، خرم‌آباد ، خرمشهر ، خمین، خواف، خوزستان، دربندیخان، دزفول، دمشق، دورود ، رباط کریم، رشت، رشتخوار ، ریاض، زابل ، زاهدان، زنجان، ساری، ساوه، سمنان، سنندج، سوئدی ، سودان، سوریه، سوییس ، سیستان، شاهرود ، شبستر ، شمیرانات، شهرکرد ، شهریار ، شیراز، صومعه‌سرا، عراق، عربستان، فرانسه ، فلسطین، قره‌باغ، قزاقستان، قزوین، قشم، قطر، قفقاز، قم، کابل، کاشان، کانادا ، کرانی، کردستان، کرمان، کرمانشاه ، کره جنوبی، کره‌جنوبی، کنگو ، کهریزک، کهگیلویه، کوالالامپور ، گرجستان، گرگان، گرمسار ، گیلان، لاریجان، لبنان، لس‌آنجلس، لندن، مادرید، مازندران، مالزی، مسکو ، مشهد ، مکزیک ، مهاباد ، مونترال ، نیجریه، نیکشهر، نیویورک ، هرمزگان، هلند ، همدان، هند ، هنگ‌کنگ، واشنگتن، ویتنام، وین، یاسوج، یزد ، یمن، یوگسلاوی، سیرجان، سیرالئون، ژاپن، برزیل، چالوس، نیکاراگوئه، دهلی، سریلانکا، نپال، بوتان، نیوزیلند، شیروان، ورزقان، اسفراین، براکو، گناباد، تربت‌حیدریه، کفرکلا، یزد، دهلران

8-4-2. اسامی خاص

اسامی خاص، به کلماتی گفته می‌شود که برای نامیدن اشخاص استفاده می‌گردد. این جدول برای استخراج ویژگی‌های گروه‌های اسمی کاربرد دارد. باید توجه داشت که درحالت کلی اسم خاص به تنهایی مبین مفهوم خاصی نمی باشد. برای مثال نمی‌توان از اسم خاص "خاتمی" جز در مواردی که اطلاعات تکمیلی همچون زمان ریداد و غیره مشخص باشد، استنباط یک متن سیاسی نمود.

جدول 13 لیست اسامی خاص

اسم خاص
جونیچیرو ، حسین، خیرالله، رحمان، سید ، عبدالرحمان، عثمان، غلامعلی، مصیب، هنری، یورگ، ابراهیم، ابوالفضل، ابوالفضل‌العباس، ابوفخر، احمد ، اصغر ، اعلایی، افروغ، افضلی‌نژاد ، خمینی، اکبر ، الیاسی، انتونی، انصاری، بمانیان، بهروش، تاجدین، جوادی، حاتمیان، حاجیلو ، حبیب، حداد ، حسین‌زاده، خاتمی، خادمی، خبیر ، خداداد ، خرازی، راد، ربیع ، رحیم، رسولی، رضا ، زهره، شریفلو، صدرایی،طباطبایی، طباطبایی‌نژاد، عالی‌پور ، عامری، علی، علیرضا ، غدیانی، غریب‌پور، فتاحی، فرج‌الله، فرشاد ، فریبرز ، فلاح، قنبر ، کریم‌نژاد ، کریمی، کویزومی، گراهام، گرین، لاری، مجتبی، محمد ، محمودیان، مختاری، مدنیان، مرضیه، مظاهر ، مظاهری، مقدسی، مقیمی، منوچهر ، مهدی، مورتیمور، موهبتی، نارملا، نصرتیان، نوری‌همدانی، هایدر، وجه‌الله، یوسف

8-4-3. عبارات مختصر شده

جدول 14 عبارات مختصر شده

عبارت مختصر شده	شرح
ایرنا	خبر گزاری جمهوری اسلامی
ایسنا	خبر گزاری دانشجویی
صا	صنایع الکترونیک
مهر	مدیریت هوشمند رایانه ای

8-5.ارزیابی

برای ارزیابی سیستم ابتدا کلیه علامت‌های نشانه‌گذاری، اعداد و کلمات بی ارزش را از متن حذف نموده و سپس تئوری بیز با در نظر گرفتن کلمه (جدول 14) و یا دو کلمه مجاور (جدول 15) به عنوان ویژگی و مدل زبانی n-gram با در نظر گرفتن n=2 (جدول 16) و n=3 (جدول 17) بر روی مجموعه داده‌های جمع‌آوری شده، پیاده‌سازی و آزمایش گردید.

جدول 15 نتایج برای الگوریتم بیز با در نظر گرفتن کلمه

	آموزشی			آزمایشی		
دسته‌های کلی	**اجتماعی**	**سیاسی**	**فرهنگی**	**اجتماعی**	**سیاسی**	**فرهنگی**
اجتماعی	2305	21	23	1731	34	77
سیاسی	36	2115	13	35	1358	9
فرهنگی	14	6	1781	7	4	1135

جدول 16 نتایج برای روش پیشنهادی (الگوریتم بیز با در نظر گرفتن انتقال کلمات)

	آموزشی			آزمایشی		
دسته‌های کلی	**اجتماعی**	**سیاسی**	**فرهنگی**	**اجتماعی**	**سیاسی**	**فرهنگی**
اجتماعی	2348	1	0	1170	35	37
سیاسی	0	2164	0	19	1381	2
فرهنگی	6	0	1795	9	25	1112

جدول 17 نتایج برای الگوریتم n-gram برای n=2

	آموزشی			آزمایشی		
دسته‌های کلی	**اجتماعی**	**سیاسی**	**فرهنگی**	**اجتماعی**	**سیاسی**	**فرهنگی**
اجتماعی	2349	0	0	1740	32	70
سیاسی	1	2163	0	29	1360	13
فرهنگی	6	0	1795	10	8	1128

جدول 18 نتایج برای الگوریتم n-gram برای n=3

	آموزشی			آزمایشی		
دسته‌های کلی	**اجتماعی**	**سیاسی**	**فرهنگی**	**اجتماعی**	**سیاسی**	**فرهنگی**
اجتماعی	2349	0	0	1504	70	268
سیاسی	2	2161	1	121	1071	210
فرهنگی	6	0	1795	23	11	1112

همان‌طور که در جدول 18 و شکل 8 و 9 نشان داده شده است:

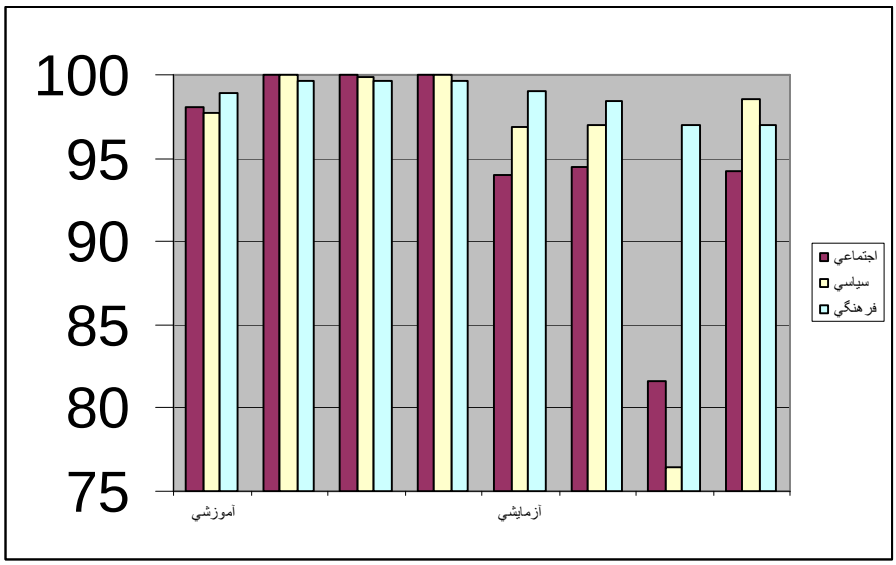
۱. مدل n-gram دقت بالاتری نسبت به تئوری بیز ساده دارد. به عبارت دیگر به دلیل استفاده از مدل زبانی و دخیل نمودن موقعیت کلمه نسبت به کلمات ماقبل خود، نتیجه از دقت بهتری برخوردار بوده است.

۲. در مدل n-gram، n=2 دقتی بیش از n=3 تولید می‌کند. بدیهی است که احتمال رخداد کلمه در صورتی که دنباله مشخصی از کلمات ماقبل آن رخ داده باشد، با افزایش طول دنباله، کاهش می‌یابد؛ به عبارت دیگر این رخداد منحصر بفردتر می‌گردد و در نتیجه اغلب کیفیت نتایج بدون افزایش مناسب متون کاهش می‌یابد.

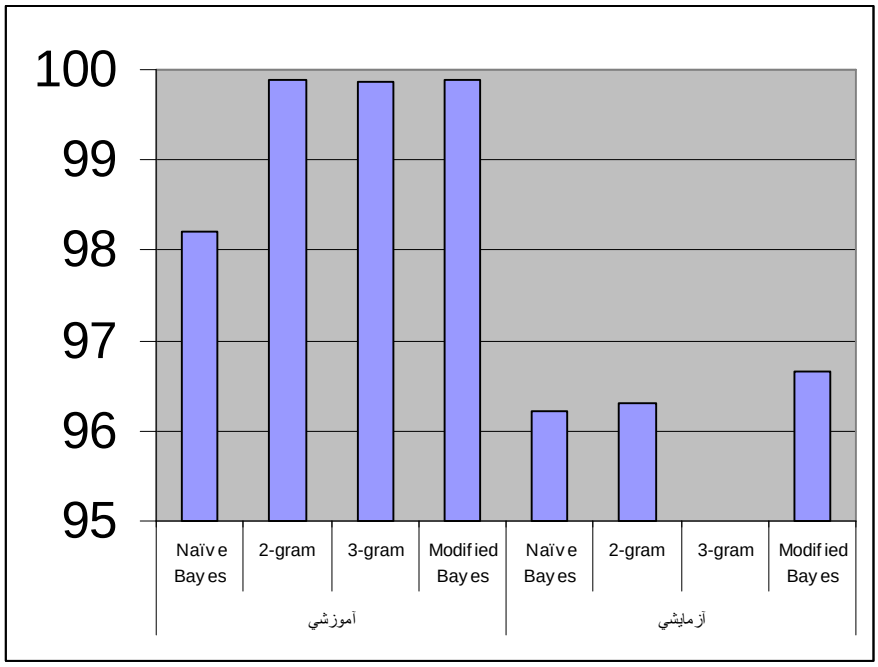
۳. در مجموع استفاده از دو کلمه متوالی به جای یک کلمه بهترین دقت را تولید می‌کند؛ همان‌طور که در بخش 2 گفته شد، کلمات فارسی در بسیاری از موارد به صورت منقطع نوشته می‌شوند و استفاده از دو کلمه متوالی به جای یک کلمه به تنهایی، کیفیت الگوریتم را بهتر می‌کند.

جدول 19 مقایسه نتایج برای روش‌های آزمایش شده و انتخاب ویژگی پیشنهادی

	آموزشی				آزمایشی			
دسته‌های کلی	**Naïve Bayes**	**2-gram**	**3-gram**	**روش پیشنهادی**	**Naïve Bayes**	**2-gram**	**3-gram**	**روش پیشنهادی**
اجتماعی	98.12686	100	100	99.95743	93.97394	94.46254	81.65038	94.2029
سیاسی	97.73567	99.95379	99.86137	100	96.86163	97.00428	76.39087	98.50214
فرهنگی	98.88951	99.66685	99.66685	99.66685	99.04014	98.42932	97.03316	97.03316
در مجموع	98.21033	99.88914	99.85746	99.88914	96.21868	96.30979	83.98633	96.64908



شکل8 مقایسه نتایج حاصله به تفکیک دسته‌ها برای روش‌های آزمایش شده و روش پیشنهادی



شکل 9 مقایسه نتایج حاصله به صورت کلی برای روش‌های آزمایش شده و روش پیشنهادی

9. نتیجه‌گیری

امروزه با توجه به گسترش روزافزون متون الکترونیکی ضرورت وجود ابزارهای دسته‌بندی متون بیش از پیش احساس می‌گردد. از طرف دیگر، اگرچه زبان فارسی از دسته زبان‌های هند و اروپایی می‌باشد ولی به طور کلی از رسم‌الخط عربی استفاده می‌کند. با توجه به آنچه در بخش 2 آمد، معمولا در زبان فارسی یافتن دقیق مرز بین کلمات دشوار است. در این مقاله سعی گردید تا به جای استفاده از کلمات (رشته‌های جدا شده توسط یک فضای خالی)، دو کلمه متوالی را به عنوان ویژگی پیشنهاد دهد یا به عبارت دیگر یک پنجره به طول 2 کلمه بر روی کل متن لغزانده شود. این تغییر ساده باعث گردید تا دقت نتایج الگوریتم بیز برای مجموعه‌ی آموزشی به دقت مدل n-gram برسد و برای مجموعه‌ی آزمایشی حتی از کلیه روش‌های توضیح داده شده پیشی بگیرد. (دقت نتایج حدوداً 3. درصد در مجموعه‌ی آزمایشی از بهترین جواب، بهتر شده است.)

مراجع

[1] Salton G., McGill M.J., “Introduction to Modern Information Retrival”, McGraw Hill, New York, 1983.

[2] Susana Eyheramendy, Alexander Genkin, Wen-Hua Ju, David D. Lewis, David Madigan, " Sparse Bayesian Classifiers for Text Categorization", Joint Statistical Meeting in San Francisco, California, 2003.

[3] FABRIZIO SEBASTIANI, "Machine Learning in Automated Text Categorization", ACM Computing Surveys, Vol. 34, No. 1, pp. 1–47, March 2002.

[4] M. Granitzer, "Hierarchical text classification using methods from machine learning", Master's Thesis, Graz University of Technology, 2003.

[5] CLEVERDON, C. 1984. Optimizing convenient online access to bibliographic databases. Inform. Serv. Use 4, 1, 37–47. Also reprinted in Willett [1988], pp. 32–41.

[6] PAZIENZA, M. T., ed. 1997. Information Extraction. Lecture Notes in Computer Science, Vol. 1299. Springer, Heidelberg, Germany.

[7] KNIGHT, K. 1999. Mining online text. Commun. ACM 42, 11, 58–61.

[8] JOACHIMS, T. AND SEBASTIANI, F. 2002. Guest editors’ introduction to the special issue on automated text categorization. J. Intell. Inform. Syst. 18, 2/3 (March-May), 103–105.

[9] LEWIS, D. D. AND HAYES, P. J. 1994. Guest editorial for the special issue on text categorization. ACM Trans. Inform. Syst. 12, 3, 231.

[10] MANNING, C. AND SCH¨UTZE, H. 1999. Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA.

[11] D. A. Grossman and O. Frieder, "Information Retrieval: Algorithms and Heuristics", Springer, 2004.

[12] BORKO, H. AND BERNICK, M. 1963. Automatic document classification. J. Assoc. Comput. Mach. 10, 2, 151–161.

[13] MERKL, D. 1998. Text classification with selforganizing maps: Some lessons learned. Neurocomputing 21, 1/3, 61–77.

[14] Alessandro Moschitti, "Answer Filtering via Text Categorization in Question Answering Systems", ICTAI, pp. 241-248, 2003.

[15] Huang, Y. "Support vector machines for text categorization based on latent semantic indexing", Technical report, Electrical and Computer Engineering Department, Johns Hopkins University.

[۱۶] مسعود تشکری، بررسی و ارزیابی روش‌های شاخص‌گذاری خودکار متون فارسی، دانشگاه امیر کبیر، استاد راهنما دکتر محمد رضا میبدی، 1380.

[۱۷] سعید ساعتی، خوشه‌بندی اسناد کتابخانه های دیجیتال با استفاده از اتومات های یادگیری توزیع شده و کلونی مورچه ها. دانشگاه امیر کبیر، استاد راهنما دکتر محمد رضا میبدی، 1384.

[۱۸] مریم آیت، یک گرامر محاسباتی برای زبان فارسی، دانشگاه امیر کبیر، استاد راهنما دکتر عبدالله زاده، 1380.

[19] D´IAZ ESTEBAN, A., DE BUENAGA RODR´IGUEZ, M., URE˜NA L´OPEZ, L. A., AND GARC´IA VEGA, M. 1998. Integrating linguistic resources in an uniform way for text classification tasks. In Proceedings of LREC-98, 1st International Conference on Language Resources and Evaluation (Grenada, Spain, 1998), 1197–1204.

[20] JUNKER, M. AND ABECKER, A. 1997. Exploiting thesaurus knowledge in rule induction for text classification. In Proceedings of RANLP-97, 2nd International Conference on Recent Advances in Natural Language Processing (Tzigov Chark, Bulgaria, 1997), 202–207.

[21] LARKEY, L. S. 1999. A patent search and classification system. In Proceedings of DL-99, 4th ACM Conference on Digital Libraries (Berkeley, CA, 1999), 179–187.

[22] YANG, Y. 1999. An evaluation of statistical approaches to text categorization. Inform. Retr. 1, 1–2, 69–90.

[23] LARKEY, L. S. AND CROFT, W. B. 1996. Combining classifiers in text categorization. In Proceedings of SIGIR-96, 19thACMInternational Conference on Research and Development in Information Retrieval (Z ¨ urich, Switzerland, 1996), 289–297.

[24] MARON, M. 1961. Automatic indexing: an experimental inquiry. J. Assoc. Comput. Mach. 8, 3, 404–417.

[25] MYERS, K., KEARNS, M., SINGH, S., AND WALKER, M. A. 2000. A boosting approach to topic spotting on subdialogues. In Proceedings of ICML-00, 17th International Conference on Machine Learning (Stanford, CA, 2000), 655– 62.

[26] SCHAPIRE, R. E. AND SINGER, Y. 2000. BoosTexter: a boosting-based system for text categorization. Machine Learning. 39, 2/3, 135–168.

[27] SABLE, C. L. AND HATZIVASSILOGLOU, V. 2000. Textbased approaches for non-topical image categorization. Internat. J. Dig. Libr. 3, 3, 261–275.

[28] FORSYTH, R. S. 1999. New directions in text categorization. In Causal Models and Intelligent Data Management, A. Gammerman, ed. Springer, Heidelberg, Germany, 151–185.

[29] CAVNAR, W. B. AND TRENKLE, J. M. 1994. N-grambased text categorization. In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval (Las Vegas, NV, 1994), 161–175.

[30] KESSLER, B., NUNBERG, G., AND SCH¨UTZE, H. 1997. Automatic detection of text genre. In Proceedings of ACL-97, 35th Annual Meeting of the Association for Computational Linguistics (Madrid, Spain, 1997), 32–38.

[31] LARKEY, L. S. 1998. Automatic essay grading using text categorization techniques. In Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval (Melbourne, Australia, 1998), 90–95.

[32] FIELD, B. 1975. Towards automatic indexing: automatic assignment of controlled-language indexing and classification from free indexing. J. Document. 31, 4, 246–265.

[33] GRAY, W. A. AND HARLEY, A. J. 1971. Computerassisted indexing. Inform. Storage Retrieval 7, 4, 167–174.

[34] HEAPS, H. 1973. A theory of relevance for automatic document classification. Inform. Control 22, 3, 268–278.

[35] FUHR, N. AND KNORZ, G. 1984. Retrieval test evaluation of a rule-based automated indexing (AIR/PHYS). In Proceedings of SIGIR-84, 7th ACM International Conference on Research and Development in Information Retrieval (Cambridge, UK, 1984), 391–408.

[36] ROBERTSON, S. E. AND HARDING, P. 1984. Probabilistic automatic indexing by learning from human indexers. J. Document. 40, 4, 264–270.

[37] TZERAS, K. AND HARTMANN, S. 1993. Automatic indexing based on Bayesian inference networks. In Proceedings of SIGIR-93, 16th ACM International Conference on Research and Development in Information Retrieval (Pittsburgh, PA, 1993), 22–34.

[38] BELKIN, N. J. AND CROFT, W. B. 1992. Information filtering and information retrieval: two sides of the same coin? Commun. ACM 35, 12, 29–38.

[39] HAYES, P. J., ANDERSEN, P. M., NIRENBURG, I. B., AND SCHMANDT, L. M. 1990. Tcs: a shell for content-based text categorization. In Proceedings of CAIA-90, 6th IEEE Conference on Artificial Intelligence Applications (Santa Barbara, CA, 1990), 320–326.

[40] ANDROUTSOPOULOS, I., KOUTSIAS, J., CHANDRINOS, K. V.,AND SPYROPOULOS, C. D. 2000. An experimental comparison of naive Bayesian and keywordbased anti-spam filtering with personal e-mail messages. In Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval (Athens, Greece, 2000), 160–167.

[41] DRUCKER, H., VAPNIK, V., AND WU, D. 1999. Automatic text categorization and its applications to text retrieval. IEEE Trans. Neural Netw. 10, 5, 1048–1054.

[42] LIDDY, E. D., PAIK, W., AND YU, E. S. 1994. Text categorization for multiple users based on semantic features from a machine-readable dictionary. ACM Trans. Inform. Syst. 12, 3, 278–295.

[43] LEWIS, D. D. 1995. The TREC-4 filtering track: description and analysis. In Proceedings of TREC-4, 4th Text Retrieval Conference (Gaithersburg, MD, 1995), 165–180.

[44] HULL, D. A. 1994. Improving text retrieval for the routing problem using latent semantic indexing. In Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval (Dublin, Ireland, 1994), 282–289.

[45] HULL, D. A., PEDERSEN, J. O., AND SCH¨UTZE, H. 1996. Method combination for document filtering. In Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval (Z ¨ urich, Switzerland, 1996), 279–288.

[46] SCHAPIRE, R. E., SINGER, Y., AND SINGHAL, A. 1998. Boosting and Rocchio applied to text filtering. In Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval (Melbourne, Australia, 1998), 215–223.

[47] SCH¨UTZE, H., HULL, D. A., AND PEDERSEN, J. O. 1995. A comparison of classifiers and document representations for the routing problem. In Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval (Seattle, WA, 1995), 229–237.

[48] KORFHAGE, R. R. 1997. Information Storage and Retrieval. Wiley Computer Publishing, New York, NY.

[49] AMATI, G. AND CRESTANI, F. 1999. Probabilistic learning for selective dissemination of information. Inform. Process. Man. 35, 5, 633–654.

[50] IYER, R. D., LEWIS, D. D., SCHAPIRE, R. E., SINGER, Y., AND SINGHAL, A. 2000. Boosting for document routing. In Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management (McLean, VA, 2000), 70–77.

[51] KIM, Y.-H.,HAHN, S.-Y., AND ZHANG, B.-T. 2000. Text filtering by boosting naive Bayes classifiers. In Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval (Athens, Greece, 2000), 168–175.

[52] TAURITZ, D. R., KOK, J. N., AND SPRINKHUIZEN-KUYPER, I. G. 2000. Adaptive information filtering using evolutionary computation. Inform. Sci. 122, 2–4, 121–140.

[53] YU, K. L. AND LAM, W. 1998. A new on-line learning algorithm for adaptive text filtering. In Proceedings of CIKM-98, 7th ACMInternational Conference on Information and Knowledge Management (Bethesda, MD, 1998), 156–160.

[54] GALE, W. A., CHURCH, K. W., AND YAROWSKY, D. 1993. A method for disambiguating word senses in a large corpus. Comput. Human. 26, 5, 415–439.

[55] ESCUDERO, G., M`ARQUEZ, L., AND RIGAU, G. 2000. Boosting applied to word sense disambiguation. In Proceedings of ECML-00, 11th European Conference on Machine Learning (Barcelona, Spain, 2000), 129–141.

[56] ROTH, D. 1998. Learning to resolve natural language ambiguities: a unified approach. In Proceedings of AAAI-98, 15th Conference of the American Association for Artificial Intelligence (Madison, WI, 1998), 806–813.

[57] ATTARDI, G., DI MARCO, S., AND SALVI, D. 1998. Categorization by context. J. Univers. Comput. Sci. 4, 9, 719–736.

[58] CHAKRABARTI, S., DOM, B. E., AND INDYK, P. 1998. Enhanced hypertext categorization using hyperlinks. In Proceedings of SIGMOD-98, ACM International Conference on Management of Data (Seattle, WA, 1998), 307–318.

[59] F¨URNKRANZ, J. 1999. Exploiting structural information for text classification on the WWW. In Proceedings of IDA-99, 3rd Symposium on Intelligent Data Analysis (Amsterdam, The Netherlands, 1999), 487–497.

[60] G¨OVERT, N., LALMAS, M., AND FUHR, N. 1999. A probabillistic description-oriented approach for categorising Web documents. In Proceedings of CIKM-99, 8th ACM International Conference on Information and Knowledge Management (Kansas City, MO, 1999), 475–482.

[61] OH, H.-J., MYAENG, S. H., AND LEE, M.-H. 2000. A practical hypertext categorization method using links and incrementally available class information. In Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval (Athens, Greece, 2000), 264–271.

[62] YANG, Y., SLATTERY, S., AND GHANI, R. 2002. A study of approaches to hypertext categorization. J. Intell. Inform. Syst. 18, 2/3 (March-May), 219–241.

[63] DUMAIS, S. T. AND CHEN,H. 2000. Hierarchical classification of Web content. In Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval (Athens, Greece, 2000), 256–263.

[64] CHAKRABARTI, S., DOM, B. E., AGRAWAL, R., AND RAGHAVAN, P. 1998a. Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. J. Very Large Data Bases 7, 3, 163–178

[65] KOLLER, D. AND SAHAMI, M. 1997. Hierarchically classifying documents using very few words. In ACM Computing Surveys, Vol. 34, No. 1, March 2002. Proceedings of ICML-97, 14th International Conference on Machine Learning (Nashville, TN, 1997), 170–178.

[66] MCCALLUM, A. K., ROSENFELD, R.,MITCHELL, T. M., AND NG, A. Y. 1998. Improving text classification by shrinkage in a hierarchy of classes. In Proceedings of ICML-98, 15th International Conference on Machine Learning (Madison, WI, 1998), 359–367.

[67] RUIZ, M. E. AND SRINIVASAN, P. 1999. Hierarchical neural networks for text categorization. In Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval (Berkeley, CA, 1999), 281–282.

[68] WEIGEND, A. S., WIENER, E. D., AND PEDERSEN, J. O. 1999. Exploiting hierarchy in text catagorization. Inform. Retr. 1, 3, 193–216.

[69] MITCHELL, T.M. 1996. Machine Learning. McGraw Hill, New York, NY.

[70] F. Sebastiani. Machine learning in automated text categorization. ACM Computing Surveys (CSUR), 34(1):1.47, 2002.

[71] APT´E, C., DAMERAU, F. J., AND WEISS, S. M. 1994. Automated learning of decision rules for text categorization. ACM Trans. on Inform. Syst. 12, 3, 233–251.

[72] DUMAIS, S. T., PLATT, J., HECKERMAN, D., AND SAHAMI, M. 1998. Inductive learning algorithms and representations for text categorization. In Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management (Bethesda, MD, 1998), 148–155.

[73] LEWIS, D. D. 1995a. Evaluating and optmizing autonomous text classification systems. In Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval (Seattle, WA, 1995), 246–254.

[74] SALTON, G. AND BUCKLEY, C. 1988. Term-weighting approaches in automatic text retrieval. Inform. Process. Man. 24, 5, 513–523. Also reprinted in Sparck Jones and Willett [1997], pp. 323–328.

[75] FUHR, N., HARTMANN, S., KNORZ, G., LUSTIG, G., SCHWANTNER, M., AND TZERAS, K. 1991. AIR/X—a rule-based multistage indexing system for large subject fields. In Proceedings of RIAO-91, 3rd International Conference "Recherche d'Information Assistee par Ordinateur" (Barcelona, Spain, 1991), 606–623.

[76] DENOYER, L., ZARAGOZA, H., AND GALLINARI, P. 2001. HMM-based passage models for document classification and ranking. In Proceedings of ECIR-01, 23rd European Colloquium on Information Retrieval Research (Darmstadt, Germany, 2001).

[77] FRASCONI, P., SODA, G., AND VULLO, A. 2002. Text categorization for multi-page documents: A hybrid naive Bayes HMM approach. J. Intell. Inform. Syst. 18, 2/3 (March–May), 195–217.

[78] SINGHAL, A., SALTON, G., MITRA, M., AND BUCKLEY, C. 1996. Document length normalization. Inform. Process. Man. 32, 5, 619–633.

[79] DAGAN, I., KAROV, Y., AND ROTH, D. 1997. Mistakedriven learning in text categorization. In Proceedings of EMNLP-97, 2nd Conference on Empirical Methods in Natural Language Processing (Providence, RI, 1997), 55–63.

[80] LEWIS, D. D., SCHAPIRE, R. E., CALLAN, J. P., AND PAPKA, R. 1996. Training algorithms for linear text classifiers. In Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval (Zurich, Switzerland, 1996), 298–306.

[81] NIGAM, K.,MCCALLUM, A. K., THRUN, S., ANDMITCHELL, T. M. 2000. Text classification from labeled and unlabeled documents using EM. Mach. Learn. 39, 2/3, 103–134.

[82] RILOFF. E. 1995. Little words can make a big difference for text classification. In Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval (Seattle, WA, 1995), 130–136.

[83] BAKER, L. D. AND MCCALLUM, A. K. 1998. Distributional clustering of words for text classification. In Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval (Melbourne, Australia, 1998), 96–103.

[84] COHEN, W. W. AND SINGER, Y. 1999. Contextsensitive learning methods for text categorization. ACM Trans. Inform. Syst. 17, 2, 141–173.

[85] WEISS, S. M., APT´E, C., DAMERAU, F. J., JOHNSON, D.E., OLES, F. J., GOETZ, T., AND HAMPP, T. 1999. Maximizing text-mining performance. IEEE Intell. Syst. 14, 4, 63–69.

[86] I. Moulinier, G. Raskinis, and J. Ganascia. Text categorization: a symbolic approach. In In Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval, 1996.

[87] I. Dhillon, S. Mallela, and R. Kumar. Enhanced word clustering for hierarchical text classification, 2002. to be published.

[88] M. W. Berry, S. T. Dumais, and G. W. O'Brien. Using linear algebra for intelligent information retrieval. Technical Report UT-CS-94-270, 1994.

[89] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. Journal of the American Society of Information Science, 41(6):391. 407, 1990.

[90] A. W. E. Wiener, J.O. Pedersen. A neural network approach to topic spotting. In Proceedings of SDAIR '95, pages 317.332, Las Vegas, NV, US, 1995.

[91] Yang, Y., Pedersen J.P. A, "Comparative Study on Feature Selectioin Text Categorization", Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97), pp412-420, 1997.

[92] Franca Debole , Fabrizio Sebastiani, Supervised term weighting for automated text categorization, Proceedings of the 2003 ACM symposium on Applied computing, Melbourne, Florida, March 09-12, 2003.

[93] Gary Noel Boone, "Extreme Dimensionality Reduction for Text Learning Cluster-generated Feature Space", Ph.D Thesis, Geogia Institute of Technology, August 2000.

[94] Mitchell, T. "Bayesian Learning In Machine Learning", WCB/McGraw-Hill, pp.154–200, 1997.

[95] Fuchun Peng, Dale Schuurmans, Shaojun Wang, "Language and Task Independent Text Categorization with Simple Language Models", Proceedings of HLT-NAACL, 2003.

[96] Susan Dumais, John Platt, David Heckerman, Mehran Sahami. "Inductive learning algorithms and representations for text categorization", CIKM '98, pages 148--155, 1998.

[97] Adam Wilcox, George Hripcsak, "Medical Text Representations for Inductive Learning", Proceeding of AMIA Symposium, 2000.

[98] R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification. Wiley-Interscience, 2001.

[99] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. In Knowledge Discovery and Data Mining, number 2, 1998.

[100] K.-R. M¨uller, S. Mika, G. R¨atsch, K. Tsuda, and B. Sch¨olkopf. An introduction to kernel-based learning algorithms. In IEEE Neural Networks, number 12(2), pages 181.201, 2001.

[101] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. In Proceedings of the seventh international conference on Information and knowledge management, pages 148.155. ACM Press, 1998.

[102] T. Joachims. Text categorization with support vector machines: learning with many relevant features. In Proceedings of ECML-98, 10th European Conference on Machine Learning, number 1398, pages 137.142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.

[103] Michael Granitzer, “Hierarchical Text Classi_cation using Methods from Machine Learning”, MS Thesis, Graz University of Technology, 2003

[۱۰٤]ایران کلباسی

[105] Karine Megerdoomian, "Unification-Based Persian Morphology", InProceedings of CICLing 2000, Alexander Gelbukh, Center ofInvestigation on Computation-IPN, Mexico, 2000.

[106] Luigi Galavotti, Fabrizio Sebastiani, Maria Simi, "Feature Selection and Negative Evidence in Automated Text Categorization, 2000.

[107] Hu, Yu, Irina Matveeva, John Goldsmith and Colin Sprague, “Using Morphology and Syntax Together in Unsupervised Learning.” ACL-05. Psychocomputational Models of Human Language Acquisition. Proceedings of the Workshop., 2005

[108] Oard, D. W., Levow, G.-A., and Cabezas, C. I. CLEF experiments at Maryland: Statistical stemming and backoff translation. In Cross-language information retrieval and evaluation:Proceedings of the CLEF 2000 workshop, C. Peters, Ed.: Springer Verlag, pp. 176-187, 2001.

[109] Pirkola, A. Morphological typology of languages for IR. Journal of Documentation, 57 (3), pp. 330-348, 2001.

[110] John A. Goldsmith, Derrick Higgins, Svetlana Soglasnova: Automatic Language-Specific Stemming in Information Retrieval. CLEF 2000: 273-284

[111] L. Larkey, L. Ballesteros, and M. Connell. Improving stemming for Arabic information retrieval: Light stemming and co-occurrence analysis. In Proceedings of ACM SIGIR, pages 269--274, 2002.

[112] Karine Megerdoomian, "Unification-Based Persian Morphology"

[113] Kraaij, W., Pohlmann, R. Viewing stemming as recall enhancement. In: Proceedings, 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '96) Zurich (1996) 40-48.

[114] Paice, C.D. Method for evaluation of stemming algorithms based on error counting. Journal of the American Society for Information Science 47 (8) (1996) 632-49

[115] Goldsmith, John. 2001. Unsupervised learning of the morphology of a natural language. Computational Linguistics, 27(2): 153-198.

[116] De Roeck, A. N. and Al-Fares, W. A morphologically sensitive clustering algorithm for identifying Arabic roots. In Proceedings ACL-2000. Hong Kong, 2000.

[117] Goweder, A. and De Roeck, A. Assessment of a significant Arabic corpus. Presented at the Arabic NLP Workshop at ACL/EACL 2001, Toulouse, France, 2001.

[118] Tanja Gaustad and Gosse Bouma, "Accurate Stemming of Dutch for Text Classification"

[119] Porter, M. F. An algorithm for suffix stripping. Program 14 (1980) 130-7.

[120] Kazem Taghva, Russell Beckley, Mohammad Sadeh: A Stemming Algorithm for the Farsi Language. ITCC (1) 2005: 158-162

[121] M. F. Porter. An algorithm for sux stripping. Program, 14(3):130137, 1980.